




4D Designmodus

-  Einleitung
-  4D Dateien verwalten
-  4D 64-bit Versionen nutzen
-  Einstellungen
-  Datenbank-Eigenschaften
-  Explorer
-  Runtime Explorer
-  Suchen und Ersetzen in der Struktur
-  Objekte bewegen
-  Datenbankstruktur erstellen
-  ORDA
-  Formulare verwalten
-  Formulare erstellen
-  Mit aktiven Objekten arbeiten
-  Eigenschaften für aktive Objekte
-  Listboxen
-  Objekte der Bibliothek
-  Unterformulare und Widgets
-  Ausgabeformulare und Berichte
-  Benutzerformulare
-  Methoden bearbeiten
-  Benutzer und Gruppen
-  Menüs und Menüleisten
-  Bildbibliothek
-  Hilfetipps
-  Auswahllisten
-  Stilvorlagen
-  Filter und Formate
-  Ressourcen Explorer
-  Datensätze verwalten
-  Datensätze suchen
-  Datensätze sortieren
-  Formeleditor
-  Schnellberichte
-  Schnellberichte (64-bit)
-  Etiketteneditor
-  Etiketteneditor (64-bit)
-  Daten importieren und exportieren
-  Web Services publizieren und einsetzen
-  Backup und Anwendung wiederherstellen
-  Maintenance und Security Center
-  Kompilieren
-  Eigenständige Anwendung erstellen und weitergeben
- 4D Komponenten entwickeln und installieren
- Anhänge
- Neuerungen der Version

Einleitung

-  Willkommen in 4D
-  Arbeitsumgebungen
-  Integrierte Server

4D kombiniert eine leistungsstarke relationale Datenbankanwendung, eine native integrierte SQL Engine, ein umfassendes multiplattformfähiges Entwicklungswerkzeug und einen dynamischen Web Server.

Mit 4D können Sie Ihre eigenen Daten verwalten oder Komplettlösungen für alle Anforderungen an ein Datenbank-Management System entwickeln. Sie können zum Beispiel:

- Eine Datenbankstruktur mit Tabellen und Feldern anlegen
- Unterschiedliche Formulare zum Eingeben, Ändern und Anzeigen von Datensätzen erstellen
- Datensätze suchen und sortieren
- Berichte oder Etiketten von Informationen aus der Datenbank erstellen
- Daten zwischen 4D Datenbanken und anderen Anwendungen importieren bzw. exportieren
- Ihre Datenbank im World Wide Web veröffentlichen

Mit 4D können Sie gängige Operationen der Datenverwaltung durch folgende Funktionalitäten verbessern:

- Der leistungsstarke Formularassistent ermöglicht, mit ein paar Mausklicks ausgeklügelte Formulare und Berichte zu erstellen
- Ein Kennwortsystem mit Zugriffsrechten schützt sensible Daten
- Sie können eigene Anwendungen mit eigenen Menüs, Dialogfenstern, Werkzeugleisten und Schaltflächen erstellen; diese Anwendungen lassen sich kompilieren und an Endbenutzer vertreiben
- Unzählige Web-Dienste nutzen, die innerhalb Ihrer Datenbank im Web veröffentlicht werden
- Die umfangreiche Programmiersprache ermöglicht, Befehle und Funktionen aus anderen Programmiersprachen zu integrieren
- Eine SQL Engine in Übereinstimmung mit Standards, die ermöglicht, Tabellen und Felder anzulegen sowie Daten in der Datenbank zu verwalten und danach zu suchen
- Das Maintenance und Security-Center ermöglicht die zentrale Überprüfung und Administration der Anwendung

Einsteiger können rasch Datenbanken erstellen und beginnen, ihre Daten zu verwalten. Fortgeschrittene Benutzer können ihre Datenbanken mit den 4D Entwicklungswerkzeugen an eigene Bedürfnisse anpassen. Erfahrene Entwickler können mit Hilfe der leistungsstarken 4D Programmiersprache bzw. SQL die Datenbank mit ausgeklügelten Funktionalitäten ausstatten, wie z.B. Datenübertragung, Kommunikation und Web-Tauglichkeit.

4D arbeitet mit zwei verschiedenen Umgebungen: Der Designumgebung und der Anwendungsumgebung.

Designumgebung

Im Designmodus erstellen, entwickeln und testen Sie Ihre Anwendungen. Hier bestimmen Sie das Aussehen Ihrer Anwendung. Sie erstellen Tabellen und Datenfelder, bestimmen die Verknüpfungen der Tabellen zueinander, erstellen Formulare für die Dateneingabe und -anzeige, sowie den Druck, fügen ein Kennwortsystem ein, erstellen eigene Menüs und fügen Methoden für Datenbankobjekte hinzu.

Sie möchten z.B. laufend Informationen über jeden Angestellten einer Firma erhalten. In der Designumgebung erstellen Sie eine Tabelle [Angestellte] und fügen dieser Tabelle Datenfelder hinzu, in denen Sie die Daten der Angestellten speichern, wie z.B. Name, Tätigkeit, Einstellungsdatum und Gehalt. Sie können auch eine Tabelle [Abteilung] hinzufügen, in der Informationen über jede Abteilung der Firma enthalten sind und dann zwischen diesen Tabellen eine Verknüpfung herstellen. So lässt sich leicht feststellen, in welcher Abteilung ein Angestellter arbeitet oder welche Angestellten zu einer bestimmten Abteilung gehören.

Der Designmodus bietet Zugriff auf verschiedene Fenster und Editoren, über die Sie Standardoperationen ausführen können: Sie können Daten eingeben, nach einem bestimmten Datensatz suchen, Daten im- und exportieren, Berichte erstellen, Etiketten für Mailing erstellen, usw. So können Sie die Funktionsweise Ihrer Anwendung testen, indem Sie z.B. einige Datensätze eingeben bzw. importieren, oder Ihre Methoden ausführen. Die verfügbaren Editoren sind dieselben, mit denen der Benutzer in der Anwendungsumgebung arbeitet.

In der Designumgebung können Sie:

- Tabellen und Datenfelder zum Speichern von Daten erstellen
- Verknüpfungen zwischen Tabellen herstellen
- Formulare für die Eingabe, Anzeige, Drucken und Publizieren von Daten erstellen
- Auswahllisten erstellen, welche die Dateneingabe vereinfachen und steuern
- Methoden schreiben und ausführen, um Operationen in der Anwendung zu automatisieren
- Eigene Menüs erstellen und Methoden oder automatische Standardaktionen zuweisen
- Parallele Prozesse erstellen und verwalten, mit denen Sie mehrere Datenbankoperationen gleichzeitig durchführen können
- Ein Kennwortsystem zum Steuern des Zugriffs auf Daten einrichten
- Einen Web Server oder SQL Server starten, um über externe Anwendungen mit den Daten zu arbeiten
- Web Services aufrufen oder publizieren
- Die Datenbank kompilieren, um ihre Ausführung zu beschleunigen
- Eine Datenbank mit einkompilierter 4D Engine für eine eigenständige Anwendung erstellen
- Ein automatisches Backup-System einrichten, einschließlich Datenreplikation
- Daten eingeben und verändern
- Daten ansehen und drucken
- Datensätze suchen und sortieren
- Berichte, Etiketten und Grafiken erstellen
- Daten im- und exportieren
- Mit allen in der Datenbank installierten Plug-Ins arbeiten

Anwendungsumgebung

Dies ist die Umgebung, die Sie für eine eigene Anwendung verwenden können – eine Anwendung, die 4D benutzt, aber ihr eigenes Menüsystem und Bildschirmdesign hat. Dies ist die Umgebung, in der der Entwickler seine im Designmodus erstellten Anwendungen vertreibt. Um vom Designmodus in den Anwendungsmodus zu wechseln, rufen Sie einfach im Menü **Start** den Befehl **Anwendung testen** auf.

Im Anwendungsmodus steuern Sie alles, von den Menüs und Formularen bis zu den Methoden, die zum Bestätigen, Verarbeiten und Anzeigen von Daten dienen. Sie sind verantwortlich für Menüeinträge und zugeordnete Methoden oder Standardaktionen zum Verwaltung von Basisaufgaben, wie z.B. Dateneingabe und -änderung, Suchen und Sortieren, sowie Erstellen von Berichten. Sie können einige oder alle Standardeditoren von 4D verwenden, z.B. Sortiereditor, Etiketteditor, oder Bildschirmdarstellungen bzw. Editoren selbst gestalten.

Der Anwendungsmodus kann für jede Anwendung, die Sie erzeugen, völlig unterschiedlich sein. Aus der Sicht des Benutzers ist der Anwendungsmodus eine vollständige Anwendung für einen spezifischen Zweck, z.B. zum Verwalten von Kunden und Rechnungen.

Die Standard Menüleiste von 4D enthält das Menü **Modus**, über das Sie in die Designumgebung zurückkehren können.

Haben Sie gleichzeitig die Fenster mehrerer Umgebungen geöffnet, können Sie durch Klicken auf das entsprechende Fenster zwischen den Umgebungen umschalten. Wenn Sie auf ein Fenster klicken, bringt 4D das entsprechende Fenster in den Vordergrund des Bildschirms und aktiviert diese Umgebung.










Integrierte Server

4D hat drei integrierte Server: Einen Daten- und Anwendungsserver (4D Server), einen Web Server und einen SQL Server. Die Verwaltung dieser Server ist im allgemeinen ganz einfach und benötigt nur ein paar Einstellungen. Über Administrationsbefehle lässt sich ihre Ausführung steuern. Beachten Sie, dass Sie zum Aktivieren dieser Server entsprechende Lizenzen benötigen.

Das Starten und Verwalten dieser Server wird in folgenden Handbüchern beschrieben:

- Daten- und Anwendungsserver: Handbuch *4D Programmiersprache* im Abschnitt **4D Server Verwaltungsfenster**
- 4D Web Server: Handbuch *4D Programmiersprache* im Abschnitt **Web Server konfigurieren und Verbindung verwalten**
- 4D SQL Server: Handbuch *4D - SQL Reference* im Abschnitt **Configuration of 4D SQL Server** (nur in Englisch)

4D Dateien verwalten

-  4D starten
-  Lokale/Remote Architektur
-  Neue Anwendung erstellen
-  Lokale Anwendung öffnen
-  Anmelden über Remote Anwendung
-  Plug-Ins oder Komponenten installieren
-  Datendatei ändern
-  Beschreibung der 4D Dateien
-  Konvertierung älterer Anwendungen

4D starten

Wenn Sie Ihre 4D Version nach den Anweisungen im **OBJECT Get pointer** installiert und registriert haben, können Sie das Programm starten.

Markieren Sie das Programm-Icon von 4D und wählen im Menü **Datei/Ablage** den Befehl **Öffnen** oder doppelklicken Sie direkt auf das Programm-Icon:



Auf dem Bildschirm erscheint der Willkommen Dialog:

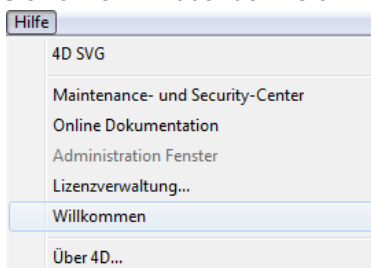


Hier können Sie folgendes ausführen:

- **Anmelden** mit Ihrem 4D Konto. Sind Sie bereits angemeldet, können Sie sich **abmelden**.
- Unter **Zuletzt verwendet** bereits verwendete 4D Anwendungen sehen und öffnen. Klicken Sie auf den jeweiligen Namen, erscheint der komplette Pfad der Anwendung als Tipp:
 - ▼ **Zuletzt verwendet**
 - ☑ Kunden
 - ☑ `M:\C:\Users\Barbara\Desktop\BeispielDB\15\Kunden4dbase`
- Unter **Anwendungen starten** sich an eine 4D Server Anwendung anmelden, eine lokale Anwendung öffnen bzw. neu erstellen
- Zum **4D Store** gehen.

Der Assistent erscheint standardmäßig bei jedem Start. Sie können ihn ausblenden, wenn Sie am unteren Rand die Option **Diesen Dialog beim Starten nicht mehr anzeigen** markieren. In diesem Fall zeigt 4D beim Starten direkt den Öffnen-Dialog bzw. die in den Einstellungen gewählte Option (siehe **Seite Allgemein**).

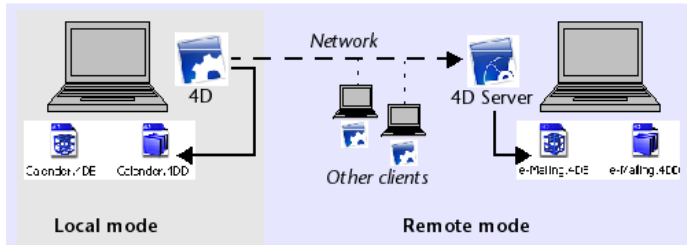
Sie können ihn über den Befehl **Willkommen** im Menü **Hilfe** jederzeit wieder anzeigen:



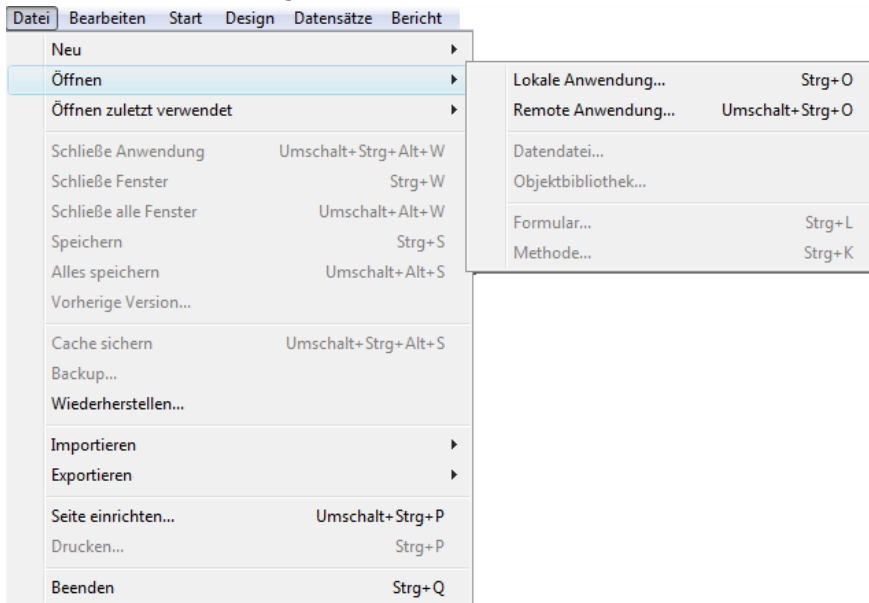
Wenn Sie den Willkommen Assistenten schließen, bleibt das 4D Programm gestartet, aber es ist keine Datenbank geöffnet. Sie können die Datenbank auch über das Maintenance und Security-Center öffnen. Weitere Informationen dazu finden Sie im Abschnitt **Anzeige im Wartungsmodus**. Sie können eine Datenbank auch über das Menü **Datei/Ablage** oder die Werkzeugleiste öffnen bzw. anlegen. Weitere Informationen dazu finden Sie in den Abschnitten **Neue Anwendung erstellen** und **Lokale Anwendung öffnen**.

Lokale/Remote Architektur

Das 4D Programm kann sowohl lokale als auch remote Datenbanken, die über 4D Server ausgeführt werden, öffnen. Im ersten Fall arbeitet die Anwendung im lokalen Modus (Einzelplatz), im zweiten im remote Modus (Client/Server).



Wenn Sie eine Anwendung zum Öffnen auswählen, wählen Sie auch den Modus, in welchem die Anwendung laufen soll:



- **Öffnen > Lokale Anwendung:** Zeigt den Standard-Öffnen Dialog, um eine 4D Strukturdatei auszuwählen. (siehe Abschnitt **Lokale Anwendung öffnen**).
- **Öffnen > Remote Anwendung:** Zeigt den Verbindungsdialog zu 4D Server, um eine veröffentlichte Anwendung auszuwählen. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Anmeldung an eine 4D Server Datenbank**.

Neue Anwendung erstellen

Sie können eine leere Datenbank erstellen oder aus einer zuvor exportierten Strukturdefinition. Es macht keinen Unterschied, ob Sie die Datenbank mit 4D oder 4D Server anlegen.

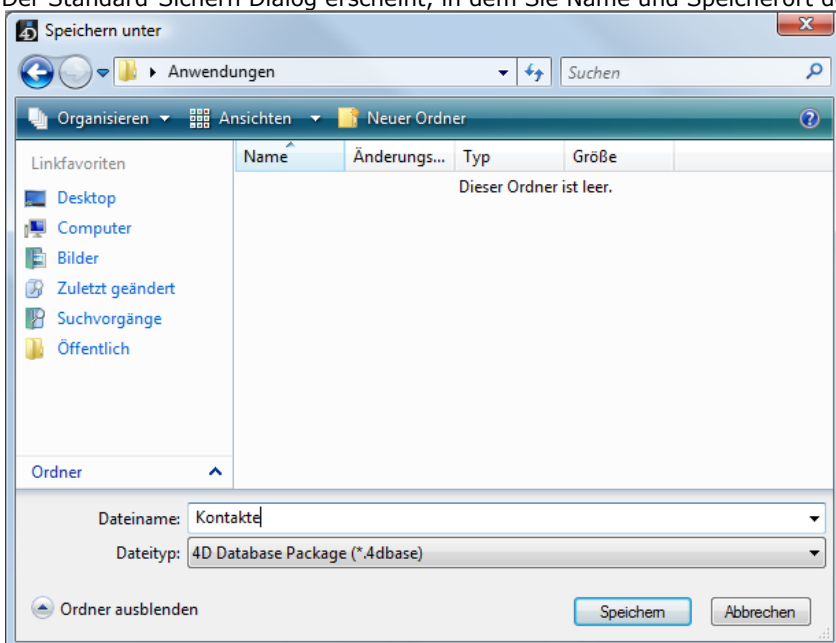
Leere Datenbank erstellen

Sie können eine leere Datenbank erstellen, d.h. sie enthält weder Tabellen, Felder noch Formulare. Das entspricht einer leeren Seite für Datenbanken.

Um eine leere Datenbank zu erstellen:

1. Wählen Sie im Menü **Datei/Ablage** oder über das Icon **Neu** der Werkzeugleiste am oberen Rand den Eintrag **Neu>Struktur**.

Der Standard-Sichern Dialog erscheint, in dem Sie Name und Speicherort der Datenbank wählen können.



2. Geben Sie den Namen Ihrer Datenbank ein und klicken auf die Schaltfläche **Speichern**.

Sie können jeden Dateinamen verwenden, den Ihr Betriebssystem erlaubt.

Achtung: Ist Ihre Datenbank auch zum Arbeiten auf anderen Systemen vorgesehen, müssen Sie die jeweils geltenden Einschränkungen berücksichtigen. Zum Beispiel wird "backslash" (\) auf Mac OS X akzeptiert, ist dagegen unter Windows nicht erlaubt.

Bestätigen Sie das Dialogfenster, schließt 4D - sofern vorhanden - die aktuelle Datenbank, erzeugt im gewählten Verzeichnis einen Ordner mit dem angegebenen Namen (dies ist auch der Name der Datenbank) und legt alle Dateien für eine funktionsfähige Datenbank darin ab. Weitere Informationen zum Aufbau von Datenbanken finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Architektur der Anwendung**.

Auf dem Bildschirm erscheint dann das Fenster der 4D Anwendung mit dem Explorer im Vordergrund. Hier können Sie z.B. Projektformulare anlegen oder den Struktureditor anzeigen und Tabellen, Felder, etc. definieren.

Datenbank über Strukturdefinition erstellen

Mit Strukturdefinitionen, die im XML Format exportiert wurden, können Sie sofort neue identische Datenbanken anlegen. Die Strukturdefinition lässt sich als Strukturvorlage betrachten, die Sie nach Belieben duplizieren können. Weitere Informationen über Import und Export von Strukturdefinitionen finden Sie im Abschnitt **Strukturdefinitionen exportieren und importieren**.

Sie können die XML Strukturdefinition ohne Änderung verwenden oder zuerst über einen XML Editor bearbeiten, d.h. Sie können jede Art von Mechanismus zum Erzeugen von Strukturen per Programmierung in Betracht ziehen.

Da das interne Format der XML Dateien der 4D Strukturdefinition öffentlich ist (siehe **Format einer 4D Strukturdefinition**), können Sie diese Datei auch aus anderen Datenbankumgebungen oder in einem beliebigen Design-Programm erstellen, um daraus automatisch 4D Datenbanken zu generieren.

Um eine Datenbank aus einer Strukturdefinition zu erstellen:

1. Wählen Sie im Menü **Datei/Ablage** oder über das Icon **Neu** der Werkzeugleiste am oberen Rand den Eintrag **Neu>Struktur von Strukturdefinition**. Der Standard-Öffnen Dialog erscheint, in dem Sie die Strukturdefinition wählen können. Sie müssen ein XML Format wählen, das die „Grammatik“ der 4D Strukturdefinition berücksichtigt (das Programm bestätigt die Datei via DTD).

2. Wählen Sie eine XML Datei mit der Strukturdefinition und klicken dann auf die Schaltfläche **OK**.
Auf dem Bildschirm erscheint das Dialogfenster, um Name und Speicherort der neuen Datenbank zu definieren.
3. Legen Sie Name und Speicherort für die neue Datenbank fest und klicken auf die Schaltfläche **Speichern**.

Bei gültiger XML Datei schließt 4D - sofern vorhanden - die aktuelle Datenbank und legt anhand der Strukturdefinitionsdatei eine neue Struktur an. Im gewählten Verzeichnis wird ein Ordner mit dem angegebenen Namen (dies ist auch der Name der Datenbank) erzeugt, der alle Dateien für eine funktionsfähige Datenbank enthält. Der **Explorer** erscheint dann im Vordergrund.

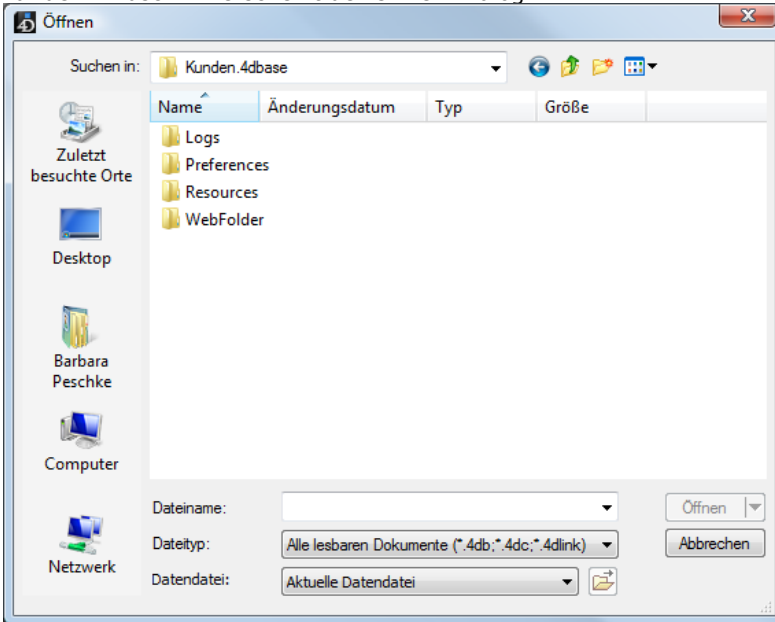
Lokale Anwendung öffnen

Über Öffnen-Dialog

Im Standard-Öffnen Dialog können Sie eine beliebige 4D Struktur auswählen und verschiedene Optionen einstellen. Gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü **Datei/Ablage** oder über das Icon **Öffnen** der Werkzeugleiste den Befehl **Öffnen>Lokale Anwendung**.

Auf dem Bildschirm erscheint der Öffnen-Dialog:



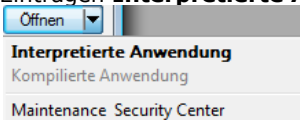
2. Wählen Sie die gewünschte Anwendung
 - Unter Windows wählen Sie die Strukturdatei aus. Sie hat die Endung .4DB oder .4DC.
 - Auf Mac OS können Sie entweder das Datenbank Paket (Endung .4dbase) oder die Strukturdatei (Endung .4DB oder .4DC) auswählen. Weitere Informationen zur Architektur des Pakets finden Sie im Abschnitt **Ordner .4dbase**.
 - Sie können auch eine Zugriffsdatei vom Typ .4DLink auswählen. 4DLink Dateien speichern die Parameter, die zum Öffnen der Datenbank benötigt werden, wie Adresse, Zugangsdaten u.ä. Sie funktionieren mit lokalen und remote Anwendungen. Weitere Informationen zum Einrichten und Verwenden von 4DLink Dateien finden Sie im *4D Server Handbuch* im Abschnitt **Anmeldung an eine 4D Server Datenbank**.
3. Klicken Sie am unteren Rand in der Schaltfläche **Öffnen** auf den Pfeil für weitere Optionen (siehe nächster Abschnitt) oder klicken Sie auf **Öffnen**, um die Anwendung zu öffnen.

Hinweis: Sie können das 4D Programm auch direkt per Drag&Drop oder per Doppelklick auf das .4dbase Paket, die Strukturdatei .4DB oder .4DC, bzw. die .4DLink Datei öffnen.

Optionen des Öffnen-Dialogs


Zusätzlich zu den Standardoptionen des Systems bietet der Öffnen-Dialog in 4D über die Schaltfläche **Öffnen** und das DropDown-Menü **Datendatei** weitere Optionen.

- **Öffnen:** Klicken Sie in der Schaltfläche am rechten Rand auf den Pfeil, öffnet sich ein DropDown-Menü mit den Einträgen **Interpretierte Anwendung/Kompilierte Anwendung** und **Maintenance Security Center**:



- **Interpretierte Anwendung/Kompilierte Anwendung.** Beide Modi erscheinen nur, wenn die gewählte Anwendung interpretierten und kompilierten Code enthält. Ist ein Modus nicht verfügbar, erscheint die entsprechende Option als nicht auswählbar. Das ist z.B. der Fall, wenn die Anwendung nicht kompiliert bzw. nach Änderungen nicht erneut kompiliert wurde oder nur die kompilierte Datei verfügbar ist.
- **Maintenance Security Center:** Über diese Option können Sie auf beschädigte Anwendungen zugreifen und die erforderlichen Reparaturen durchführen. Weitere Informationen dazu finden Sie im Abschnitt **Anzeige im Wartungsmodus**.

Klicken Sie direkt auf die Schaltfläche **Öffnen**, wird die Anwendung im zuletzt verwendeten Modus geöffnet.

- **Datendatei:** Hier legen Sie fest, welche Datendatei mit der Anwendung verwendet werden soll. Standardmäßig ist **Aktuelle Datendatei** markiert. Die Anwendung öffnet sich mit dieser Datei, wenn Sie auf die Schaltfläche **Öffnen** klicken. Über die Schaltfläche  können Sie den Speicherort dieser Datei auf Ihrer Festplatte in einem Systemfenster anzeigen.

Dieses Menü enthält zwei weitere Optionen, da Sie mit 4D die gleiche Strukturdatei mit verschiedenen Datendateien verwenden können.

- **Andere Datendatei öffnen:** Damit öffnen Sie die Anwendung mit einer anderen bereits vorhandenen Datendatei. Klicken Sie auf **Öffnen**, erscheint der Standard-Öffnen Dialog, in dem Sie die gewünschte Datendatei angeben können.
- **Neue Datendatei anlegen:** Damit legen Sie eine leere Datendatei für die Anwendung an. Wählen Sie diese Option und klicken dann auf **Öffnen**, erscheint der Standard-Sichern Dialog, in dem Sie Name und Speicherort der neuen Datendatei definieren können.

Die hier festgelegte Datei wird dann standardmäßig beim nächsten Öffnen ausgewählt, außer 4D findet auf der Ebene der Strukturdatei eine Datei mit der Endung .4DD, die denselben Namen wie die Strukturdatei hat. Wenn Sie die Datendatei umbenennen oder verschieben, müssen Sie sie erneut auswählen.

Sie können die Datendatei beim Starten der Datenbank auch direkt ändern. Weitere Informationen dazu finden Sie im Abschnitt **Datendatei ändern**.

Sie können jederzeit prüfen, welches die aktuelle Datendatei ist. Gehen Sie dazu im Maintenance und Security-Center auf die **Seite Information**.

Anwendung direkt öffnen

4D bietet verschiedene Funktionen, um Anwendungen direkt zu öffnen: Drag&Drop, den Menübefehl „Zuletzt verwendet“ und automatisches Öffnen der zuletzt verwendeten Anwendung.

Um eine Anwendung per Drag&Drop zu öffnen:

1. Markieren Sie die entsprechende Datendatei und ziehen sie auf das 4D Programm-Icon.
 - Unter Windows müssen Sie die Strukturdatei (Endung .4DB oder .4DC) auswählen.
 - Auf Mac OS können Sie entweder das Datenbank Paket (Endung .4dbase) oder die Strukturdatei (Endung .4DB oder .4DC) auswählen. Weitere Informationen zum Datenbank Paket finden Sie im Abschnitt **Ordner .4dbase**.
 - Sie können auf beiden Plattformen eine .4DLink Zugriffsdatei auswählen.

Um die zuletzt verwendete Anwendung zu öffnen:

1. Wählen Sie im Menü **Datei/Ablage** oder über das Icon **Öffnen** der Werkzeugleiste den Befehl **Öffnen>Zuletzt verwendet**.

Das DropDown-Menü ist beim ersten Start leer. Es füllt sich schrittweise, wenn verschiedene lokale oder remote Anwendungen verwendet werden; 4D speichert automatisch Namen und Speicherort aller geöffneten Anwendungen. Über den Menübefehl **Lösche Menü** können Sie alle Anwendungsnamen aus dem DropDown-Menü entfernen.

Um automatisch die zuletzt verwendete Anwendung zu öffnen:

1. Gehen Sie in den 4D Einstellungen auf die Seite "Öffnen".
Weitere Informationen dazu finden Sie im Abschnitt **Zugriff**.
2. Wählen Sie im Bereich "Optionen" im DropDown-Menü "Beim Starten" den Eintrag **Zuletzt verwendete Anwendung öffnen**.

4D öffnet dann beim Starten immer die zuletzt verwendete Anwendung.

Hinweis: Soll bei dieser Option der Öffnen-Dialog angezeigt werden, drücken Sie beim Öffnen der Anwendung unter Windows die **Alt-Taste**, auf macOS die **Wahltaste**.

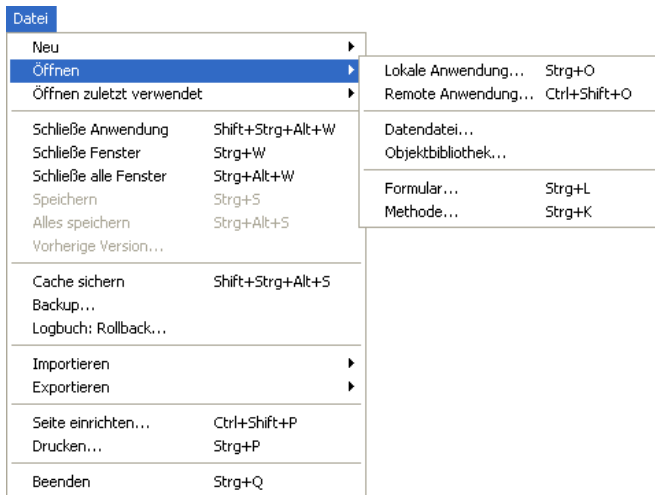
Anmelden über Remote Anwendung

Es gibt drei Wege, sich über remote 4D an eine 4D Server Datenbank anzumelden:

- Über den Anmeldedialog
- Über den Befehl **Öffnen zuletzt verwendet**
- Über eine Verknüpfungsdatei 4D Link mit den Zugriffsparametern für die Datenbank.

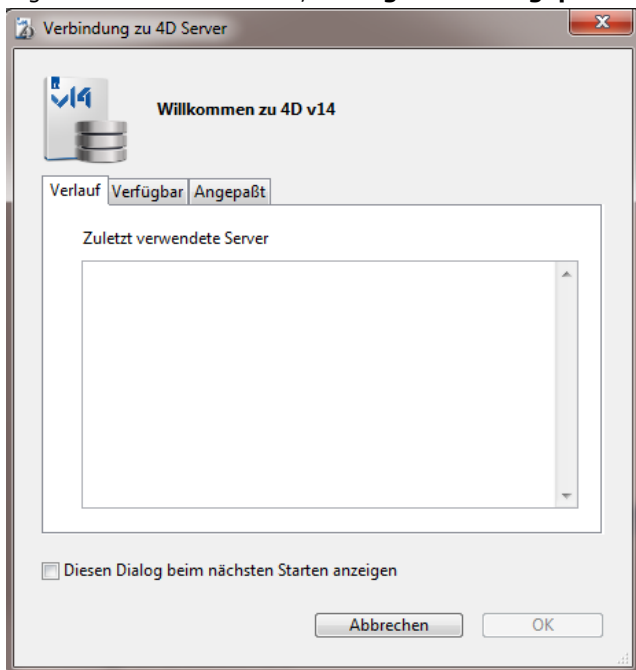
Über den Anmeldedialog

Um den Anmeldedialog für 4D Server anzuzeigen, starten Sie zuerst die 4D Anwendung. Über das Menü **Datei/Ablage>Öffnen** bzw. die entsprechende Schaltfläche in der 4D Toolbar können Sie die Art zum Öffnen der 4D Anwendung auswählen:



Wählen Sie den Befehl **Öffnen>Remote Anwendung**.

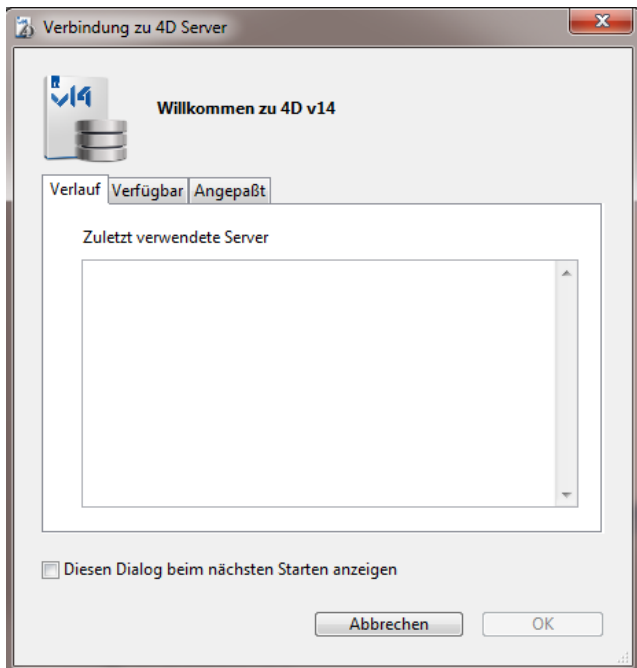
Auf dem Bildschirm erscheint der Anmeldedialog für 4D Server. Er enthält 3 Registerkarten, über die Sie auf folgende Seiten zugreifen können: **Verlauf**, **Verfügbar** und **Angepaßt**.



Markieren Sie die Option **Diesen Dialog beim nächsten Starten anzeigen**, erscheint das Fenster automatisch beim Starten der 4D Anwendung.

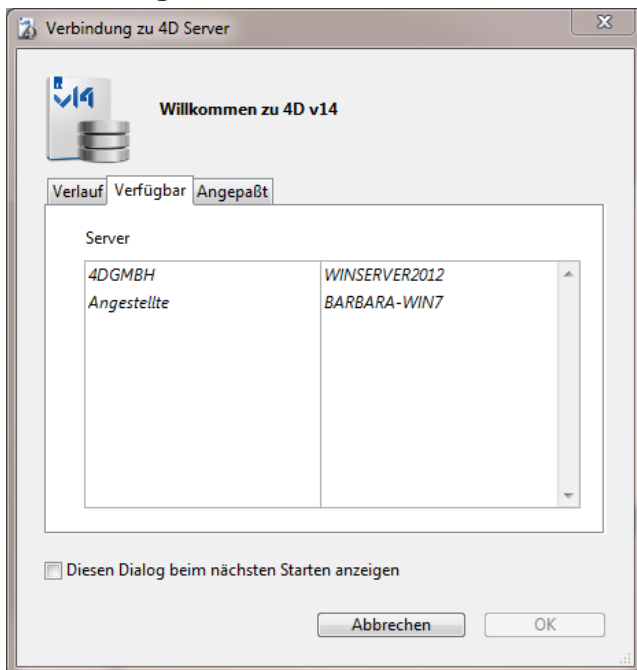
Hinweis: Sie können dieses Dialogfenster auch öffnen, wenn Sie im 4D Willkommen-Assistent auf den Link **Anmelden an 4D Server** klicken.

Seite Verlauf



Die Seite **Verlauf** zeigt die Liste der zuletzt verwendeten 4D Server in alphabetischer Reihenfolge. Um sich an einen Server in der Liste anzumelden, doppelklicken Sie auf seinen Namen oder markieren ihn und klicken auf die Schaltfläche **OK**.

Seite Verfügbar



4D Server enthält ein integriertes TCP/IP System, das standardmäßig die Namen der 4D Server Datenbanken veröffentlicht, die über das Netzwerk verfügbar sind. Die Namen erscheinen im Anmeldedialog auf der Seite **Verfügbar**.

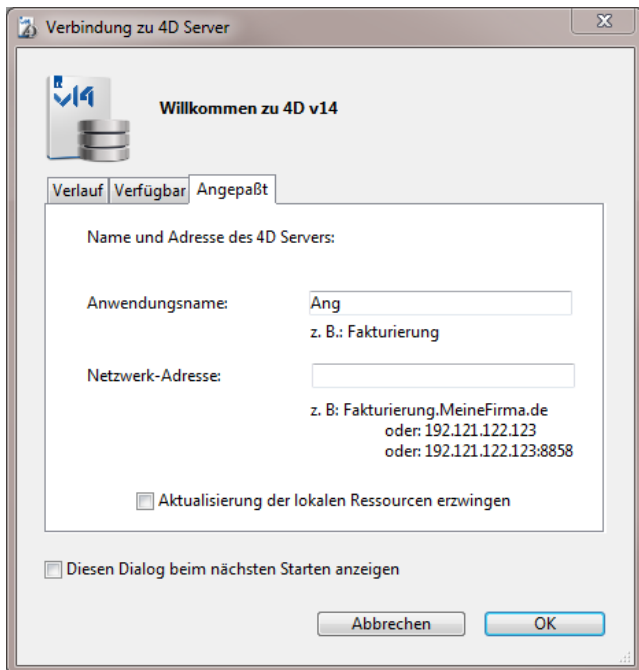
Die Liste ist in der Reihenfolge der Veröffentlichung sortiert und wird dynamisch aktualisiert. Wollen Sie sich über diese Liste an einen Server anmelden, doppelklicken Sie auf seinen Namen oder markieren ihn und klicken auf die Schaltfläche **OK**.

Hinweise:

- Ein Circumflex (^) vor dem Namen der Datenbanken gibt an, dass sie verschlüsselt publiziert werden. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Client/Server Verbindungen verschlüsseln**.
- Es ist möglich, die dynamische Veröffentlichung des Datenbanknamen im Netzwerk zu unterbinden. In diesem Fall lässt sich die Verbindung nur manuell über die Seite "Angepasst" herstellen. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **IP Einstellungen**.

Seite Angepasst

Auf dieser Seite können Sie einen im Netzwerk veröffentlichten Server über seine IP Adresse zuweisen und einen eigenen Namen vergeben:



Sie können das TCP/IP Übertragungssystem von 4D Server so anpassen, dass die Namen der Datenbanken nicht automatisch über das Netzwerk publiziert werden (siehe Abschnitt **IP Einstellungen**). In diesem Fall erscheinen die Namen nicht auf der Seite **Verfügbar**. Kennen Sie jedoch die IP-Adresse einer nicht angezeigten Server-Datenbank, können Sie die Adresse manuell eingeben.

- **Anwendungsname:** Hier definieren Sie den Namen der 4D Server Datenbank. Dieser Name wird auf der Seite **Verlauf** verwendet, wenn Sie Bezug auf die Datenbank nehmen.
- **Netzwerk-Adresse:** Hier geben Sie die IP-Adresse des Rechners an, auf dem der 4D Server gestartet wurde. Laufen auf einem Rechner zwei Server gleichzeitig, müssen an die IP Adresse ein Doppelpunkt und eine Port-Nummer angehängt werden, zum Beispiel 192.168.92.104:19814. Der Port zum Veröffentlichen eines Servers ist standardmäßig 19813. Diese Nummer lässt sich in den Einstellungen der Anwendung verändern. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Netzwerk und Client-Server Optionen**.

Hinweis: Haben Sie auf den Seiten **Verlauf** oder **Verfügbar** eine Datenbank ausgewählt und klicken auf die Registerkarte **Angepasst**, enthalten die beiden Felder die entsprechende Information.

Sobald diese Seite einen Server zuweist, können Sie auf die Schaltfläche **OK** klicken, um sich an den Server anzumelden. Der Server erscheint dann auf der Seite **Verlauf**.

Hinweis: Wird die Datenbank verschlüsselt veröffentlicht, müssen Sie vor den Namen ein Circumflex (^) setzen. Andernfalls wird die Anmeldung zurückgewiesen. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Client/Server Verbindungen verschlüsseln**.

Aktualisieren der lokalen Ressourcen erzwingen

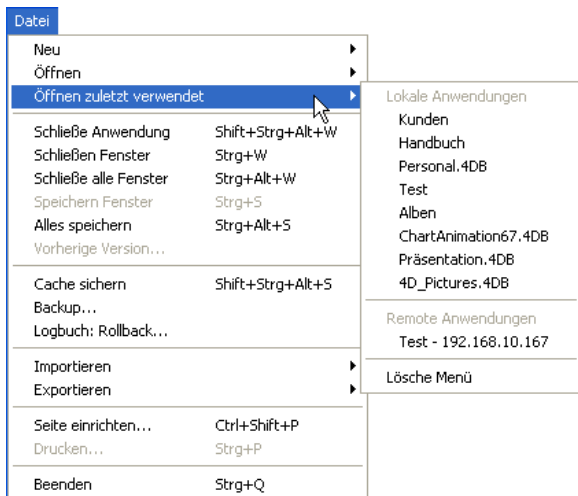
Ist diese Option markiert, werden die lokalen Ressourcen auf dem Client-Rechner beim Anmelden systematisch aktualisiert. Das sind Informationen zur Struktur der Datenbank, die auf jedem Client-Rechner gespeichert sind.

In der Regel werden die lokalen Ressourcen auf dem remote Rechner automatisch bei jeder Anmeldung aktualisiert, wenn die Struktur der Datenbank geändert wurde. In den meisten Fällen ist diese Option nicht erforderlich. In bestimmten Fällen kann es jedoch erforderlich sein, die Aktualisierung zu erzwingen.

Über den Befehl Öffnen zuletzt verwendet

Über das Menü **Datei/Ablage>Öffnen zuletzt verwendet** können Sie sich direkt an eine 4D Server Datenbank anmelden, die sie zuvor schon verwendet haben.

Verwenden Sie die 4D Anwendung zum Öffnen lokaler und zum Anmelden an remote Anwendungen, zeigt das Untermenü beide Arten an. Die remote Anwendungen erscheinen im unteren Teil:



Die IP-Adresse des Server erscheint neben dem Namen der Anwendung. Über den Befehl **Lösche Menü** können Sie die Einträge entfernen.

Über die Verknüpfungsdatei 4DLink

Sie können Zugriffsdateien für Datenbanken anlegen. Sie enthalten Parameter, um das Öffnen oder Anmelden an 4D Datenbanken zu automatisieren und zu vereinfachen. Eine Zugriffsdatei kann in der Regel die Adresse einer remote Anwendung sowie die Anmelde-Identifikation sichern, wodurch für den Benutzer mehrere Operationen wegfallen. Über diese Zugriffsdateien können Sie entweder eine lokale Datenbank öffnen oder sich an eine von 4D Server veröffentlichte Datenbank anmelden.

Zugriffsdatei einrichten

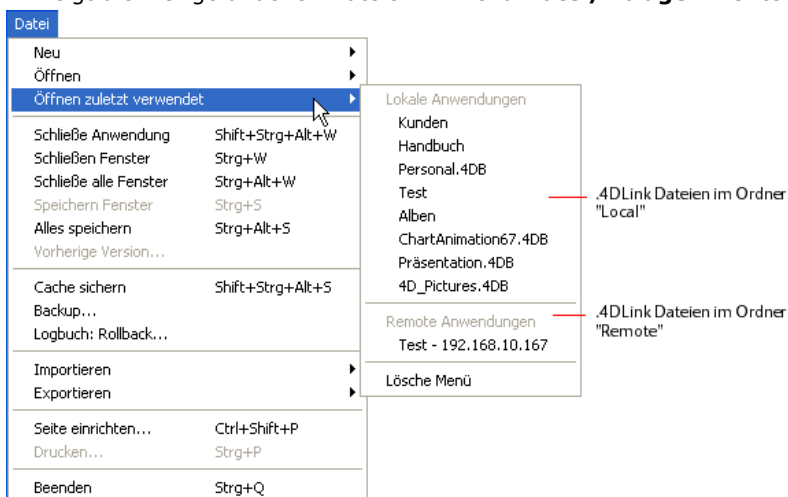
Die Zugriffsdateien von 4D Anwendungen sind XML Dateien mit der Endung ".4DLink". 4D erstellt und verwendet diesen Dateityp zum Einrichten des Untermenüs **Zuletzt verwendet**: 4D erstellt automatisch die Datei .4DLink, wenn eine Datenbank zum ersten Mal auf dem Rechner geöffnet wird oder beim ersten Anmelden an den Server. Die automatisch angelegten Dateien .4DLink werden im lokalen Ordner Preferences des Benutzers abgelegt. Es werden zwei Verzeichnisse angelegt: Local und Remote. Der Ordner Local enthält die Dateien ".4DLink" zum Anmelden an Datenbanken auf dem Rechner, der Ordner Remote die Dateien ".4DLink" zum Anmelden an remote Datenbanken.

Die lokalen Ordner Preferences liegen an folgender Stelle:

- Windows 7 und höher: C:\Users\UserName\AppData\Roaming\4D\Favorites vXX\
- Mac OS: Users/UserName/Library/Preferences/4D/Favorites vXX

... XX ist die Versionsnummer des Programms (zum Beispiel "Favorites v14" für 4D v14).

4D zeigt die hier gefundenen Dateien im Menü **Datei/Ablage** im Untermenü **Öffnen zuletzt verwendet** an:



Die Dateien ".4DLink" lassen sich auch mit einem XML Editor erstellen. Sie können auch angepasste Information, wie die Anmelde-Identifikation oder den Öffnungsmodus der Datenbank enthalten. 4D liefert ein DTD mit der Beschreibung der XML Keys, mit der Sie eine ".4DLink" Datei einrichten können. Die DTD lautet database_link.dtd und liegt im Unterordner `\Resources\DTD\` des 4D Programms.

Dateien verwenden

Über eine Zugriffsdatei .4DLink können Sie das 4D Programm starten und die 4D Zieldatenbank öffnen. Es gibt folgende Möglichkeiten:

- Doppelklick oder Drag&Drop auf das 4D Programm,
- Über das Untermenü **Zuletzt verwendet** (Datei im lokalen Ordner Preferences).
Eine Datei .4DLink vom Typ "remote Anwendung" können Sie kopieren und auf verschiedenen Rechnern verwenden.

Hinweis: Sie können eine Datei .4DLink auch im Öffnen-Dialog von 4D und 4D Server auswählen (nur beim Öffnen lokaler Anwendungen).

Plug-Ins oder Komponenten installieren

Plug-Ins und Komponenten rüsten Ihre Anwendungen mit zusätzlichen Funktionalitäten aus.

- Plug-Ins sind externe Programme, die Zugriff auf verschiedene Funktionalitäten bieten. So fügt 4D Write die Funktionen der Textverarbeitung hinzu, z.B. um Serienbriefe zu erstellen. Manche Plug-Ins wie 4D Internet Commands sind kostenlos verfügbar, für andere müssen Sie eine Lizenz erwerben. Plug-Ins können von 4D Mitarbeitern oder Drittherstellern entwickelt worden sein.
- Komponenten sind unabhängige Programme, die mit 4D entwickelt wurden und weitere high-level Funktionalitäten bieten. So ergänzt z.B. die 4D SVG Komponente die standardmäßigen Möglichkeiten von 4D beim Arbeiten mit SVG.

Plug-Ins und Komponenten lassen sich in 4D und 4D Server einsetzen. Damit sie innerhalb Ihrer Anwendungen verwendbar sind, müssen sie an der richtigen Stelle installiert werden. Je nach Verwendungszweck und Betriebssystem sind verschiedene Stellen möglich.

Hinweis: Weitere Informationen zum Entwickeln und Vertreiben von Komponenten finden Sie im Kapitel **4D Komponenten entwickeln und installieren**.

Installation

Um Plug-Ins oder Komponenten in der 4D Umgebung zu installieren, kopieren Sie die Dateien in die passenden Ordner: Plug-Ins in den Ordner mit Namen **Plugins**, Komponenten in den Ordner mit Namen **Components**. Diese Ordner können je nach Ihren Bedürfnissen an zwei verschiedenen Stellen liegen (siehe nächster Abschnitt).

Hier sehen Sie die Elemente für die Ordner:

- Das Bundle *PluginName.bundle* (auf Mac OS Paket) enthält die Windows- und Mac-Version der Plug-Ins. Ihre spezifische interne Architektur ermöglicht 4D Server, je nach der Plattform des Client-Rechners, die passende Version zu laden. Um ein Plug-In in Ihrer Umgebung zu installieren, legen Sie den Ordner *PluginName.bundle* oder das entsprechende Paket in den Ordner **Plugins**.
- Komponenten sind Matrix Datenbanken, deren Dateien die Endungen *.4db* (interpretiert), *.4dc* (kompiliert) oder *.4dbase* (Paket vom Typ Matrix Datenbank, siehe Abschnitt **Ordner .4dbase**). Sie können Aliasse (Mac OS) oder Tastenkürzel (Windows) auf diese Matrix Datenbanken verwenden. Der Ordner **Components** kann alle eigenen Dateien oder Ordner enthalten, die für die Operationen der Komponente erforderlich sind (xliff, Bilder, etc.). Er kann dagegen keine Plug-Ins oder Unterordner von Komponenten oder Datendateien bzw. Benutzer Strukturdateien (*.4DA*) enthalten. Sind solche Einträge vorhanden, werden sie von 4D ignoriert. Weitere Informationen dazu finden Sie im Abschnitt **Überblick**.

4D lädt die Plug-Ins und Komponenten beim Starten des Programms. Beenden Sie deshalb 4D, bevor Sie diese installieren. Öffnen Sie dann Ihre Datenbank mit 4D. Erfordert ein Plug-In eine spezifische Benutzerlizenz, wird es geladen, jedoch nicht aktiviert.

Position der Ordner PlugIns und Components

Sie können die Ordner **PlugIns** und **Components** an zwei verschiedene Stellen legen:

- **Auf dieselbe Ebene wie das ausführbare 4D Programm**, z.B.:
 - Unter Windows neben die *.exe* Datei
 - Auf Mac OS auf die oberste Ebene des Ordners **Contents** im Software-Paket.In diesem Fall sind die Plug-Ins und Komponenten in jeder Anwendung verfügbar, die über dieses 4D Programm geöffnet wird.
- **Auf dieselbe Ebene wie die Strukturdatei der Anwendung**.
In diesem Fall sind die Plug-Ins und Komponenten nur in dieser spezifischen Anwendung verfügbar.

Ihre Wahl richtet sich danach, wie Sie das Plug-In oder die Komponente einsetzen wollen.

Liegt dasselbe Plug-In bzw. die Komponente an beiden Stellen, lädt 4D nur das Plug-In bzw. die Komponente, welches neben der Struktur liegt. Existieren in einer Anwendung mit einkompilierter 4D Volume Desktop mehrere Instanzen desselben Plug-In bzw. derselben Komponente, lässt sich die Anwendung nicht öffnen.

Datendatei ändern

Sie können bei Bedarf die Datendatei verändern, denn Sie können mit derselben Struktur verschiedene Datendateien verwenden.

Datendatei und Strukturdatei müssen zueinander passen. Zur Wahrung der Datenintegrität erlaubt 4D standardmäßig nicht, eine Datendatei zu öffnen, wenn sie nicht mit der aktuellen Strukturdatei erstellt wurde. Das Programm weist der Datendatei und der Strukturdatei beim Erstellen oder Konvertieren automatisch interne Link-Nummern (UUID) zu. Diese Nummern werden beim Öffnen der Datendatei überprüft.

Um die Datendatei zu wechseln, können Sie folgendes ausführen:

- Wählen Sie beim Öffnen der Datenbank im Öffnen-Dialog am unteren Rand im DropDown-Menü **Datendatei** die Option **Andere Datendatei wählen** oder **Neue Datendatei erzeugen**. Sie können eine vorhandene Datei wählen oder eine neue anlegen. Weitere Informationen dazu finden Sie im Abschnitt **Lokale Anwendung öffnen**.
- Drücken Sie beim Starten der Struktur unter Windows die **Alt-Taste**, auf Macintosh die **Wahltaste**. Auf dem Bildschirm erscheint folgendes Dialogfenster:



Wählen Sie **Eine andere Datendatei auswählen** oder **Eine neue Datendatei erzeugen** und klicken auf die Schaltfläche **Weiter**.

- Ist die Datenbank geöffnet, wählen Sie im Menü **Datei/Ablage** oder über das Icon **Öffnen** der Werkzeugleiste den Befehl **Neu>Datendatei** oder **Öffnen>Datendatei**.

In allen Fällen können Sie eine vorhandene Datendatei öffnen oder eine neue erstellen. Bei einer neuen Datendatei öffnet 4D die Datenbank mit der ursprünglichen Struktur, jedoch ohne Datensätze.

Verwenden Sie eine andere Datendatei oder erstellen eine neue, wird diese die neue aktuelle Datendatei, die beim nächsten Start der Datenbank automatisch verwendet wird.

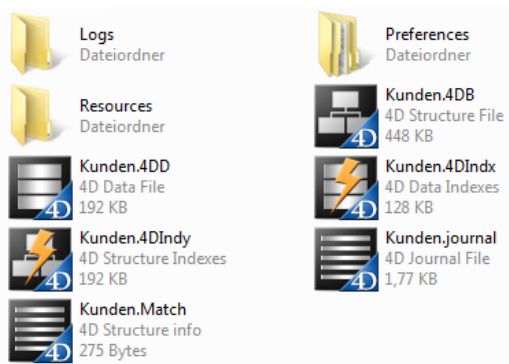
Wenn Sie die Datendatei verschieben oder umbenennen, müssen Sie diese anschließend manuell ablegen.

4D erstellt für jede Anwendung standardmäßig verschiedene Dateien und Ordner. Bei konvertierten Datenbanken aus einer älteren Version gibt es darüberhinaus noch zusätzliche Dateien.

Architektur der Anwendung

Beim Erstellen einer Anwendung werden auf der Festplatte standardmäßig folgende Dateien und Ordner angelegt:

- **Strukturdatei** (Endung ".4DB")
Die Strukturdatei enthält sämtliche Festlegungen für die Struktur der Datenbank, d.h. Tabellen, Datenfelder, Feldeigenschaften, Formulare, Methoden, Menüs, Kennwörter und Auswahllisten. Sie erhält den für die Anwendung festgelegten Namen mit der Endung ".4DB".
- **Datendatei** (Endung ".4DD")
Die Datendatei enthält die Daten, die in den Datensätzen eingegeben werden und alle dazugehörigen Daten. Sie erhält den für die Anwendung festgelegten Namen mit der Endung ".4DD".
Wenn Sie eine 4D-Strukturdatei öffnen, öffnet die Anwendung standardmäßig die aktuelle Datendatei. Verändern Sie den Namen oder Speicherort dieser Datei, erscheint der Öffnen-Dialog, in dem Sie die gewünschte Datendatei auswählen oder eine neue anlegen können. Weitere Informationen dazu finden Sie im Abschnitt **Datendatei ändern**.
Die Datendatei wird automatisch der Strukturdatei zugeordnet, die sie erstellt hat, um eine inkorrekte Verwaltung auszuschließen.
- **Struktur-Indexdatei** (Endung ".4DIndy") und **Daten-Indexdatei** (Endung ".4DIndx")
In einer 4D Anwendung erstellte Indizes werden als eigene Dateien gespeichert und automatisch auf derselben Ebene wie die Strukturdatei abgelegt. Sie dürfen weder verschoben noch umbenannt werden, da 4D sie sonst erneut anlegen muss.
AnwendungName.4DIndx enthält die Indizes der Daten, *Anwendung.4DIndy* die Indizes der Struktur (besonders beim Suchen in der Struktur). Ein Hauptvorteil ist, dass sich die Datei bei einem defekten Index manuell entfernen lässt und bei Neustart automatisch eine neue Indexdatei angelegt wird.
- **Logbuch der Daten** (Endung ".journal") — Diese Datei wird nur angelegt, wenn die Anwendung ein Logbuch verwendet.
Das Logbuch dient zur Sicherung der Daten zwischen zwei Backups. Es schreibt alle Operationen in den Daten der Anwendung mit. Jede ausgeführte Operation löst also zwei simultane Operationen aus: Einmal in den Daten der Anwendung (die Anweisung wird ausgeführt), einmal im Logbuch (die Beschreibung der Operation wird gespeichert).
Das Logbuch wird unabhängig aufgebaut und beeinträchtigt nicht die Arbeit des Benutzers. Eine Anwendung kann gleichzeitig immer nur ein Logbuch führen.
Das Logbuch speichert Operationen wie Hinzufügen, Ändern oder Löschen von Datensätzen, Transaktionen, etc. Es wird standardmäßig erzeugt, wenn eine Anwendung angelegt wird. Weitere Informationen dazu finden Sie im Abschnitt **Logbuch verwalten**.
- **Ordner Resources** mit externen Ressourcen.
Dieser Ordner muss auf derselben Ebene wie die Strukturdatei (.4DB oder .4DC) der Anwendung liegen. Er enthält alle Dateien, die für die Oberfläche benötigt werden. Hier können Sie auch Dateien für die Lokalisierung oder Anpassung der Anwendungsoberfläche ablegen, z.B. Bilder, Text- oder XLIFF-Dateien. 4D nutzt den Inhalt dieser Ordner über automatische Mechanismen, insbesondere für XLIFF Dateien (siehe **PICTURE TO BLOB**) und Bilder (siehe Abschnitt **Statische Bilder verwenden**).
Bei Verwenden im remote Modus ermöglicht der Ordner **Resources** Dateien zwischen dem Server-Rechner und allen Client-Rechnern gemeinsam zu nutzen. Weitere Informationen dazu finden Sie im *Handbuch 4D Server* im Abschnitt **Ordner Resources verwalten**.
- **Ordner Preferences** mit speziellen Konfigurationsdateien für die Datenbank - dieser Ordner wird nur bei Bedarf angelegt.
Die Konfigurationsdateien werden in der Regel im XML Format gespeichert. Verwenden Sie die Backup-Funktion (Datei *Backup.xml*) oder die Applikation-Builder Funktion (Datei *BuildApp.xml*), wird der Ordner **Preferences** automatisch angelegt.
- **Ordner Logs** mit den Logbüchern der aktuellen Anwendung. Dieser Ordner liegt auf derselben Ebene wie die Strukturdatei. Er enthält folgende Logbücher:
 - Konvertierung der Anwendung,
 - Web Server Anfragen,
 - Datenüberprüfung und Reparatur,
 - Strukturüberprüfung und Reparatur,
 - Journal für Backup/Wiederherstellen Aktivitäten,
 - Debugging Befehle,
 - 4D Server Anfragen (auf Client-Rechnern und dem Server generiert).Diese Dateien werden beim Erstellen der Datenbank im selben Verzeichnis abgelegt. Sie sollten ihre Position nicht verändern, denn das beeinträchtigt die automatischen Abläufe in 4D.



Warnung: Auf macOS werden Dateien der Datenbank nicht von 4D oder dem System gesperrt, selbst wenn sie gerade vom Programm genutzt werden. Damit es nicht zu Beschädigungen der Datenbank kommt, achten Sie darauf, dass Dateien in Gebrauch weder umbenannt, verschoben oder gelöscht werden.

Zusätzliche Dateien (konvertierte Anwendungen)

Verwenden Sie eine Anwendung, die mit einer früheren 4D Version erstellt wurde, gibt es weitere Dateien:

- **Ressourcendatei der Struktur** (Endung ".RSR")
Sie enthält die „früheren“ Ressourcen vom Typ Macintosh, die zur Struktur der Anwendung gehören. Die Dateien .4DB und .RSR müssen immer den gleichen Namen haben und sich im gleichen Verzeichnis befinden. Andernfalls können Sie die Anwendung nicht öffnen.
- **Ressourcendatei der Daten** (Endung ".4DR")
Diese Datei enthält die „früheren“ Ressourcen, die zu den Daten der Anwendung gehören.

Weitere Informationen zum Konvertieren älterer Datenbanken finden Sie im Abschnitt [Konvertierung älterer Anwendungen](#).

Ordner .4dbase

4D Anwendungen werden standardmäßig in einem Ordner mit der Endung **.4dbase** erstellt. Lautet Ihre Anwendung z.B. „Kunden“, wird sie im Ordner [Kunden.4dbase] angelegt. Dieser Ordner speichert alle Elemente, die zur korrekten Funktionsweise der Anwendung erforderlich sind.

Aus diesem Grund erscheinen Anwendungsordner auf Mac OS als Paket. Durch Doppelklick auf das Paket können Sie direkt 4D, die Datenbank und die aktuelle Datendatei starten. Sie können das Paket auch per Drag&Drop auf das 4D Programm-Icon ziehen.

Unter Windows hat diese Änderung keine Auswirkung.

Sie können dieses Standardverhalten deaktivieren. Dazu deaktivieren Sie in den 4D Einstellungen auf der [Seite Allgemein](#) die Option **Für neue Anwendungen Paket erstellen**.

Konvertierung älterer Anwendungen

Anwendungen, die mit vorherigen Versionen von 4D bzw. 4D Server erstellt wurden, sind kompatibel mit 4D Version 17 (Struktur- und Datendateien).

Hinweise:

- Datendateien der Version 15, die mit 4D v16 geöffnet wurden, lassen sich weiterhin in ihrer ursprünglichen Version öffnen.
- Sie können jede Strukturdatei im interpretieren Modus konvertieren. Enthält die Datei kompilierten Code, müssen Sie die Anwendung nach dem Konvertieren erneut kompilieren.

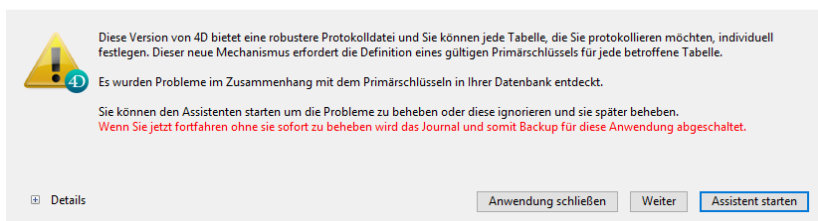
Versionen	Kommentare
15 & 16	Anwendungen werden direkt beim Öffnen mit 4D v17 konvertiert. Ein Dialogfenster gibt an, dass die Strukturdatei konvertiert wird und sich dann nicht mehr mit einer früheren Version des Programms öffnen lässt.
13 & 14	Anwendungen werden direkt beim Öffnen mit 4D v17 konvertiert. Es erscheint das Dialogfenster für die Strukturdatei (siehe oben) und dann ein weiteres Dialogfenster mit der Meldung, dass die Dateidatei konvertiert wird und sich dann nicht mehr mit einer Version vor v17 öffnen lässt.
Älter	Weitere Informationen zum Konvertieren von Anwendungen in älteren Versionen finden Sie in der Dokumentation zu 4D v15 unter Konvertierung älterer Anwendungen .
Komponenten	4D v17 kann Komponenten in v16, v15, v14, v13, v12 oder v11 öffnen, kompiliert oder interpretiert, es erscheint kein Dialog zur Konvertierung oder zum Bestätigen. Beachten Sie, dass Komponenten immer im Nur-Lesen Modus geöffnet werden. Sie müssen Komponenten nicht erneut kompilieren. Die Konvertierung in v17 ist jedoch nur für .4DB Dateien und nicht für .4DC Dateien möglich.

Konvertieren Sie eine Anwendung in einer Version älter als v14, die keine Primärschlüssel hat, erscheint eine Meldung, dass Primärschlüssel fehlen. Weitere Informationen dazu siehe nächster Absatz.

Fehlerdialog zu Primärschlüssel

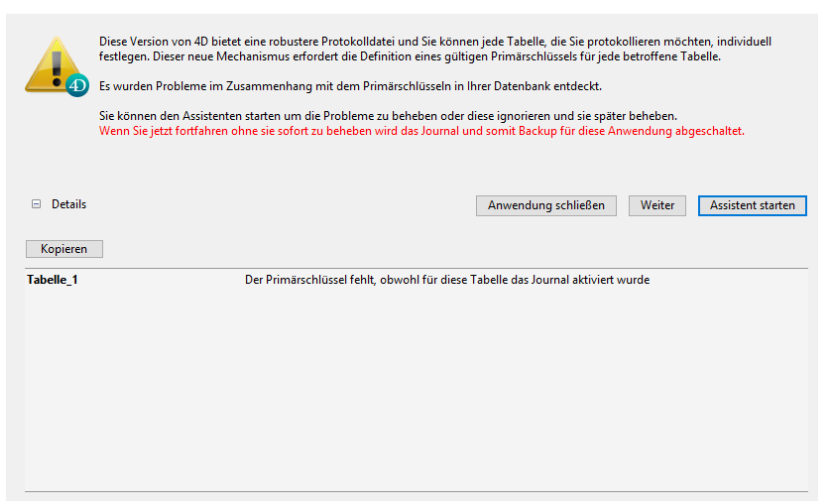
Ab 4D v14 müssen alle Tabellen, die im Logbuch protokolliert werden, einen gültigen Primärschlüssel haben (siehe [Logbuch verwalten](#)). Ab 4D v17 müssen alle exponierten Tabellen einen Primärschlüssel haben, damit sie die Funktionalitäten von ORDA verwenden können (siehe unter [ORDA](#)).

Öffnen Sie in v17 eine Anwendung älter als v14, erscheint ein Fehlerdialog, wenn mindestens eine Tabelle in der 4D Anwendung keinen gültigen Primärschlüssel hat.



Hinweis: Ist das Backup in der Anwendung nicht aktiviert, bietet der Fehlerdialog die Option, die Anwendung trotz fehlender Primärschlüssel zu öffnen.

Haben Sie Zugriff auf die Struktur der Anwendung, zeigt das Fenster mehrere Optionen. Erweitern Sie den Bereich Details, erscheint die Liste der Tabellen, die keinen Primärschlüssel haben:






Sie können dann folgendes ausführen:

- **Kopieren:** Kopiert die Information des Fensters in die Zwischenablage zur weiteren Analyse.
- **Anwendung schließen:** Lässt die Anwendung unverändert und schließt sie.
- **Weiter:** Öffnet die Anwendung ohne Fehlerbehebung. In diesem Fall wird das Protokollieren der Operationen für die Anwendung deaktiviert. Falls die Option **Benutze Logbuch** markiert ist, wird sie deaktiviert. Die Anwendung arbeitet normal in 4D. Sie können diese Option wählen, wenn Sie Ihre Anwendung ohne Logbuch verwenden wollen (wird nicht empfohlen).
- **Assistent starten:** Öffnet das Fenster Primärschlüssel-Manager, um alle Tabellen zu aktualisieren. Wir empfehlen, diese Option zu wählen, um die Anwendung für die neue Version zu aktualisieren. Weitere Informationen dazu finden Sie im Abschnitt **Primärschlüssel-Assistent**.

Haben Sie keinen Zugriff auf die Struktur der Anwendung, können Sie das Protokollieren der Operationen nicht aktivieren. 4D zeigt dann die Meldung, dass Sie sich an den Datenbank-Administrator wenden müssen.

4D 64-bit Versionen nutzen

-  Architektur der 64-bit Applikationen
-  Spezifische Features von 64-bit Applikationen
-  Von 32-bit Versionen auf 64-bit Versionen wechseln

Architektur der 64-bit Applikationen

4D 64-bit Produktreihe

Die gesamte 4D Produktreihe ist jetzt in 64-bit Versionen verfügbar:

- **4D Developer Edition 64-bit** (macOS und Windows)
- **4D Volume Desktop 64-bit** (macOS und Windows)
- **4D Server 64-bit** (macOS und Windows)

Mit diesen Versionen können Sie Ihre 4D Einzelplatz und 4D Remote Anwendungen mit der vollen Leistungsstärke von 64-bit Betriebssystemen nutzen. Die Zertifizierungsmatrix [auf der 4D Web Site](#) enthält die Liste der Betriebssysteme, die für 64-bit Versionen von 4D zertifiziert sind.

Hauptvorteil der 64-bit Technologie ist die Zuweisung von mehr RAM Speicher. Außerdem bietet die 64-bit Architektur neue, leistungsstarke Funktionalitäten wie die Verwaltung von **Preemptive 4D Prozesse** (Multithreading), OS-basierte Druckarchitektur, moderne Editoren für Schnellbericht und Etiketten und die Unterstützung von **Native Objekt Animationen (4D 64-bit Versionen von OS X)**.

Plug-Ins und Komponenten

Nachfolgende Plug-Ins und Komponenten sind für 4D Server, 4D Developer Edition oder 4D Volume Desktop jeweils in 32-bit und 64-bit Architekturen ladbar und verwendbar:

- 4D for OCI
- 4D Internet Commands
- 4D ODBC Pro
- 4D Progress
- 4D SVG
- 4D Widgets
- 4D Write Pro Oberfläche

4D View und 4D Write

4D View und 4D Write sind veraltete Plug-Ins in 32-bit und sind nur mit 32-bit Versionen von 4D verwendbar. 4D bietet nicht-ausführbare 64-bit Versionen dieser Plug-Ins, so dass Entwickler sie laden und in 64-bit Versionen arbeiten können (macOS oder Windows) und dann für 32-bit Versionen kompilieren und weitergeben können. 4D Server 64-bit lädt keine 32-bit Plug-Ins. Sie können jedoch im Ordner Plugins des Server gespeichert werden und auf remote Rechnern ohne Initialisierung vertrieben werden.

Third party Plug-Ins

Da die Komponente **Altura Mac2Win** komplett aus den 64-bit Versionen von 4D entfernt wurde, müssen Sie mit dem Hersteller dieser Komponente abklären, ob sie mit 64-bit Versionen von 4D kompatibel ist.

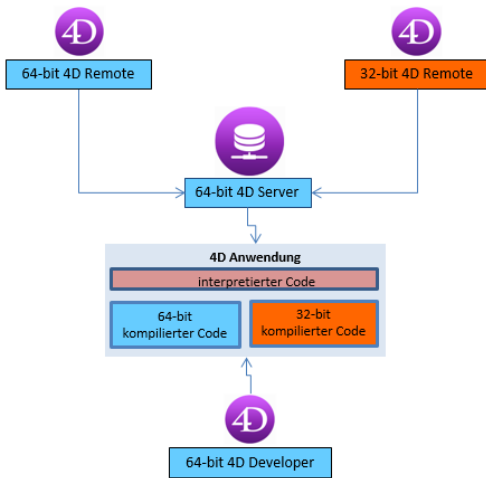
Kompatibilität der Versionen

4D Applikationen für 64-bit Architekturen sind spezifische Versionen für diese Umgebung; d.h. sie laufen nicht auf einem 32-bit System. In der Client/Server Umgebung kann sich jede 4D remote Applikation, egal ob 32-bit oder 64-bit, macOS oder Windows, an eine 64-bit Version von 4D Server anmelden, wenn die entsprechende Version des Datenbankcode auf dem Server verfügbar ist (siehe Überblick unten).

Im interpretierten Modus lässt sich die gleiche 4D Anwendung mit einem 4D Programm in 64-bit oder in 32-bit ausführen (Server oder lokal). Die Entwicklung ist für beide Applikationen gleich (bis auf die im Abschnitt **Spezifische Features von 64-bit Applikationen** genannten Einschränkungen).

Im kompilierten Modus müssen die Anwendungen für den entsprechenden Prozessor kompiliert sein: 64-bit zum Ausführen in einem 4D Programm in 64-bit und 32-bit zum Ausführen in einem 4D Programm in 32-bit. Eine Anwendung, die nur in 32-bit kompiliert wurde und keinen interpretierten Code enthält, lässt sich nicht mit einem 4D Programm in 64-bit ausführen, und umgekehrt. Sie können Ihre Anwendung nur für eine Architektur oder für beide kompilieren. Weitere Informationen dazu finden Sie im Abschnitt **64-bit Kompilierung**.

Überblick der 64-bit Architektur:



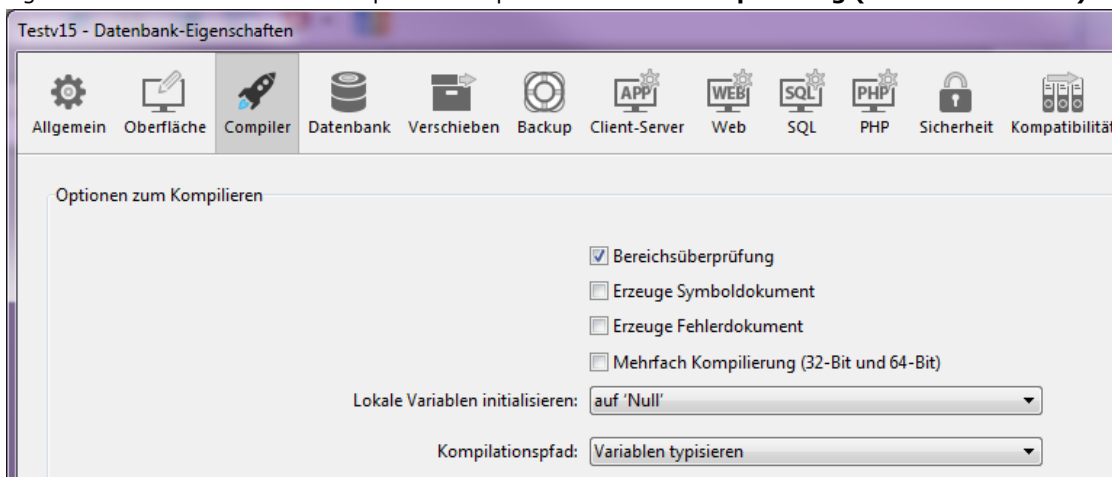
Nachfolgende Tabelle gibt einen Überblick über die Kompatibilität zwischen den verschiedenen 4D Ausführungsumgebungen und dem Code der Datenbank:

	Verfügbarer Code	4D 32-bit	4D 64-bit
4D Server 64-bit	Interpretiert	OK	OK(*)
	64-bit nur kompiliert	-	OK(*)
	64-bit und 32-bit kompiliert	OK	OK(*)
4D Server 32-bit	Interpretiert	OK	OK(*)
	32-bit nur kompiliert	OK	-
	32-bit und 64-bit kompiliert	OK	OK(*)
Lokale Anwendung	Interpretiert	OK	OK
	32-bit nur kompiliert	OK	-
	64-bit nur kompiliert	-	OK
	32-bit und 64-bit kompiliert	OK	OK

(*) Mit 32-bit Versionen von 4D Server (beide Plattformen) und 64-bit Versionen von 4D Server für Windows müssen Sie sicherstellen, dass auf dem Server die Netzwerk-Schicht **ServerNet** aktiviert ist, da die legacy Netzwerk-Schicht in 64-bit Versionen von 4D nicht verfügbar ist. Weitere Informationen dazu finden Sie im Abschnitt **Neue Netzwerk-Schicht ServerNet (Kompatibilität)**.

64-bit Kompilierung

4D Anwendungen lassen sich für 32-bit und 64-bit Prozessoren kompilieren. Dazu verwenden Sie in den Datenbank-Eigenschaften auf der Seite "Compiler" die Option **Mehrfach Kompilierung (32-bit und 64-bit)**:



Ist diese Option markiert, enthält das Kompilat in den Dateien .4DC und .4DB den 64-bit und 32-bit Code. Diese Dateien können dann mit einem 32-bit oder mit einem 64-bit Server laufen. Diese Option ist standardmäßig nicht markiert.

Kompatibilität von kompiliertem Code

Um die OS X 64-bit Architektur zu unterstützen, wurde der in 4D eingebaute Compiler geändert. So können nur Anwendungen, die mit 4D v15 (oder höher) kompiliert wurden, auf OS X 64-bit laufen (Die Änderung des Compiler gilt ab 4D v14 R3). Daraus ergibt sich folgendes:

- Wollen Sie vorhandene 4D Anwendungen auf OS X 64-bit im kompilierten Modus verwenden, müssen Sie diese mit 4D v15 (oder höher) erneut kompilieren
- Verwenden Ihre Anwendungen kompilierte Komponenten, müssen Sie die Komponenten erneut mit 4D v15 (oder höher) kompilieren

Spezifische Features von 64-bit Applikationen

Auch wenn 4D Applikationen in 64-bit in großen Teilen neu geschrieben wurden, sind sie trotzdem weitgehend mit 4D Anwendungen kompatibel, die in 32-bit Versionen entwickelt wurden. Da sie die neuesten Technologien verwenden, mussten wir einige Funktionalitäten aktualisieren und die Unterstützung anderer Funktionalitäten einstellen. Dieser Abschnitt erläutert die verschiedenen Evolutionen.

Hinweise:

- Der Abschnitt [Von 32-bit Versionen auf 64-bit Versionen wechseln](#) beschreibt Schritt-für-Schritt das Migrieren von einer 32-bit Applikation auf eine 64-bit Applikation.
- Features, die bereits in 32-bit Versionen von 4D überholt sind, werden in 64-bit Versionen nicht unterstützt. Weitere Informationen dazu finden Sie im Abschnitt [Übersicht](#) des Handbuchs "[Überholte und entfernte Funktionalitäten](#)".

Aktualisierte Features

Zur Unterstützung der 64-bit Architektur wurden viele 4D Features und Dialoge angepasst oder komplett neu geschrieben. Die meisten Änderungen sind transparent und funktionieren wie in den 32-bit Releases. Einige Editoren wurden jedoch geändert und unterscheiden sich somit von den 32-bit Versionen. Einige grundlegende Features wie z.B. Drucken wurden aktualisiert:

Feature	Betroffene 4D Version	Kommentar
Schnellberichteditor	Mac & Win	Komplett neugeschrieben, siehe Schnellberichte (64-bit)
Etikketteneditor	Mac & Win	Komplett neugeschrieben, siehe Etikketteneditor (64-bit)
Diagramme	Mac & Win	Der Befehl GRAPH akzeptiert einen Parameter vom Typ Objekt zum Definieren von Grafikeinstellungen
Drucken	Mac & Win	Aktualisierte Dialogfenster zum Drucken (Verwenden der standardmäßigen Systemdialogfenster). Das Fenster Drucken erscheint nicht mehr automatisch (siehe Befehl PRINT SETTINGS). Änderungen in den Befehlen SET CURRENT PRINTER und SET PRINT OPTION .
Import/Export Dialogfenster	Mac & Win	Arbeiten wie in 32-bit Version, außer für XML Exporte (XSLT wird nicht mehr unterstützt, siehe unten) und via ODBC Quelle (deaktiviert, siehe unten)
Eingebundene Web Rendering Engine in Web Areas verwenden	Mac & Win	Blink wird als eingebundene Engine in 64-bit Versionen verwendet
Mehrere Instanzen starten	Win	Jedes Starten derselben 4D Applikation öffnet eine neue eigenständige Instanz (mit 32-bit Versionen ließ sich nur eine Instanz derselben 4D Applikation starten)
Plug-Ins 4D Write und 4D View	Mac & Win	Bisherige Plug-Ins sind nicht kompatibel mit 64-bit Versionen von 4D; 4D Write Pro (siehe 4D Write Pro Handbuch) und 4D View Pro verwenden.

Größe des Prozess-Stapels

Der Prozess-Stapel auf einer 64-bit Version von 4D Server benötigt mehr Speicher als eine 32-bit Version (etwa doppelt soviel). Erstellen Sie mit einer 64-bit Version von 4D einen Prozess mit den Funktionen [Execute on server](#) oder [New process](#), empfehlen wir, im Parameter *Stapel* den Standardwert (0) zu setzen, andernfalls erhalten Sie u.U. den Fehler "out of stack".

Deaktivierte Features

Einige Features sind in 64-bit Versionen von 4D deaktiviert:

Feature/Technologie	Betroffene 4D Version	Kommentar
Import/Export via ODBC Datenquelle	Mac & Win	Deaktiviert
Schnellberichteditor: Ränder	Mac & Win	Deaktiviert
Schnellberichteditor: Die Oberfläche per Programmierung steuern (QR ON COMMAND , QR EXECUTE COMMAND)	Mac & Win	Deaktiviert. Der neue Editor mit Werkzeugleiste lässt sich nicht per Programmierung steuern. Verwenden Sie zum Erstellen eigener Oberflächen den eingebundenen Bereich und andere "QR" Befehle.
Etikketteneditor Standard Codes	Mac & Win	Deaktiviert

Von 32-bit Versionen auf 64-bit Versionen wechseln

Für das Upgraden einer bestehenden 4D Anwendung auf OS X von einer 32-bit Version auf eine 64-bit Version sind ein paar Vorarbeiten erforderlich.

Läuft Ihre Anwendung auf 4D Server 64-bit Windows oder OS X, ist das meiste schon erledigt. Für 64-bit Versionen von Desktop Anwendungen sind evtl. zusätzliche Schritte erforderlich. Im folgenden werden die einzelnen Punkte erläutert, damit Sie alle notwendigen Operationen vor und nach dem Upgrade prüfen können.

Für die Migration Ihrer Produkte auf 64-bit wurden verschiedene Features upgedated, deaktiviert, oder als überholt deklariert. Alle Details dazu finden Sie im Abschnitt .

Hinweis: Wie bei allen Upgrade Operationen ist es eine bewährte Praxis, MSC zu verwenden und vor jedem größeren Schritt einen Überprüfungsprozess zu starten, um sicherzustellen, dass Daten und Struktur OK sind.

Plug-Ins prüfen

In der ersten Etappe updaten Sie Ihre Plug-Ins (sofern vorhanden) auf die jeweiligen 64-bit Versionen:

- **4D Plug-Ins:**

Alle Plug-Ins arbeiten bereits in 64-bit Versionen, mit Ausnahme von 4D Write und 4D View.

- Verwenden Sie Ihre Anwendung 4D Write, sollten Sie die Migration Ihres Code auf 4D Write Pro vornehmen. Ein guter Übergang ist, Ihren vorhandenen 32-bit Code beizubehalten und daneben ein neues auf 64-bit basierendes Modul mit 4D Write Pro zu starten.
- Verwenden Sie Ihre Anwendung 4D View, müssen Sie die Funktionalitäten von 4D View Pro oder andere Alternativen einsetzen.

- **Third-party Plug-Ins:**

Wenden Sie sich an die jeweiligen Hersteller, um 64-bit Versionen zu erhalten

Vorbereitungen in der 32-bit Version

1. Upgraden Sie Ihre Anwendung auf das neueste 32-bit Release, z.B. 4D v16 32-bit
2. Stellen Sie sicher, dass Unicode aktiviert ist.
3. Konvertieren Sie alle Bilder im Format PICT/cicn/QuickTime.
Über den Befehl **GET PICTURE FORMATS** können Sie veraltete Bildformate in Ihren Daten ausfindig machen. Sie müssen auch alle nicht-unterstützten Bilder in der Struktur Ihrer Datenbank ersetzen. Eine Überprüfung mit dem MSC findet veraltete Bilder in Ressourcen Dateien für Bild- oder 3D Schaltflächen sowie statische Bilder.
4. Ersetzen Sie auf XSLT-basierende Features (Befehle **_o_XSLT APPLY TRANSFORMATION**, **_o_XSLT SET PARAMETER** oder **_o_XSLT GET ERROR**) z.B. mit dem Befehl **PROCESS 4D TAGS**.
5. Ersetzen Sie Aufrufe von **_o_Font number** durch Aufrufe von Schriftnamen.
6. Entfernen Sie allen Code, der Ressourcen-Dateien erstellt oder ändert.

Jetzt ist alles bereit, um Ihre Anwendung mit einer 64-bit Version von 4D zu öffnen.

Anwendung in 64-bit öffnen und prüfen

1. Öffnen Sie Ihre Anwendung mit einer 4D Developer Edition in 64-bit.
2. Setzen Sie die eingebundene Web Rendering Engine für Ihre Web Bereiche ein, überprüfen Sie diese, da automatisch zu Blink gewechselt wird (siehe **Blink verwenden (Integrierte Web Rendering Engine in 4D 64-bit Versionen)**).
3. Verwenden Sie Ihr Code die Konstante [Mac spool file format option](#) von **SET PRINT OPTION**, müssen Sie diese ersetzen mit Aufrufen von **SET CURRENT PRINTER** mit der Konstanten [Generic PDF driver](#).
4. Erstellen Sie Ihre Code Prozesse über die Funktion **Execute on server** oder **New process**, überprüfen Sie, ob im Parameter *Stapel* der Standardwert (0) gesetzt ist.
5. Überprüfen Sie Aufrufe und Verwendungen des Etiketteneditors (siehe **Etiketteneditor (64-bit)**).
6. Überprüfen Sie Aufrufe und Verwendungen des Schnellberichteditors (siehe **Schnellberichte (64-bit)**)

Ihre Anwendung ist jetzt voll kompatibel mit 64-bit und Sie können die Vorzüge aller neuen Features in der 64-bit Version von 4D nutzen.







Ihre Vorteile mit 64-bit Features

Die Highlights sind:

- Die **64-bit Architektur** hebt die Begrenzung des Datenbank Cache auf. Verbessern Sie die Performance Ihrer Anwendung einfach durch Verwenden eines größeren Cache.
Setzen Sie die **leistungsstarken neuen Features** in 64-bit ein, wie z.B. preemptive Prozesse, animierte Formularobjekte oder neue Features zum Drucken
Erstellen Sie Ihre Anwendungen mit der 4D Runtime Volume License 64-bit.
Verwenden Sie die final 64-bit Versionen von 4D Server auf Windows und macOS
- **Neuer Schnellberichteditor**, kompatibel mit Berichten, die mit einer vorigen Version erstellt wurden - siehe unter **Etiketteneditor (64-bit)**.

- **Neuer Etiketteneditor**, kompatibel mit Etikettendateien, die mit einer vorigen Version erstellt wurden - siehe unter **Schnellberichte (64-bit)**.
- Mit dem Befehl **GRAPH** können Sie **Diagramme** mit dem Parameter vom Typ *Objekt* in einem Aufruf erstellen.

Einstellungen

-  Überblick
-  Seite Allgemein
-  Seite Struktur
-  Seite Formulare
-  Seite Methoden
-  Seite Abkürzungen

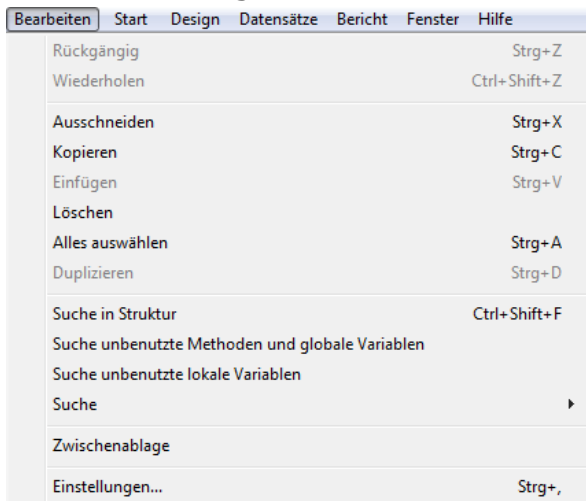
Benutzereinstellungen bestimmen die Optionen des Standardverhaltens Ihrer Arbeitsumgebung, z.B. die Farben im Methodeneditor, Optionen für die automatische Erstellung von Formularen, Standardanzeige in Formularen, etc. Sie gelten für alle Anwendungen, die mit dem Programm 4D oder 4D Server geöffnet werden.

4D Server: Objektsperre tritt ein, wenn zwei oder mehr Benutzer zur gleichen Zeit versuchen, die 4D Einstellungen zu verändern. Das Dialogfenster 4D Einstellungen kann immer nur ein Benutzer verwenden.

Hinweis: 4D bietet weitere Parameter, die nur für die geöffnete Anwendung gelten. Weitere Informationen dazu finden Sie im Kapitel **PICTURE LIBRARY LIST**.

Zugriff

Das Dialogfenster Einstellungen rufen Sie unter Windows über den Befehl **Einstellungen** im Menü **Bearbeiten**, auf Mac OS im Menü **Anwendung** auf:



Dieser Menübefehl ist auch verfügbar, wenn keine Anwendung geöffnet ist.

Das Dialogfenster Einstellungen lässt sich auch im Anwendungsmodus über die Standardaktion "Einstellungen" aufrufen, die einem Menübefehl oder einer Schaltfläche zugeordnet ist oder über den 4D Befehl **OPEN SETTINGS WINDOW**.

Speichern

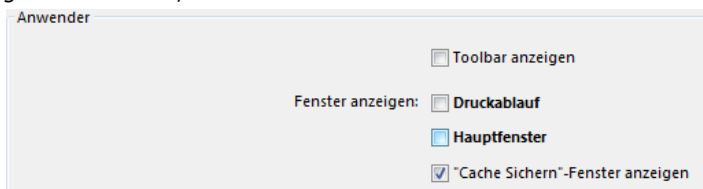
Die hier gewählten Optionen werden in einer Datei Voreinstellungen im XML Format gespeichert. Sie hat den Namen **4D Preferences vXX.4DPreferences** und wird im Ordner Preferences des aktuellen Benutzers gespeichert:

- Windows 7 und höher: {Festplatte}:C\Users\BenutzerName\AppData\Roaming\4D
- macOS: {Festplatte}:Users:BenutzerName:Library:Application Support:4D

Hinweis: XX gibt die Nummer der 4D Hauptversion an. In Version 17 hat die Datei den Namen **4D Preferences v17.4DPreferences**

Parameter anpassen und "Standardeinstellungen"

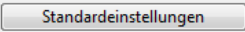
In den Dialogfenstern Einstellungen und Datenbank-Eigenschaften erscheinen die Parameterbezeichnungen, deren Werte sich geändert haben, in **Fett**:



Die Änderung kann im Dialogfenster selbst erfolgt sein, oder bei einer konvertierten Datenbank von einer Voreinstellung kommen, die zuvor verändert wurde.

Ein Parameter erscheint weiterhin in Fettschrift, wenn sein Wert manuell mit dem Standardwert ersetzt wird. Auf diese Weise lässt sich immer nachvollziehen, welche Parameter angepasst wurden.

Um die Parameter auf ihre Standardwerte zurückzusetzen und die Markierung zu entfernen, können Sie am unteren Rand auf die Schaltfläche **Standardeinstellungen** klicken:

A rectangular button with a light gray background and a thin border. The text "Standardeinstellungen" is centered on the button in a dark gray font. The button is located in the top-left corner of a larger light gray rectangular area.

Standardeinstellungen

Diese Schaltfläche setzt alle Parameter der aktuellen Seite zurück. Sie wird aktiv, wenn mindestens ein Parameter auf dieser Seite geändert wurde.

Diese Seite enthält verschiedene Optionen zur allgemeinen Funktionsweise Ihrer 4D Anwendung.

Optionen

Die Optionen in diesem Bereich konfigurieren verschiedene automatische Funktionen, die beim Starten und beim Wechseln in die Anwendungsumgebung verwendet werden.

Beim Starten

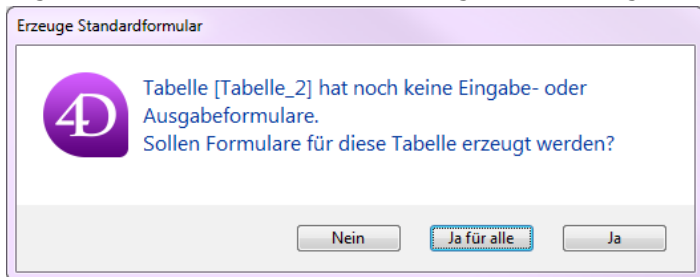
Diese Option konfiguriert die Standardanzeige von 4D beim Starten, wenn der Benutzer nur das Programm startet.

- **Nichts:** Es erscheint nur das leere Anwendungsfenster.
- **Öffne Dialog für lokale Anwendung:** 4D zeigt den Standard-Öffnen Dialog, in dem Sie eine lokale Anwendung bestimmen können.
- **Zuletzt verwendete Anwendung öffnen:** 4D öffnet direkt die zuletzt verwendete Anwendung. Es erscheint kein Öffnen-Dialog.
Hinweis: Soll das Öffnen-Fenster erscheinen, wenn diese Option ausgewählt ist, drücken Sie beim Starten der Anwendung unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste**.
- **Öffne Dialog für Server Auswahl:** 4D zeigt die Verbindung zum 4D Server Dialog, in dem Sie eine Anwendung bestimmen können, die im Netzwerk veröffentlicht wird.
- **Willkommen Assistent öffnen** (Standardeinstellung): 4D zeigt den Willkommensdialog.

Hinweis 4D Server: Die 4D Server Applikation ignoriert diese Option. In dieser Umgebung wird immer die Option **Nichts** verwendet.

Automatische Formularerstellung

Hier legen Sie das Verhalten von 4D beim Erstellen einer Tabelle im Struktureditor fest und öffnen dann z.B. das Fenster zum Anzeigen der Datensätze. Standardmäßig teilt 4D mit, dass für die neue Tabelle kein Formular erstellt wurde und bietet die Möglichkeit, automatisch ein Standard-Eingabe- und -ausgabeformular anzulegen:



Es gibt drei Optionen:

- **Niemals:** Es erscheint weder eine Meldung, noch wird ein Standardformular angelegt.
- **Fragen:** Die Meldung erscheint systematisch, wenn für die Tabelle kein Formular angelegt wurde. Das ist die Standardeinstellung.
- **Immer ja für alle:** Es erscheint keine Meldung. Für alle Tabellen werden automatisch Standardformulare angelegt.

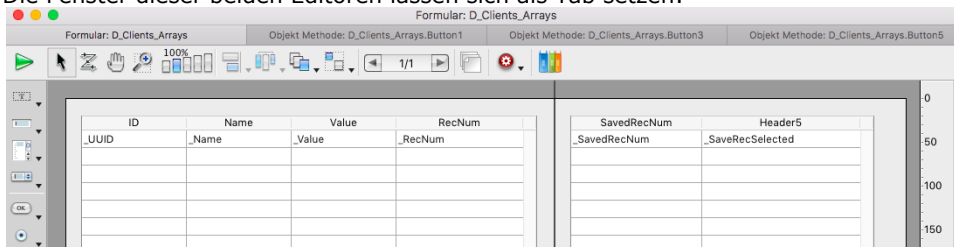
Fenster Tabs (nur macOS)

Ab macOS Sierra können Mac Anwendungen automatisch die Funktionalität *Fenster Tab* zum Organisieren mehrfacher Fenster nutzen: Dokumentfenster werden in einem einzigen übergeordneten Fenster gestapelt und lassen sich über Tabs aufrufen. Das ist hilfreich bei kleinen Bildschirmen oder bei Pfadverfolgung.

Sie können dieses Feature nur mit 4D 64-bit Versionen nutzen, und zwar für:

- Fenster des Methodeneditors
- Fenster des Formulareditors

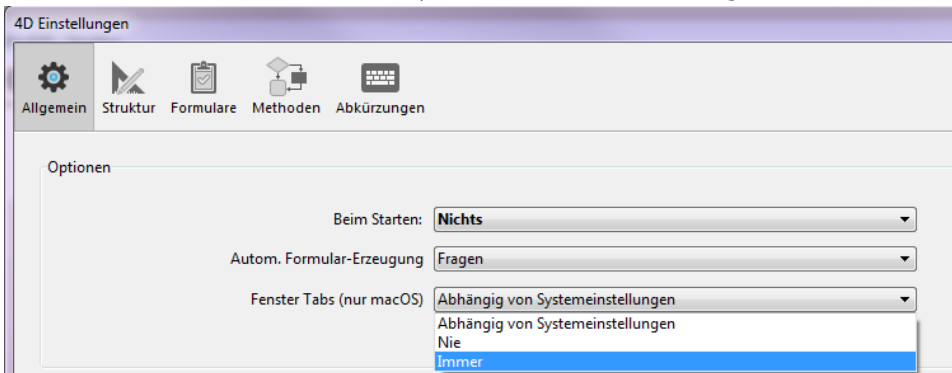
Die Fenster dieser beiden Editoren lassen sich als Tab setzen:



Sie werden über einen Befehlssatz im Menü **Fenster** verwaltet:



Dieses Feature steuern Sie über die Option **Fenster Tab** im Dialogfenster 4D Einstellungen:



Es gibt drei Optionen:

- **Abhängig von Systemeinstellungen** (Standard): 4D Fenster verhalten sich, wie in den macOS Systemeinstellungen definiert wurde (Vollbildschirm, Immer oder Manuell).
- **Nie**: Beim Öffnen eines neuen Dokuments im 4D Formular- oder Methodeneditor wird immer ein neues Fenster erstellt (Tabs werden nie erstellt).
- **Immer**: Beim Öffnen eines neuen Dokuments im 4D Formular- oder Methodeneditor wird immer ein neuer Tab erstellt.

Design-Umgebung verlassen beim Wechsel in Anwendungs-Modus

Ist diese Option markiert, werden beim Wechsel in die Anwendungs-Umgebung über den Menübefehl **Anwendung testen** alle Fenster der Designumgebung geschlossen. Diese Option ist standardmäßig nicht markiert, d.h. die Fenster der Designumgebung bleiben im Hintergrund der Anwendungs-Umgebung sichtbar.

Beim Erzeugen einer neuen Anwendung

Diese Gruppe Optionen definiert die Konfiguration, die beim Erstellen einer neuen Anwendung gelten soll.

Automatische Backup-Strategie für alle neuen Datenbanken aktivieren

Standardmäßig verwendet jede mit 4D erstellte Datenbank ein Logbuch (die Option auf der [Seite Allgemein](#) der 4D Einstellungen ist markiert). Weitere Informationen dazu finden Sie im Abschnitt [Logbuch verwalten](#).

Für neue Anwendungen Paket erstellen

Ist diese Option markiert, werden 4D Anwendungen automatisch in einem Ordner mit der Endung `.4dbase` angelegt. Auf diese Weise erscheinen die Datenbankordner auf Mac OS als Pakete mit spezifischen Eigenschaften. Unter Windows hat diese Option keine Auswirkung. Weitere Informationen dazu finden Sie im Abschnitt [Beschreibung der 4D Dateien](#).

Sprache für Textvergleiche

Dieser Parameter definiert die Standardsprache zum Bearbeiten und Vergleichen von Zeichenketten in neuen Anwendungen. Die Sprachwahl wirkt sich direkt auf Sortieren und Suchen von Text und auf Zeichen aus, jedoch nicht auf die Übersetzung von Text bzw. die Formate für Datum, Zeit und Währung. Diese bleiben in der Sprache des Systems. 4D verwendet standardmäßig die aktuelle Benutzersprache des Systems.

Eine 4D Anwendung kann also in einer anderen Sprache laufen, als das System vorgibt. Beim Öffnen der Anwendung prüft die 4D Engine die von der Datendatei verwendete Sprache und gibt sie an die Programmiersprache weiter (interpretierter oder kompilierter Code). Textvergleiche werden dann in derselben Sprache ausgeführt, unabhängig ob sie von der Datenbank-Engine oder der Programmiersprache ausgeführt werden.

Beim Erstellen einer neuen Datendatei verwendet 4D die Sprache, die zuvor in diesem DropDown-Menü gesetzt wurde. Öffnen Sie eine Datendatei, die nicht in derselben Sprache wie die Strukturdatei ist, wird die Sprache der Datendatei verwendet und der Sprach-Code wird in die Struktur kopiert.

Hinweis: Sie können diesem Parameter für die geöffnete Anwendung über die Datenbank-Eigenschaften ändern. Weitere Informationen dazu finden Sie im Abschnitt [Text-Vergleiche](#).

Speicherort der Dokumentation

Dieser Bereich legt den Zugriff auf die 4D HTML Dokumentation fest, die im aktuellen Browser angezeigt wird:

- Wenn der Benutzer auf der [Seite Befehle](#) im Explorer auf einen Befehl doppelklickt.
- Wenn der Benutzer auf einen Befehlsnamen im Methodeneditor klickt (siehe [Komplette Beschreibung eines Befehls anzeigen](#)) und die Taste **F1** drückt.

Sie können wählen, ob Sie direkt auf die 4D online Dokumentation (*4D Doc Center*) zugreifen wollen oder auf die statische Version, die lokal gespeichert wird.

Lokaler Ordner

Gibt den Speicherort der statischen HTML Dokumentation an. Das ist standardmäßig der Unterordner `Help/command/de`. Sie können den Ort sehen, wenn Sie in das zugeordnete DropDown-Menü klicken. Ist der Unterordner nicht vorhanden, erscheint er in rot.

Sie können diese Position bei Bedarf verändern, z.B. wenn Sie die Dokumentation in einer anderen Sprache als die der Anwendung anzeigen wollen. Die statische HTML Dokumentation kann auf einem anderen Volume, einer CD-Rom, etc. liegen. Um eine andere Position zu bestimmen, klicken Sie auf die Schaltfläche [...] neben dem Eingabebereich und wählen einen Dokumentation Root Ordner (entsprechend der Sprache: *fr, en, es, de* oder *ja*).

Hinweis: Weitere Informationen zum Downloaden der statischen HTML Dokumentation finden Sie auf der [Home Page](#) von 4D Doc Center.

Web Site

URL Zugriff auf die online Dokumentation für die Version in 4D Doc Center. 4D erstellt Aufrufe auf die Seiten der Dokumentation gemäß dieser URL. Sie können ihn verändern, wenn Sie z.B. die Dokumentation in einer anderen Sprache als die der Anwendung anzeigen wollen. Der Test-Icon neben dem Bereich startet den Standardbrowser, um zur angegebenen URL zu gehen.

Zuerst lokaler Ordner

Diese Option (standardmäßig markiert) bestimmt, wo 4D nach der Seite Dokumentation sucht, die über den Explorer oder die Taste F1 aufgerufen wird.

- Ist sie markiert, sucht 4D zuerst im lokalen Ordner nach der Seite. Wird sie gefunden, zeigt 4D sie im aktuellen Browser an. Wenn nicht, sucht 4D automatisch in der Online-Dokumentation der Web Site danach. So ist es möglich, auf die Dokumentation über eine lokale Version zuzugreifen, wenn Sie nicht im angemeldeten Modus sind.
- Ist sie nicht markiert, sucht 4D direkt in der Online-Dokumentation der Web Site nach der gewünschten Seite und zeigt sie im aktuellen Browser an. Wird sie nicht gefunden, zeigt 4D eine Fehlermeldung im Browser. Diese Option bietet schnelleren Zugriff, ist aber nur verwendbar, wenn 4D im angemeldeten Modus ist.

Primärschlüssel

In den 4D Einstellungen auf der Seite Struktur können Sie Name und Typ der Felder mit Primärschlüssel ändern, die 4D oder der **Primärschlüssel-Assistent** automatisch beim Erstellen einer neuen Tabelle verwenden.

Es gibt folgende Optionen:

- **Name** (standardmäßig "ID"): Setzt den Standardnamen der Felder mit Primärschlüssel. Sie können einen beliebigen Namen verwenden, sofern er die geltenden Namensregeln für 4D Standardtabellen berücksichtigt (siehe **Namensregeln für Tabellen und Datenfelder**).
- **Standardtyp** (standardmäßig Lange Ganzzahl): Setzt den Standardtyp der Felder mit Primärschlüssel. Wählen Sie den Typ UUID, sind die mit Standard Primärschlüssel erstellten Felder vom Typ Alpha und die Eigenschaften und markiert.

Struktur-Editor

Diese Gruppe Optionen bestimmt die Anzeige des 4D Struktureditors.

Graphische Qualität der Struktur

Diese Option variiert die grafische Darstellung im Struktureditor. Standardmäßig ist **Hoch** eingestellt. Wählen Sie **Standard**, um eine raschere Anzeige zu erzielen. Die Auswirkung dieser Einstellung ist hauptsächlich bei Verwenden der Zoom-Funktion erkennbar. Weitere Informationen dazu finden Sie im Abschnitt "Zoom" im **Struktureditor**.

Ist ein Ordner ausgeblendet, Darstellung

Diese Option bestimmt die Anzeige ausgeblendeter Tabellen im Struktureditor, wenn Sie Auswahlen nach Ordner ausführen (siehe Abschnitt **Tabellen nach Ordner ein-/Ausblenden**). Sie können wählen zwischen **Umriss** (die Tabelle wird als Schatten dargestellt) und **Unsichtbar** (die Tabelle verschwindet komplett).

Auf dieser Seite definieren Sie die Standardeinstellung und -anzeige des 4D Formulareditors.

Verschieben

Diese Gruppe Optionen setzt Parameter zum Bewegen von Objekten im Formulareditor über Tastatur oder Maus.

Verschieben mit Tastatur

Diese Option setzt den Schritt der Verschiebung in Punkt, wenn ein Objekt per Tastatur und **Umschalttaste** bewegt bzw. in der Größe verändert wird.

Wenn außerhalb Fensterbereich verschoben

Diese Option bestimmt das Verhalten des Formulareditors, wenn ein Objekt mit der Maus über das Fenster hinaus bewegt wird.

- **Autoscroll:** Damit wird das Formular im Fenster gescrollt, als ob Sie auf einen der Rollbalken geklickt hätten. Diese Einstellung ist hilfreich, um Objekte in großen Formularen zu bewegen.
- **Starte Drag & Drop:** Damit wird die Aktion als Drag&Drop interpretiert. Das Formularfenster wird nicht geändert und das bewegte Objekt kann in ein anderes Fenster gezogen werden (bei kompatiblen Inhalten), z.B. in einem anderen Formular. Dieses Verhalten ist hilfreich, um Objekte zwischen verschiedenen Formularen zu recyceln oder bei Objektbibliotheken. Weitere Informationen dazu finden Sie im Abschnitt [Eigene Objektbibliothek anlegen und einsetzen](#).

Sie können diese Option je nach Ihren Arbeitsweisen oder Entwicklungsanforderungen einstellen.

Aktiviere autom. Ausrichtung als Standard

Diese Option aktiviert die automatische Ausrichtung als Standard in allen neuen Fenstern im Formulareditor. Sie können diese Option für jedes Fenster individuell verändern. Weitere Informationen dazu finden Sie im Abschnitt [Magnetraster verwenden](#).

Standard Anzeige

Hier bestimmen Sie, welche Elemente in den neuen Fenstern des Formulareditors standardmäßig angezeigt bzw. ausgeblendet sind. Über das hierarchische Menü **Anzeige** des Formulareditors können Sie die Anzeige für jedes Fensters individuell einstellen.

- **Standard Anzeige:** Hier bestimmen Sie, welche Elemente in den neuen Fenstern des Formulareditors standardmäßig angezeigt werden. Weitere Informationen dazu finden Sie im Abschnitt [Erkennungszeichen verwenden](#).
- **Farbe für Markierungslinien:** Hier können Sie die Farbe für Markierungslinien im Formulareditor verändern, um die einzelnen Bereiche besser zu unterscheiden (Kopfteil, Umbruch, Hauptbereich, Fußteil, etc.) Weitere Informationen dazu finden Sie im Abschnitt [Steuerlinien für Ausgabe verwenden](#).

Diese Seite enthält Parameter für die Oberfläche des Methodeneditors und seine Standardanzeige sowie dazugehörige Optionen. Sie ist aufgeteilt in die Bereiche **Stile**, **Optionen** und **Farben**.

Stile

Schrift

Diese DropDown-Menüs setzen Schriftart und -größe für den Eingabebereich des Methodeneditors:

- **Schrift:** Name der verwendeten Schrift
- **Größe:** Größe der Zeichen

Sie können den Schrifttyp für jeden Objekttyp anders einstellen.

Syntax Stile

Die Optionen in diesem Bereich können jedem Elementtyp der 4D Programmiersprache, z.B. Felder, Tabellen, Variablen, Parameter, SQL, einen bestimmten Stil und Farbe zuweisen. Die Kombination verschiedener Farben und Stilarten sind besonders hilfreich zum Pflegen des Code.

- **Sprache:** In diesem DropDown-Menü können Sie entweder die Sprache **4D** oder **SQL** auswählen. Sie können dann jedem Element in der Liste einen eigenen Stil zuordnen. Die Einträge variieren je nach der gewählten Sprache.
- **Liste der Elementtypen:** Markieren Sie die Stilooptionen in der Tabelle, um den Einträgen entsprechende grafische Attribute zuzuweisen. Sie können pro Eintrag auch mehrere Optionen kombinieren. Der Eintrag in der Spalte Elementtyp zeigt die Änderung. Klicken Sie auf den Farb-Icon zum Eintrag, öffnet sich eine Farbpalette, wo Sie eine neue Farbe auswählen können.

Hinweise:

- In der 4D Programmiersprache bezeichnet "Normaler Text" allen Text, der nicht zu einem bestimmten Typ gehört, wie z.B. Symbol, Zeichensetzung, Konstanten.
- In der 4D Programmiersprache bezeichnet "Schlüsselwörter" Code Strukturen, wie If/End if, Case of/End case, die über Makros zugänglich sind. In der SQL Sprache bezeichnet "Schlüsselwörter" alle Befehle und die dazugehörigen Schlüsselwörter.

Optionen

Verwende regionale Systemeinstellungen: ermöglicht, für die aktuelle 4D Applikation die "internationale" Einstellung für Code zu deaktivieren/aktivieren:

- Ist diese Option deaktiviert (Standard in 4D v15 und höher), verwenden 4D Methoden die Englisch-US Einstellung und die englischen Namen der Programmiersprache.
- Ist diese Option markiert, verwenden 4D Methoden die regionalen Einstellungen wie in den bisherigen 4D Versionen.

Ändern Sie diese Option, müssen Sie die 4D Applikation neu starten, damit die Einstellung berücksichtigt wird. Weitere Informationen dazu finden Sie im Abschnitt [Einführung in die 4D Sprache](#).

Optionen

Dieser Bereich definiert die Anzeigeeoptionen des Methodeneditors:

- **Einrückung:** Hier können Sie den Wert der Einrückung für Code im Methodeneditor verändern. Er wird in Punkten angegeben. Standardmäßig sind 10 Punkte vorgegeben:

```
1  =If ($vListItemPos#0)
2      // Get the list item informatior
3      GET LIST ITEM(hlList,$vListItemPos,$v
4      // Is the item a Department item
5      =If ($vListItemRef ?? 31)
6          // If so, it is a double-click
7          ALERT("You double-clicked on the
8      =Else
9          // If not, it is a double-click
10         // Using the parent item ID to
11         $vDepartmentID:=List item parent
12         QUERY([Departments];[Departments]
13         // Tell where the Employee is v
14         ALERT("You double-clicked on the
15     End if
16 End if
17
18 standard indentation
```

Die Veränderung des Standardwerts ist hilfreich, wenn Ihre Methoden komplexe Algorithmen mit vielen eingebundenen Ebenen enthalten. Weniger Einrückung vereinfacht auch das Scrollen in horizontaler Richtung.

- **Zeilennummern anzeigen:** Damit können Sie die standardmäßigen Zeilennummern jedes Fensters im Methodeneditor anzeigen. Sie können die Nummern für das aktuelle Fenster auch direkt vom Methodeneditor aus ein- oder ausblenden.
- **Listen einblenden:** Damit bestimmen Sie, ob die Objektlisten (Befehle, Tabellen, Felder, etc.) beim Öffnen des Methodeneditors standardmäßig ein- oder ausgeblendet werden. Sie können jede Liste auch direkt im Methodeneditor ein-/ausblenden.

- **Logische Blöcke hervorheben** (*neu in v16*): Damit wird der ganze Code in einem logischen Block hervorgehoben (z.B. **If/End if**), wenn Sie die Maus auf den aufgeklappten Icon setzen. Diese Option ist standardmäßig inaktiv:

```

12  If (<>PS_EditMovies=0)
13      ↳ <>PS_EditMovies:=New process($CurrentMethName;
14  Else
15      BRING TO FRONT(<>PS_EditMovies)
16  End if
17

```

Hinweis: Die Markierungsfarbe legen Sie auf der Seite "Farben" fest.

- **Blockumrahmungen anzeigen** (*neu in v16*): Damit können Sie vertikale Blockrahmen permanent ausblenden. Die Blockrahmen verbinden Icons zum Ein- und Ausblenden visuell miteinander. Sie werden standardmäßig angezeigt (außer die Ein-/Ausblenden Icons werden nicht angezeigt, siehe nächste Option).

<pre> 9 If (Count 10 11 \$CurrentM 12 If (<>PS_ 13 <>PS_Ed: 14 Else 15 BRING TO 16 End if 17 18 Else 19 </pre>	<pre> 9 If (Count 10 11 \$CurrentM 12 If (<>PS_ 13 ↳ <>PS_Ed: 14 Else 15 BRING TO 16 End if 17 18 Else 19 </pre>
--	--

- **Ein-/Ausblenden Icons nicht anzeigen** (*neu in v16*): Damit können Sie beim Anzeigen von Code alle Ein-/Ausblenden Icons standardmäßig ausblenden. Ist diese Option markiert, erscheinen die Icons sowie lokale Blocklinien temporär, wenn Sie die Maus auf einen Icon setzen:

<pre> 9 If (Count 10 11 \$CurrentM 12 IF (<>PS_ 13 <>PS_Ed: 14 Else 15 BRING TO 16 End if 17 18 Else 19 </pre>	<pre> 9 If (Count 10 11 \$CurrentM 12 ↳ If (<>PS_ 13 ↳ <>PS_Ed: 14 ↳ Else 15 ↳ BRING TO 16 ↳ End if 17 18 ↳ Else 19 </pre>
--	--

- **Passende Klammern:** Ändert die grafische Hervorhebung zueinander passender Klammern im Code. Die Hervorhebung erscheint immer, wenn eine Klammer ausgewählt ist. Es gibt folgende Optionen:

- **Nichts:** Keine Anzeige
- **Viereck:** Klammern werden mit schwarzer Linie umrahmt
- **Hintergrundfarbe:** Klammern werden farblich hervorgehoben. Die Farbe wird im Bereich "Farbe" festgelegt, siehe unten Absatz "Farbe".
- **Fett:** Klammern erscheinen in Fettschrift

Standardmäßig ist die Option Viereck ausgewählt:

```
INSERT MENU ITEM(main bar;-1;Get indexed string(79;1);FileMenu)
```

- **Text hervorheben** (*neu in v16*): Markiert in einem offenen Methodenfenster alle Vorkommen des gleichen Textes (Variable, Feld...).

```

4  C_LONGINT(<>PS_EditMovies)
5  C_LONGINT($window)
6  C_TEXT($CurrentMethName)
7  C_LONGINT($win)
8
9  If (Count parameters=0)
10
11  $CurrentMethName:=Current method
12  If (<>PS_EditMovies=0)
13  <>PS_EditMovies:=New process($C
14  Else
15  BRING TO FRONT(<>PS_EditMovies)
16  End if

```

Es gibt folgende Optionen:

- **Nein** (Standard): Keine Hervorhebung
- **Bei Mouse-over:** alle Vorkommen werden hervorgehoben, wenn der Text angeklickt wird
- **Textauswahl:** alle Vorkommen werden hervorgehoben, wenn der Text ausgewählt wird

Hinweis: Die Markierungsfarbe legen Sie auf der Seite "Farben" fest.

- **Aktuelle Zeile hervorheben:** Hebt die aktuell laufende Zeile im Debugger hervor (siehe *4D Programmiersprache* unter **Debugging**), zusätzlich zum regulär angezeigten gelben Pfeil.

```

// Create a new empty
hlList:=New list
// Select all the rec

```

Deaktivieren Sie diese Option, erscheint nur der gelbe Pfeil am Rand.

Vorschläge

Dieser Bereich bestimmt die Funktionsweisen zur automatischen Vervollständigung im Methodeneditor, um diese an Ihre eigenen Arbeitsweisen anzupassen.

- **Automatisches Öffnen eines Fensters für:** öffnet automatisch ein Fenster mit Vorschlägen für Konstanten, lokale und Interprozess Variablen, Objektattribute und Tabellen. Ist z.B. die Option "Lokale und Interprozess Variablen" markiert, erscheint das Fenster mit Vorschlägen, wenn Sie das Zeichen \$ eingeben:

```
GET LIST ITEM($
```

```

$hSubList
$vbItemSubExpanded
$vDepartment
$vDepartmentID
$vEmployee
$vItemPos
$vItemRef
$vItemSubList
$vNbEmployees
$vsItemText

```

Sie können diese Funktionsweise für bestimmte Elemente der Programmiersprache deaktivieren.

- **Bestätigung des Vorschlags für:** legt die Eingabeart fest, über die der Methodeneditor den im Fenster ausgewählten Eintrag automatisch übernimmt.
 - **Tab und Begrenzer:** Damit können Sie den ausgewählten Eintrag durch Drücken der Tabulatortaste oder einen zum Kontext passenden Begrenzer übernehmen. Geben Sie z.B. "arr" und "{" ein, schreibt 4D automatisch "arr{" in den Editor. Folgende Begrenzer werden berücksichtigt: ; : = < [{
 - **Nur per Tab:** Damit wird der ausgewählte Vorschlag nur über die Tabulatortaste eingefügt. Das erleichtert die eigene Eingabe von Begrenzer-Zeichen in Elementnamen, wie z.B. `#{1}`.
- Hinweis:** Sie können den Vorschlag durch Doppelklick in das Fenster oder über die Zeilenschaltung bestätigen.

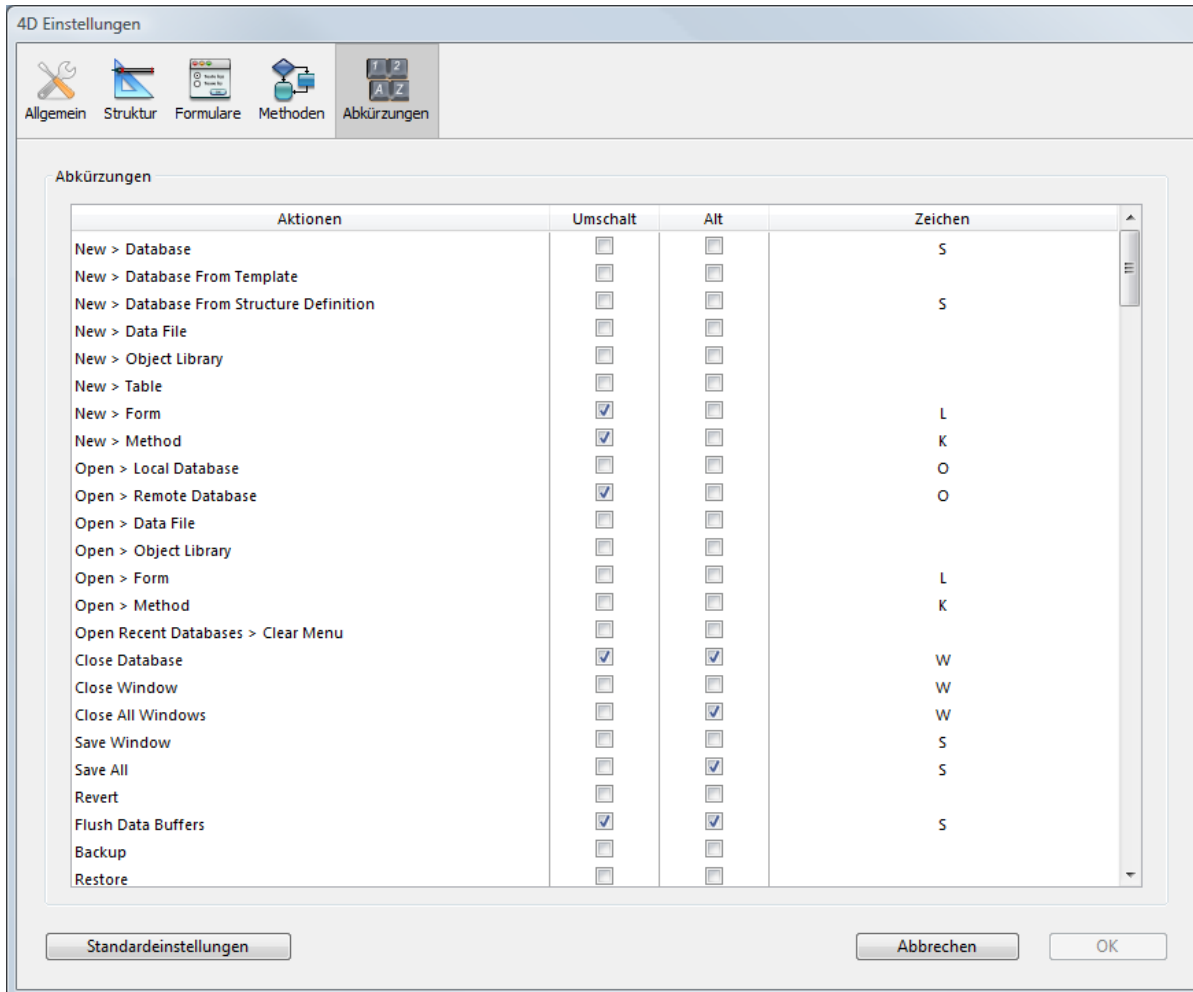
Farben

Hier können Sie die verschiedenen Farben auf der Oberfläche des Methodeneditors festlegen.

- **Zeile mit dem Cursor (aktives Fenster) / Zeile mit dem Cursor (inaktives Fenster):** Hintergrundfarbe der Zeile mit dem Cursor.
- **Hervorhebung der gefundenen Wörter:** Markierungsfarbe für Wörter, die durch eine Suche gefunden werden.
- **Farbe der Hervorhebung der Klammern:** Markierungsfarbe der entsprechenden Klammern. Sie wird verwendet, wenn Klammerpaare durch Hervorhebung angezeigt werden, siehe oben Abschnitt "Optionen".
- **Farbe des Text-Hintergrunds:** Hintergrundfarbe des Fensters Methodeneditor.
- **Farbe des vorgeschlagenen Textes:** Farbe des Textes, den der Methodeneditor zur automatischen Vervollständigung vorschlägt.
- **Hervorhebung der Blöcke** (*neu in v16*): Markierungsfarbe für ausgewählte logische Blöcke, wenn die Option "Logische Blöcke hervorheben" auf der Seite **Optionen** aktiv ist.
- **Hervorhebung des gleichen Textes** (*neu in v16*): Markierungsfarbe für andere Vorkommen desselben Textes, wenn eine Option von "Text hervorheben" auf der Seite **Optionen** aktiv ist.

Seite Abkürzungen

Diese Seite zeigt alle Tastenkombinationen für den Designmodus von 4D, mit Ausnahme von Standardkombinationen des Systems, wie z.B. **Strg+C/Befehl+C** für den Menübefehl **Kopieren**. Um eine Tastenkombination zu ändern, aktivieren/deaktivieren Sie den entsprechenden Eintrag in der Liste (Umschalt, Alt oder Buchstabe). Sie können auch auf die jeweilige Zeile doppelklicken, um sie in einem spezifischen Dialogfenster einzustellen:



Beachten Sie, dass jede Tastenkombination immer die **Strg-Taste** unter Windows, bzw. die **Befehlstaste** auf Mac OS enthält.

Diese Liste basiert auf der Datei *4DShortcutsXX.xml* im Unterordner *4D Extensions* (v12) oder *Resources* (v13). Passen Sie diese Liste im Dialogfenster an, wird diese Datei in den Ordner mit den lokalen Voreinstellungen dupliziert und dann anstelle der Standarddatei verwendet. So bleiben bei einem Update von 4D Ihre eigenen Tastenkombinationen erhalten.

Datenbank-Eigenschaften

-  Überblick
-  Benutzereinstellungen verwenden
-  Seite Allgemein
-  Seite Oberfläche
-  Seite Compiler
-  Seite Datenbank/ Datenspeicherung
-  Seite Datenbank/ Speicher
-  Seite Verschieben
-  Seite Backup/ Planer
-  Seite Backup/ Konfiguration
-  Seite Backup/ Backup & Wiederherstellen
-  Seite Client-Server/Netzwerk
-  Seite Client-Server/ IP Konfiguration
-  Seite Web/ Konfiguration
-  Seite Web/ Optionen (I)
-  Seite Web/ Optionen (II)
-  Seite Web/ Log (Typ)
-  Seite Web/ Log (Backup)
-  Seite Web/ Web Services
-  Seite Web/ REST
-  Seite SQL
-  Seite PHP
-  Seite Sicherheit
-  Seite Kompatibilität

Datenbank-Eigenschaften bestimmen die Funktionsweise der aktuellen Datenbank. Diese Parameter können für jede Datenbank anders sein. Das sind z.B. Auflisten der Ports, Zugriffsrechte zur Designumgebung, SQL Konfigurationen. 4D bietet zwei Modi für diese Einstellungen:

- **Standardmodus:** Damit werden alle Einstellungen in der Strukturdatei der Anwendung gespeichert und in allen Fällen angewendet (Standardeinstellung).
- **Modus "Benutzereinstellung":** Damit werden Teile der eigenen Einstellungen in einer externen Datei gespeichert, die für die Datenbank oder jede Dateidatei anstelle der Struktureinstellungen genutzt wird. Diesen Modus aktivieren Sie über die entsprechende Option auf der [Seite Sicherheit](#).

Dieses Kapitel beschreibt die einzelnen Seiten der Einstellungen im Standardmodus. Weitere Informationen zum Modus "Benutzereinstellung" finden Sie im Abschnitt [Benutzereinstellungen verwenden](#).


Die meisten Eigenschaften werden sofort angewendet. Ein paar davon, wie z.B. die Einstellung der Startumgebung, werden erst wirksam, wenn die Anwendung neu gestartet wurde. In diesem Fall erscheint eine Meldung, dass die Änderung erst beim nächsten Startup ausgeführt wird.

4D Server: Objektsperre tritt ein, wenn zwei oder mehr Benutzer gleichzeitig versuchen, die Datenbank-Eigenschaften zu verändern. Es kann immer nur ein Benutzer zur selben Zeit die Eigenschaften verändern.

Hinweis: 4D bietet einen weiteren Satz Parameter, die für die 4D Anwendung gilt: die Benutzereinstellungen. Weitere Informationen dazu finden Sie im Kapitel [Einstellungen](#).

Zugriff

Das Dialogfenster Datenbank-Eigenschaften können Sie folgendermaßen aufrufen:

- Über das Menü **Design > Datenbank-Eigenschaften**
- Über die letzte Schaltfläche in der 4D Werkzeugleiste 
- Auf 4D Server über das Menü **Bearbeiten > Datenbank-Eigenschaften**

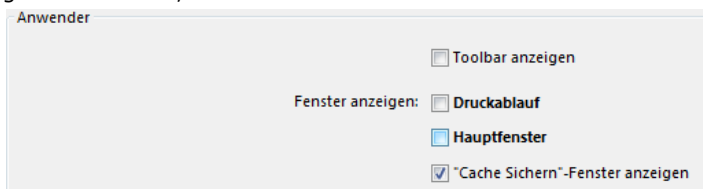
Hinweis: Im Modus "Benutzereinstellung" gibt es für jede Position mehrere Menübefehle:

- **Struktur Einstellungen:** Das entspricht dem Befehl **Datenbank Eigenschaften** im Standardmodus.
- **Benutzer Einstellungen:** Ermöglicht Zugriff auf Einstellungen, die sich extern in eine Benutzerdatei speichern lassen und anstelle der Struktureinstellungen verwendet werden, falls diese verändert wurden.
- **Benutzer Einstellungen für Datendatei:** Ermöglicht Zugriff auf Einstellungen, die sich extern in eine Benutzerdatei speichern lassen, die der aktuellen Datendatei zugeordnet ist und anstelle der Struktureinstellungen verwendet werden, falls diese verändert wurden.

Weitere Informationen dazu finden Sie im Abschnitt [Benutzereinstellungen verwenden](#).

Parameter anpassen und "Standardeinstellungen"

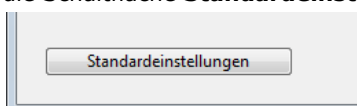
In den Dialogfenstern Einstellungen und Datenbank-Eigenschaften erscheinen die Parameterbezeichnungen, deren Werte sich geändert haben, in **Fett**:



Die Änderung kann im Dialogfenster selbst erfolgt sein, oder bei einer konvertierten Datenbank von einer Voreinstellung kommen, die zuvor verändert wurde.

Ein Parameter erscheint weiterhin in Fettschrift, wenn sein Wert manuell mit dem Standardwert ersetzt wird. Auf diese Weise lässt sich immer nachvollziehen, welche Parameter angepasst wurden.

Um die Parameter auf ihre Standardwerte zurückzusetzen und die Markierung zu entfernen, können Sie am unteren Rand auf die Schaltfläche **Standardeinstellungen** klicken:



Diese Schaltfläche setzt alle Parameter der aktuellen Seite zurück. Sie wird aktiv, wenn mindestens ein Parameter auf dieser Seite geändert wurde.

Benutzereinstellungen verwenden

Sie können eine externe Datei mit eigenen Einstellungen erzeugen. Ist diese Funktionalität aktiviert, werden nicht die Einstellungen in der Strukturdatei, genannt "Struktur-Einstellungen", sondern die Einstellungen in der externen Datei, genannt "Benutzer-Einstellungen", verwendet.

Auf diese Weise können Sie eigene Einstellungen zwischen den Updates Ihrer 4D Programme beibehalten oder je nach dem, wo Ihr 4D Programm eingesetzt wird, unterschiedliche Einstellungen verwalten. Sie können externe Einstellungsdateien im XML-Format auch per Programmierung verwalten.

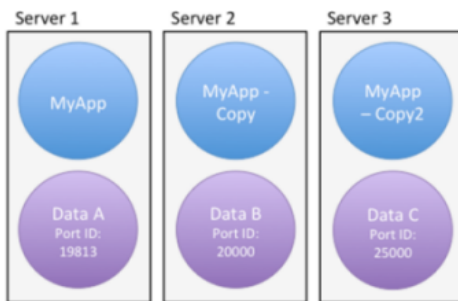
- **Benutzereinstellungen (Standard)**

Diese Benutzereinstellungen werden anstelle der Struktureinstellungen für jede mit der Anwendung geöffnete Datendatei verwendet.

- **Benutzereinstellungen für Datendatei**

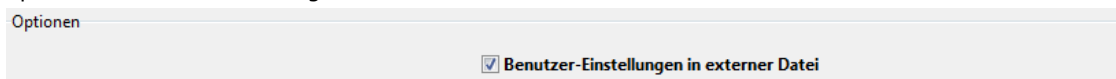
4D kann jeder Datendatei für Ihre Anwendung einen Satz Benutzereinstellungen zuweisen, also z.B. die Portnummer konfigurieren oder den Server Cache.

Mit dieser Option können Sie ganz leicht Kopien derselben Anwendung mit mehreren Datendateien mit jeweils unterschiedlichen Einstellungen weitergeben und aktualisieren. Nehmen wir z.B. folgende Konfiguration: Eine Anwendung ist dupliziert und jede Kopie verwendet eine andere Portnummer. Ist diese Benutzereinstellung mit der Datendatei verknüpft, können Sie die Anwendung direkt aktualisieren, Sie müssen die Portnummer nicht manuell ändern:

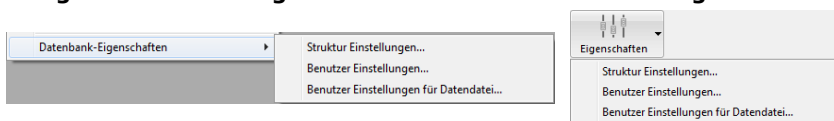


Modus Benutzer Einstellungen aktivieren

Um Benutzer-Einstellungen extern abzulegen, müssen Sie in den Datenbank-Eigenschaften auf der **Seite Sicherheit** die Option "Benutzer-Einstellungen in externer Datei" aktivieren.



Markieren Sie diese Option, werden die Datenbank-Eigenschaften in drei Dialogfenster aufgeteilt: **Struktur Einstellungen**, **Benutzer Einstellungen** und **Benutzer Einstellungen für Datendatei**. Sie können diese Dialogfenster über das Menü **Design>Datenbank-Eigenschaften** oder die Schaltfläche **Eigenschaften** in der Toolbar oben rechts öffnen:

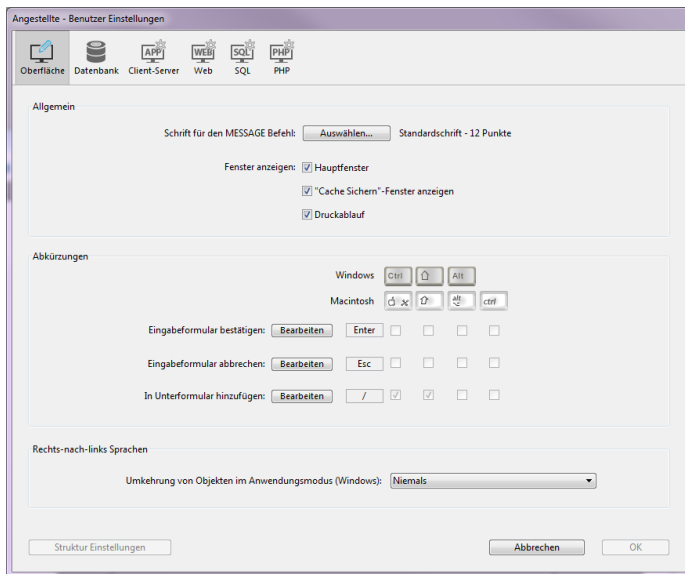


Hinweis: Liegt die Datendatei auf der gleichen Ebene wie die Strukturdatei (Standardkonfiguration beim Erstellen einer Datenbank), erscheint der Menübefehl **Benutzer-Einstellungen in externer Datei...** nicht.

Sie können auf diese Dialogfenster auch mit dem passenden Selector im Parameter *EinstellungenTyp* des Befehls **OPEN SETTINGS WINDOW** zugreifen.

Dialogfenster Benutzer Einstellungen

Ist der externe Modus aktiviert, sind die Datenbank-Eigenschaften über drei Dialogfenster verfügbar: "Struktur-Einstellungen", "Benutzer-Einstellungen" und "Benutzer-Einstellungen für Datendatei". Die Struktureinstellungen sind identisch mit den standardmäßigen Datenbank-Eigenschaften und bieten Zugang zu allen Eigenschaften. Die Benutzer-Einstellungen enthalten eine Auswahl relevanter Eigenschaften, die extern abgelegt werden können:



Nachfolgende Tabelle zeigt, welche Einstellungen verfügbar sind und beschreibt die wichtigsten Unterschiede gegenüber den Standardeinstellungen:

Seite der Datenbank-Eigenschaften

Seite Allgemein

Seite Oberfläche

Seite Compiler

Seite Datenbank/ Datenspeicherung

Seite Datenbank/ Speicher

Seite Verschieben

Seite Backup/ Planer

Seite Backup/ Konfiguration

Seite Backup/ Backup & Wiederherstellen

Seite Client-Server/ Netzwerk

Seite Client-Server/ IP Konfiguration

Seite Web/ Konfiguration

Seite Web/ Optionen (I)

Seite Web/ Optionen (II)

Seite Web/ Log (Typ)

Seite Web/ Log (Backup)

Seite Web/ Web Services

Seite SQL

Seite PHP

Seite Sicherheit

Seite Kompatibilität

Seite der Benutzer-Einstellungen/ Benutzer-Einstellungen für Datendatei

Nicht verfügbar

Identisch zu Standardeinstellungen

Nicht verfügbar

Nicht verfügbar

Identisch zu Standardeinstellungen

Nicht verfügbar

Nicht verfügbar

Nicht verfügbar

Nicht verfügbar

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Option Methodenpräfix nicht verfügbar

Identisch zu Standardeinstellungen

Identisch zu Standardeinstellungen

Nicht verfügbar

Nicht verfügbar

In diesem Dialogfenster bearbeitete Einstellungen werden automatisch in der entsprechenden Datei *settings.4DSettings* gespeichert (siehe unten).

SET DATABASE PARAMETER und Benutzereinstellungen

Die meisten Benutzereinstellungen sind auch über den Befehl **SET DATABASE PARAMETER** verfügbar. Für Benutzereinstellungen ist die Eigenschaft "Wird zwischen 2 Sitzungen beibehalten" auf "Ja" gesetzt.

Ist das Feature "Benutzereinstellungen" aktiviert, werden die vom Befehl **SET DATABASE PARAMETER** bearbeiteten Benutzereinstellungen automatisch in den Benutzereinstellungen für die Datendatei gesichert.

Hinweis: Das gilt nicht für die Konstante Table sequence number; der Wert dieser Einstellung wird immer direkt in der Datendatei gespeichert.

settings.4DSettings files

Markieren Sie in den Datenbank-Eigenschaften die Option **Benutzer-Einstellungen in externer Datei**, werden Dateien mit Benutzer Einstellungen automatisch angelegt. Ihr Speicherort richtet sich nach der definierten Benutzer Einstellungen.

Benutzer-Einstellungen (Standard)

Die standardmäßige Datei mit Benutzereinstellungen wird automatisch erstellt und in einem Ordner [Preferences] an folgender Stelle abgelegt:

<DatabaseFolder>/Preferences/settings.4DSettings

... wobei <DatabaseFolder> der Name des Ordners mit der Strukturdatei der Datenbank ist.

Hinweis: In doppelklickbaren Anwendungen liegt die Datei mit den Benutzereinstellungen an folgender Stelle:

- In Einzelplatzversionen: <DatabaseFolder>/Database/Preferences/settings.4DSettings
- In Client/Server Versionen: <DatabaseFolder>/Server Database/Preferences/settings.4DSettings

Datei settings.4DSettings

Die Datei Benutzereinstellungen, verknüpft mit der Datendatei, wird automatisch erstellt und im Ordner [Preferences] an folgender Stelle abgelegt:

```
<DataFolder>/Preferences/settings.4DSettings
```

... <DataFolder> ist der Name des Ordners mit der aktuellen Datendatei (".4dd") der Anwendung.

Hinweis: Liegt die Datendatei auf der gleichen Ebene wie die Strukturdatei (Standardeinstellung beim Erstellen der Datenbank), nutzen auf Struktur und auf Daten basierende Benutzereinstellungen den gleichen Speicherort und die gleiche Datei. Der Menübefehl **Benutzereinstellungen für Datendatei ...** ist dann nicht verfügbar.

Dateien mit Benutzereinstellungen sind im XML Format. Sie lassen sich über die integrierten 4D XML Befehle oder über einen XML Editor lesen und verändern. Folglich können Sie die Einstellungen per Programmierung verwalten, insbesondere bei Anwendungen mit einkompilierter 4D Volume Desktop. Beachten Sie, dass Änderungen per Programmierung in dieser Datei erst berücksichtigt werden, wenn die Anwendung erneut geöffnet wird.

Priorität der Einstellungen

Datenbankeinstellungen lassen sich auf drei Ebenen speichern. Jede auf einer Ebene definierte Einstellung überschreibt dieselbe Einstellung auf der vorigen Ebene, sofern sie existiert:

Prioritätsebene	Name	Speicherort	Kommentar
3 (am niedrigsten)	Struktureinstellungen (oder Datenbankeinstellungen, wenn die Option "Benutzereinstellungen" nicht aktiviert ist)	In der Strukturdatei (.4db)	Einmaliger Speicherort, wenn keine Benutzereinstellungen aktiviert sind. Wird auf alle Kopien der Anwendung angewandt.
2	Benutzereinstellungen (alle Datendateien)	Externe Datei settings.4DSettings im Ordner Preferences auf der gleichen Ebene wie die Strukturdatei	Überschreibt die Strukturreinstellungen. Wird mit dem Anwendungspaket gespeichert.
1 (am höchsten)	Benutzereinstellungen für die Datendatei (aktuelle Datendatei)	Externe Datei settings.4DSettings im Ordner Preferences auf der gleichen Ebene wie die Datendatei	Überschreibt die Strukturreinstellungen und Benutzereinstellungen. Wird nur angewandt, wenn die verknüpfte Datendatei mit der Anwendung verwendet wird.

Beachten Sie, dass die Dateien *settings.4DSettings* auch nur einen Teil der relevanten Einstellungen enthalten kann, im Gegensatz zur Strukturdatei, die alle eigenen Einstellungen enthält, inkl. Core settings?? die sich nicht in externe Dateien verschieben lassen.

Diese Seite enthält verschiedene Optionen zum Konfigurieren generischer Parameter für die 4D Anwendung.

Design

Dieser Bereich enthält die Option **Toolbar anzeigen**. Ist sie markiert, wird die 4D Werkzeugleiste in der Design-Umgebung angezeigt.

Allgemein

In diesem Bereich definieren Sie verschiedene Optionen zum Starten und zur Funktionsweise der 4D Anwendung.

Startup Umgebung

Über dieses DropDown-Menü wählen Sie den standardmäßigen Startup Modus für die Anwendung: **Design** oder **Anwendung**. 4D öffnet die Datenbank standardmäßig in der Designumgebung, außer das Kennwortsystem ist aktiviert.

Hinweis: Im Öffnen-Dialogfenster können Sie wählen, ob Sie die Anwendung im kompilierten oder im interpretierten Modus öffnen (siehe **Lokale Anwendung öffnen**).

Aktiviere automatische Kommentare

Diese Option aktiviert und setzt in Verbindung mit dem darunterliegenden Bereich das automatische Kommentarsystem in Ihrer Datenbank. Weitere Informationen dazu finden Sie im Abschnitt **Kommentare verwenden**.

Hinweis zur Kompatibilität: Ab 4D Version 12 werden alle Kommentare, die im Kopfteil einer Methode mit vorangestellten // gesetzt werden, als Hilfetipps angezeigt, wenn in einer anderen Methode auf diese Methode verwiesen wird (siehe **Hilfetipps verwenden**). Diese Funktion ist besonders hilfreich zum Dokumentieren von Benutzermethoden. Sie ist jedoch nicht kompatibel mit Kommentaren, die im Explorer gesetzt werden. Wollen Sie Methodenkopfteile für Dokumentationsbereiche verwenden, aktivieren Sie nicht die automatischen Kommentare.

CPU Priorität

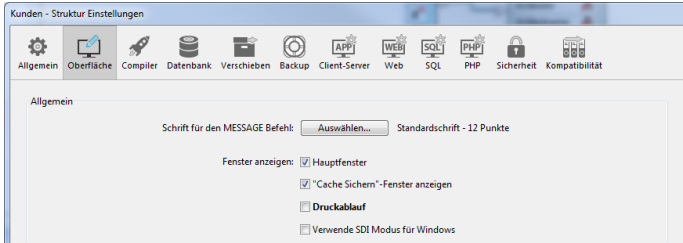
Hinweis zur Kompatibilität: Diese Einstellung ist jetzt überholt. Dieser Bereich erscheint nur in konvertierten Anwendungen, wo ein eigener Wert gesetzt wurde. Wird er angezeigt, empfehlen wir, auf die Schaltfläche **Standardeinstellungen** zu klicken, um diese Parameter zurückzusetzen und aus dem Dialogfenster zu entfernen.

Seite Oberfläche

Auf der Seite Oberfläche setzen Sie verschiedene Anzeigoptionen für die Oberfläche der Datenbank.

Allgemein

In diesem Bereich definieren Sie verschiedene Optionen zur Anzeige der Anwendung.



Schrift für den MESSAGE Befehl

Über die Schaltfläche **Auswählen** können Sie Schrifttyp und -größe von Zeichen setzen, die der Befehl **MESSAGE** verwendet. Standardschrift und -größe richten sich nach der Plattform, auf der 4D läuft.

Hinweis: Diese Eigenschaft betrifft auch folgende Bereiche von 4D:

- Einige Vorschaubereiche des **Explorer**
- Lineal des **Formulareditor**

Andere Optionen definieren die Anzeige bestimmter Fenster im Anwendungsmodus:

- **Hauptfenster:** Hier können Sie die Anzeige des Startbildschirms der aktuellen Menüleiste im Anwendungsmodus aktivieren oder deaktivieren (siehe **Menüleisten verwalten**). Ist dieses Fenster ausgeblendet, können Sie die Anzeige aller Fenster per Programmierung steuern, z.B. über die **Datenbankmethode On Startup**.
- **„Cache-Sichern“-Fenster anzeigen:** Ist diese Option markiert, erscheint beim Schreiben des Daten-Cache auf die Festplatte unten links ein kleines Fenster. Diese Operation blockiert kurzzeitig die Aktionen des Benutzers. Die visuelle Anzeige informiert, dass der Cache gerade gesichert wird:



Hinweis: Sie können die Zeitspanne für Sichern des Cache in den Datenbank-Eigenschaften auf der **Seite Datenbank/Speicher** einstellen.

- **Druckablauf:** Damit können Sie die Anzeige des Druckverlaufs aktivieren oder deaktivieren.
 - **Verwende SDI Modus für Windows:** Ist diese Option markiert, aktiviert 4D automatisch den SDI Modus (Single-Document Interface) in Ihrer doppelklickbaren Anwendung, wenn sie in einem unterstützten Kontext ausgeführt wird (siehe **SDI Modus unter Windows**).
- Hinweis:** Diese Option lässt sich auch auf macOS auswählen, wird aber beim Ausführen auf dieser Plattform ignoriert.

Abkürzungen

In diesem Bereich können Sie die standardmäßig vorgegebenen Tastenkürzel für drei Grundoperationen von 4D ansehen und verändern: Eingabeformular bestätigen, Eingabeformular abrechnen und In Unterformular hinzufügen. Die Tastenkombinationen sind für beide Plattformen gleich. Die Tastensymbole geben die jeweiligen Tasten für Windows und Mac OS an.

Es gibt folgende Standardeinstellungen:

- Eingabeformular bestätigen: **Enter**
- Eingabeformular abrechnen: **Esc**
- In Unterformular hinzufügen: **Strg- + Umschalttaste + /** (Windows) oder **Befehls- + Umschalttaste + /** (Mac OS)

Wollen Sie die Tastenkombination für eine Operation ändern, klicken Sie auf die Schaltfläche **Ändern**. Auf dem Bildschirm erscheint folgendes Fenster:



Um eine Tastenkombination zu ändern, geben Sie die neue Kombination auf Ihrer Tastatur ein und klicken auf die Schaltfläche **OK**. Wollen Sie einer Operation keine Tastenkombination zuweisen, klicken Sie auf die Schaltfläche **Löschen**.

Rechts-nach-links Sprachen

Der Begriff „Rechts-nach-links Sprachen“ bezieht sich auf Sprachen, die von rechts nach links gelesen werden, wie z.B. Arabisch.

4D bietet unter Windows erweiterten Support für Sprachen, die von rechts nach links gelesen werden. Aktivieren Sie den Modus Rechts-nach-links, wird die Oberfläche der 4D Datenbank (Formulare und Menüs) automatisch in der Anwendungsumgebung invertiert. Das gilt nur für Windows. Es gibt keine entsprechende Funktion auf Mac OS X.

Die Umstellung betrifft nicht nur die umgekehrte Ausrichtung von Textobjekten, sondern auch grafische Objekte, ihre Position innerhalb von Formularen, Beschriftung von Fenstertiteln und Schaltflächen, die Anordnung der Menüs, etc.

Hinweis: In den aktuellen Versionen von 4D ist der invertierte Modus nicht zum Drucken von Formularen verfügbar.

Mit der Option **Umkehrung von Objekten im Anwendungsmodus (Windows)** können Sie den Modus Rechts-nach-links generell für die Datenbank aktivieren und konfigurieren. Sie können zwischen 3 Optionen wählen:

- **Nie:** Ist diese Option gewählt, wechselt die Datenbank nie in den Modus "Rechts-nach-links", selbst wenn das Betriebssystem in diesem Modus konfiguriert ist. Diese Option ist standardmäßig ausgewählt.
- **Automatisch:** Ist diese Option gewählt, wechselt die Datenbank automatisch in den Modus "Rechts-nach-links", wenn das Betriebssystem in diesem Modus konfiguriert ist. Das ist z.B. bei der arabischen Version von Windows der Fall.
- **Immer:** Ist diese Option gewählt, wechselt die Datenbank sofort in den Modus "Rechts-nach-links", selbst wenn das Betriebssystem im gängigen Modus, also „Links-nach-rechts“, konfiguriert ist.

Ändern Sie die Option, wird sie auf die Datenbank angewendet, sobald Sie das Dialogfenster bestätigen.

Hinweise:

- Diese globale Einstellung lässt sich mit dem 4D Befehl **SET DATABASE PARAMETER** auch per Programmierung steuern.
- Ist der Modus "Rechts-nach-links" aktiviert, können Sie ihn für jedes Formular lokal deaktivieren (siehe "Objekte nicht umkehren (Windows)" unter **Formulareigenschaften**).
- Standardmäßig kehrt der Modus "Rechts-nach-links" den Inhalt von Bildern in Formularen nicht um. Um Bilder bei Bedarf umzukehren, gibt es die zusätzliche Option **Spiegeleffekt** (siehe "Spiegeleffekt für Bilder (Windows)" unter **Statische Bilder verwenden**).

Auf dieser Seite definieren Sie die Parameter zum Kompilieren einer Datenbank. Weitere Informationen dazu finden Sie im Kapitel [Kompilieren](#).

Optionen zum Kompilieren

Dieser Bereich enthält allgemeine Optionen, die beim Kompilieren verwendet werden.

- **Bereichsüberprüfung:** Mit dieser Option erfolgt eine zusätzliche Überprüfung, die den Code zur Laufzeit und abhängig vom Status der Datenbankobjekte in einem bestimmten Moment überprüft. Weitere Informationen dazu finden Sie im Abschnitt [Bereichsprüfung](#).
- **Erzeuge Symboldokument:** Diese Option generiert eine Textdatei vom Typ ASCII (text only), welche die Variablen mit Typ und Methode, aus der sie stammen, auflistet. Die Liste enthält auch die Methoden und Funktionen mit den dazugehörigen Parametertypen und Ergebnissen (sofern vorhanden). Die Datei liegt im gleichen Ordner wie die Strukturdatei der Datenbank. Sie lautet: *DatabaseName_symbols.txt*. Weitere Informationen dazu finden Sie im Abschnitt [Symboldatei](#).
- **Erzeuge Fehlerdokument:** Diese Option erzeugt während der Syntaxprüfung ein Fehlerdokument. Es zeigt allgemeine Fehler, in spezifischen Zeilen auftretende Fehler sowie Warnungen. Generell sind alle Fehler, die der Compiler anzeigt, automatisch im Menü **Methode** verfügbar. Ein Fehlerdokument ist jedoch hilfreich, denn es lässt sich leicht auf andere Rechner übertragen, insbesondere wenn mehrere Entwickler in einer Client/Server-Umgebung zusammenarbeiten. Das Fehlerdokument wird im XML Format erstellt, um das automatische Analysieren (Parsen) seines Inhalts zu erleichtern. Damit können Sie auch eigene Formate zum Anzeigen der Fehler erstellen. Das Fehlerdokument erhält den Namen *DatabaseName_errors.xml* und liegt neben der Strukturdatei der Datenbank. Weitere Informationen dazu finden Sie im Abschnitt [Fehlerdokument](#).
- **Mehrfach Kompilierung (32-Bit und 64-Bit):** Ist diese Option nicht markiert (Standardeinstellung), wird nur der kompilierte Code der aktuellen Architektur (32-Bit oder 64-Bit) erzeugt. Ist diese Option markiert, enthalten die Dateien .4DC und .4DB den kompilierten Code für 32-Bit und für 64-Bit. Sie können dann entweder mit einer 4D Applikation in 32-Bit oder in 64-Bit laufen. Beachten Sie, dass:
 - die 4D Server Version in 64-Bit für Windows und für Mac OS verfügbar ist
 - ab 4D v15 R2 die 4D Version in 64-Bit als Vorabversion verfügbar ist.
 - Nur 4D Datenbanken, die in Unicode Modus konvertiert wurden, sich für 64-Bit kompilieren lassen.
- **Lokale Variablen initialisieren:** Diese Option definiert den Initialisierungsmodus der lokalen Variable zu Beginn der Methoden:
 - **Auf 'Null':** Variablen werden standardmäßig auf Null gesetzt (leerer String für Zeichenketten, 0 für Ziffern...).
 - **Auf zufälligen Wert:** Der Compiler weist Variablen je nach Typ einen bestimmten Wert zu (1919382119 für Lange Ganzzahl, "rgrg" für Zeichenketten, True für Boolean, ...). Mit dieser Option können Sie lokale Variablen abstecken, die Sie vergessen haben, zu initialisieren.
 - **Nicht:** Der Compiler initialisiert die Variablen nicht. So gewinnen Sie Zeit beim Ausführen der Datenbank. Das setzt allerdings eine korrekte Initialisierung voraus.
- **Kompilationspfad:** Diese Option definiert die Anzahl der Durchgänge, die der Compiler ausführt und somit die Dauer der Kompilierung.
 - **Variablen typisieren:** Durchläuft alle Stadien der Typisierung, die für die Kompilierung erforderlich sind.
 - **Prozess- und Interprozessvariablen sind typisiert:** Der Durchlauf zum Typisieren von Prozess- und Interprozessvariablen wird nicht ausgeführt. Verwenden Sie diese Option, wenn Sie alle Prozess- und Interprozessvariablen bereits selbst oder über die Funktion „Compilermethoden automatisch generieren“ typisiert haben.
 - **Alle Variablen sind typisiert:** Der Durchlauf zum Typisieren von lokalen Variablen, Prozess- und Interprozessvariablen wird nicht ausgeführt. Verwenden Sie diese Option, wenn Sie sicher sind, dass alle Prozess-, Interprozess- und lokalen Variablen eindeutig typisiert wurden.

Standard Typisierung

In diesem Bereich setzen Sie den Standardtyp für zweideutige Datenbankobjekte.

- **Numerisch:** Diese Option sorgt für eine eindeutige Zahlentypisierung, entweder als **Zahl** oder als **Lange Ganzzahl**. Sie hat keinen Vorrang gegenüber Direktiven, die u.U. in Ihrer Datenbank festgelegt wurden. Sie können das Laufen Ihrer Datenbank optimieren, wenn Sie den Typ Lange Ganzzahl wählen.
- **Schaltflächen:** Diese Option sorgt für eine eindeutige Typisierung von Schaltflächen, entweder als **Zahl** oder als **Lange Ganzzahl**. Sie hat keinen Vorrang gegenüber Direktiven, die u.U. in Ihrer Datenbank festgelegt wurden. Sie gilt für Standard Schaltflächen und für folgende Objekte: Kontrollkästchen, 3D Kontrollkästchen, hervorgehobene Schaltflächen, Unsichtbare Schaltflächen, 3D Schaltflächen, Bildschaltflächen, Schaltflächengitter, Optionsfelder, 3D Optionsfelder, Bild Optionsfelder, Bild PopUp-Menüs, hierarchische PopUp-Menüs und PopUp/DropDown-Listen.

Compilermethoden für ...

In diesem Bereich können Sie Compilermethoden umbenennen, die der Compiler automatisch generiert. Diese Methoden fassen alle Deklarationen zum Typisieren von Variablen, Prozess- und Interprozess-Arrays sowie von lokalen Variablen

zusammen. Sie werden im Kompilierungsfenster generiert. Weitere Informationen dazu finden Sie im Abschnitt **Compiler Fenster**.

Es lassen sich bis zu 5 Compilermethoden generieren; die Methode wird nur angelegt, wenn die Datenbank die entsprechenden Einträge enthält:

- **Variablen**: Gruppiert Compiler-Deklarationen von Prozessvariablen
- **Interprozess-Variablen**: Gruppiert Compiler-Deklarationen von Interprozessvariablen
- **Arrays**: Gruppiert Compiler-Deklarationen von Prozess Arrays
- **Interprozess-Arrays**: Gruppiert Compiler-Deklarationen von Interprozess Arrays
- **Methoden**: Gruppiert Compiler-Deklarationen von lokalen Variablen, die Parameter angeben, z.B. **C_LONGINT**(MeineMethode;\$1).

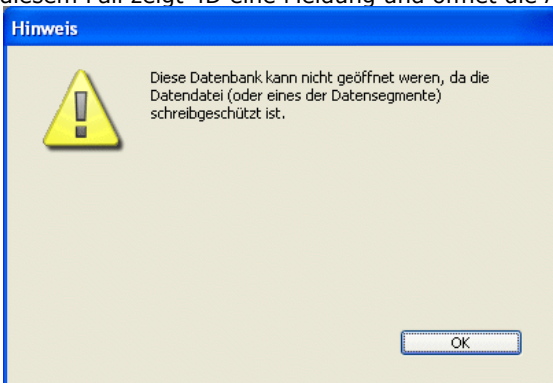
Sie können die Methoden im jeweiligen Eingabebereich umbenennen.

Sie beginnen jedoch immer mit dem fest vorgegebenen Eintrag "Compiler_". Der Methodenname inkl. Vorsilbe muss einmalig sein und kann max. 31 Zeichen enthalten. Zeichen aus dem erweiterten ASCII Code (Akzente, Symbole) sowie Leerzeichen sind nicht zulässig.

Auf dieser Seite konfigurieren Sie die Datenspeicherung auf der Festplatte für die 4D Anwendung.

Allgemeine Einstellungen

- **Keine Warnung anzeigen, wenn Struktur im Nur-Lesen Modus:** Öffnen Sie eine Datenbank, deren Strukturdatei im Nur-Lesen Modus ist, erscheint eine Meldung. Öffnen Sie die Datenbank in diesem Status, werden die Änderungen an der Strukturdatei nicht gesichert. Diese Meldung kann in bestimmten Fällen unerwünscht sein, z.B. bei Datenbanken auf CD-ROM, die nur zur Konsultation dienen. Markieren Sie für solche Fälle diese Option.
- **Erlaube "Nur-Lesen"-Status für Datendatei:** Mit dieser Option können Sie die Arbeitsweise der Anwendung einstellen, wenn auf Systemebene eine gesperrte Datendatei geöffnet wird. 4D verhindert automatisch das Öffnen einer Anwendung, wenn die Datendatei bzw. ein Segment davon gesperrt ist. In diesem Fall zeigt 4D eine Meldung und öffnet die Anwendung nicht:



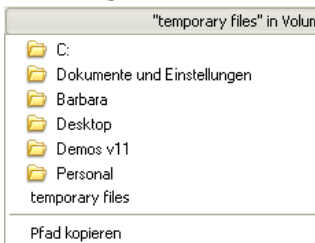
Nur wenn diese Option markiert ist, wird die Anwendung auch bei gesperrter Datendatei geöffnet. Das ist die Standardeinstellung für 4D Anwendungen.

Über gesperrte Dateien: Sie können gesperrte Dateien lesen, jedoch nicht verändern. Dateien sind z.B. gesperrt, wenn sie auf einem nicht beschreibbaren Datenträger, z.B. einer CD-Rom liegen bzw. von dort kopiert werden. 4D arbeitet in transparenter Weise mit gesperrten Datendateien, so dass Sie auch auf CD-Rom gespeicherte Anwendungen verwenden können. Das Risiko dabei ist jedoch, dass die gesperrte Datei unbeabsichtigt verändert und nicht gesichert wird. Deshalb erlaubt 4D standardmäßig nicht, Anwendungen mit einer gesperrten Datendatei zu öffnen.

Temporärer Ordner

Hier können Sie die Position des Ordners mit temporären Dateien verändern, die angelegt werden, wenn 4D läuft. 4D verwendet diesen Ordner bei Bedarf, um die Daten vorübergehend im Speicher auf der Festplatte abzulegen.

Die aktuelle Position des Ordners erscheint in der DropDown-Liste neben *Aktuell*. Klicken Sie in die Liste, um den vollständigen Pfadnamen anzuzeigen und bei Bedarf zu kopieren:



Es gibt drei Optionen:

- **System:** Ist diese Option markiert, werden die temporären Ordner von 4D in einem Ordner an der von Windows bzw. Mac OS vorgegebenen Stelle abgelegt. Mit der 4D Funktion **Temporary folder** können Sie herausfinden, welcher Ort in Ihrem System definiert wurde. Die Dateien werden in einen Unterordner gelegt mit dem Namen der Datenbank und einer einmaligen Kennung.
- **Ordner der Datendatei** (Standardoption): Ist diese Option markiert, werden die temporären Ordner von 4D in einen Ordner mit Namen „temporary files“ auf derselben Ebene wie die Datendatei der Datenbank gelegt.
- **Benutzerdefiniert:** Verändern Sie diese Einstellung, müssen Sie die Datenbank neu starten, damit die neue Option berücksichtigt wird. 4D prüft, ob sich der gewählte Ordner im Schreibmodus öffnen lässt. Ist dies nicht der Fall, versucht das Programm andere Optionen, bis ein gültiger Ordner gefunden wird.

Hinweis: Diese Option wird in den „Extra Eigenschaften“ der Struktur gespeichert, die verfügbar ist, wenn die Strukturdefinition in XML exportiert wird. Weitere Informationen dazu finden Sie im Abschnitt (siehe **Strukturdefinitionen exportieren und importieren**).

Text-Vergleiche

Ändern Sie eine dieser Optionen, müssen Sie die Anwendung beenden und erneut öffnen, damit die Änderungen wirksam werden. Beim erneuten Öffnen werden alle Indizes der Datenbank automatisch neu indiziert.

- **@ nur am Anfang oder Ende eines Textes als Joker betrachten:** Mit dieser Option definieren Sie, wie das Zeichen "@" w (auch „at“ Zeichen oder Joker genannt) beim Suchen oder bei Vergleichen von Zeichenketten interpretiert wird, wenn es in einem Wort vorkommt.
Diese Option ist standardmäßig inaktiv, d.h. das Joker-Zeichen dient als Platzhalter. Beispiel: B@n findet alle Wörter, die mit 'B' beginnen, mehrere Buchstaben haben und mit 'n' enden (siehe **Jokerzeichen (@)**).
Ist diese Option markiert, gilt @ als ein normales Zeichen, wenn es innerhalb einer Zeichenkette gefunden wird. Das ist besonders hilfreich beim Suchen von E-Mail Adressen, die ja das Zeichen @ enthalten. Diese Option beeinflusst Such- und Sortierläufe, String-Vergleiche sowie in Tabellen gespeicherte bzw. im Speicher gefundene Daten, z.B. Arrays. Auch in Feldern und Variablen vom Typ Alpha (indiziert oder nicht) sowie Text spielt es eine Rolle, wie das Zeichen @ in Such- und Sortierläufen interpretiert wird.
Hinweise:
 - Beachten Sie bei Suchläufen, dass "@" weiterhin als Joker gewertet wird, wenn das Suchkriterium mit @ beginnt oder endet. 4D interpretiert das Zeichen nur innerhalb eines Wortes unterschiedlich, z.B. bill@cgi.com.
 - Diese Option beeinflusst auch das Verhalten von 4D Befehlen unter dem Thema **Objekte (Formulare)**, die den Joker im Parameter *Objekt* zulassen.
 - Zur Wahrung der Sicherheit können diese Einstellung nur der Administrator oder Designer der Datenbank verändern.
- **Aktuelle Datenbanksprache:** Hier legen Sie die Sprache zum Bearbeiten und Vergleichen von Zeichenketten fest. Die Sprachwahl wirkt sich direkt auf Sortieren und Suchen von Text und auf Zeichen aus, jedoch nicht auf die Übersetzung von Text bzw. die Formate für Datum, Zeit und Währung. Diese bleiben in der Sprache des Systems. 4D verwendet standardmäßig die Systemsprache.
Eine 4D Datenbank kann also in einer anderen Sprache laufen, als das System vorgibt. Beim Öffnen der Datenbank prüft die 4D Engine die von der Datendatei verwendete Sprache und gibt sie an die Programmiersprache weiter (interpretierter oder kompilierter Code). Textvergleiche werden dann in derselben Sprache ausgeführt, unabhängig ob sie von der Datenbank-Engine oder der Programmiersprache ausgeführt werden.
Hinweis: Sie können diese Option in den 4D Einstellungen verändern (siehe unter **Seite Allgemein**). In diesem Fall gibt die Option für Datenbanken, die 4D neu erstellt.
- **Nur nicht alphanumerische Zeichen als Schlüsselwörter ansehen:** Diese Option ändert den Algorithmus, den 4D zum Erkennen von Schlüsselwort-Trennern und beim Erstellen von Indizes verwendet. Diese Option ist standardmäßig nicht markiert, d.h. 4D verwendet einen ausgeklügelten Algorithmus, der linguistische Eigenheiten berücksichtigt. Er ähnelt dem Algorithmus in Textverarbeitungsprogrammen, um die Grenzen beim Auswählen eines doppelgeklickten Wortes zu bestimmen. Weitere Informationen dazu finden Sie unter <http://userguide.icu-project.org/boundaryanalysis>.
Ist diese Option markiert, verwendet 4D einen vereinfachten Algorithmus. In dieser Konfiguration werden alle nicht-alphanumerischen Zeichen, also weder Buchstabe noch Nummer, als Schlüsselwort-Trenner gewertet. Diese Einstellung wird für spezifische Anforderungen bei bestimmten Sprachen, wie z.B. Japanisch, benötigt.

Unterstützung von Mecab (japanische Version)

4D unterstützt auf dem japanischen System die Mecab Library, die indizierende Algorithmen für Schlüsselwörter enthält, die sich speziell für die japanische Sprache eignen.

Dieser neue Algorithmus wird in der japanischen Version von 4D standardmäßig verwendet. Die erforderlichen Dateien für die Mecab Library werden für 4D Applikationen im Ordner **mecab** des Ordners **Resources** installiert (nur japanische Version). Sie können bei Bedarf die Verwendung von Mecab Algorithmen deaktivieren und die konventionelle ICU Library verwenden. Gehen Sie dazu in die Datenbank-Eigenschaften auf die Seite **Datenbank > Datenspeicherung** und markieren die Option **Nur nicht alphanumerische Zeichen als Schlüsselwörter ansehen**.

Aktuelle Datenbanksprache:

Nur nicht alphanumerische Zeichen als Schlüsselwörter ansehen

Hinweis: Sie können die Verwendung von Mecab auch deaktivieren, wenn Sie den Ordner **Resources/mecab** in Ihrer japanischen 4D Applikation löschen oder umbenennen.

Über die Optionen dieser Seite konfigurieren Sie den Cache-Speicher für die Datenbank.

Datenbank Cache Einstellungen

- **Cache automatisch anpassen:** Ist diese Option markiert, wird der Cache-Speicher vom System dynamisch verwaltet – innerhalb der von Ihnen gesetzten Grenzen. Die Größe des Cache-Speichers wird gemäß den Einstellungen dynamisch berechnet. Die vorgegebenen Werte entsprechen der standardmäßigen Verwendung von 4D und sind für die meisten Konfigurationen ausreichend.

Datenbank Cache Einstellungen

Cache automatisch anpassen

Physikalischen Speicher für andere Programme und System reservieren: MB

Prozentsatz des verfügbaren Speichers für Cache: %

Mindestgröße: MB

Maximale Größe: MB

- **Physikalischen Speicher für andere Programme und System reservieren:** Anteil des RAM-Speichers, der für das System und andere Programme zu reservieren ist. Dieser Wert wird zur Optimierung erhöht, wenn gleichzeitig mit 4D noch andere Programme auf dem Rechner laufen.
- **Prozentsatz des verfügbaren Speichers für Cache:** Prozent des verbleibenden Speichers, der standardmäßig dem Cache zugewiesen wird.
Um diese Größe zu bestimmen, führen Sie folgende Rechnung aus: (Tatsächlicher Speicher – Zu reservierender tatsächlicher Speicher) X Prozent des Speichers für den Cache. Im anpassbaren Modus variiert die Größe des Cache-Speichers je nach den Erfordernissen der Programme und des Systems. Mit folgenden Optionen können Sie Einschränkungen setzen:
- **Maximale Größe:** Maximale Größe des Speichers, die der Cache verwendet. Bei einer 32-bit Version von 4D und 4D Server ist dieser Wert aus technischen Gründen intern begrenzt (2 GB auf Windows und 1,5 GB auf Mac). Bei einer 64-bit Version von 4D ist dieser Wert virtuell unbegrenzt.
- **Mindestgröße:** Mindestgröße des Speichers, die für den Cache reserviert werden muss. Der Wert kann nicht unter 4 MB sein.

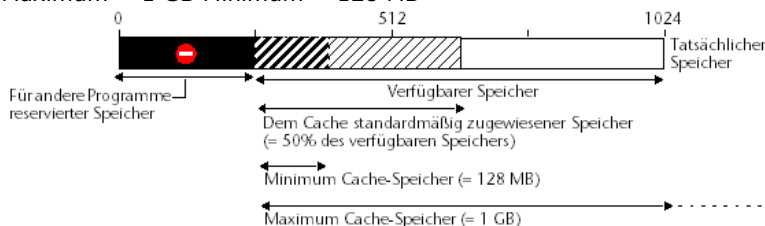
Das Einrichten von Grenzen ist besonders hilfreich für Datenbanken, die auf Rechner verteilt werden, deren Speicherkonfiguration Sie nicht kennen. So können Sie in jedem Fall eine Mindestleistung gewährleisten. Nachfolgendes Schema veranschaulicht die Speichereinteilung:

Berechnungsbeispiel für Cache-Speicher:

Reserviert für tatsächlichen Speicher = 256 MB

Prozent des für den Cache verfügbaren Speichers = 50%

Maximum = 1 GB Minimum = 128 MB



Konfigurationsbeispiele im anpassbaren Modus

die folgenden Beispiele zeigen die Vorteile des anpassbaren Cache bei verschiedenen Software- und Speicherkonstellationen.

- Rechner mit 768 MB RAM, auf dem ein 4D Server läuft. In diesem Fall ist es sinnvoll, dem Cache die maximale Größe zuzuweisen:
 - Physikalisch zu reservierender Speicher = 256 MB.
 - Prozentsatz des verfügbaren Speichers für Cache = 100%, da es nur ein 4D Server Programm gibt.
 - Mindestgröße = 10 MB und Maximale Größe = 1GB (in dieser Konfiguration sind diese Werte unbedeutend).
 --> Standard Cache Größe: $(768 - 256) \times 100\% = 512$ MB
- Dieselbe Konstellation wie oben, jedoch mit drei 4D Server Programmen auf dem Rechner. Sie wollen jedem Programm dieselbe Cache-Größe zuweisen.
Dazu müssen Sie den "Prozentsatz des verfügbaren Speichers für Cache" pro Server auf 33% setzen.
--> Standard Cache Größe pro Programm: $(768 - 256) \times 33\% = 170$ MB
- Erstellen eine ausführbaren Anwendung zum Vertreiben mit unterschiedlichen Konfigurationen. Nach dem Testen wurde herausgefunden, dass die Anwendung am besten mit einem Cache zwischen 20 und 100 MB operiert.
 - Physikalisch zu reservierender Speicher = 256 MB.
 - Prozentsatz des verfügbaren Speichers für Cache = 50%, um Speicher für andere Programme übrig zu lassen.
 - Mindestgröße = 20 MB und Maximale Größe = 100 MB.
 --> Auf einem Rechner mit 256 MB RAM verwendet 4D den minimalen Cache = 20 MB
 --> Auf einem Rechner mit 1 GB RAM, verwendet 4D den maximalen Cache = 100 MB
 --> Auf einem Rechner mit 384 MB RAM verwendet 4D $(384 - 256) \times 50\% = 64$ MB

- **Cache automatisch anpassen ist nicht markiert:** In diesem Modus definieren Sie selbst die Größe des Cache-Speichers. 4D zeigt dann einen Eingabebereich, in dem Sie selbst den Cache-Speicher festlegen, sowie Informationen zum tatsächlichen Speicher (auf dem Rechner verfügbarer RAM-Speicher), dem aktuellen Cache und dem Cache nach erneutem Hochfahren (berücksichtigt Ihre Änderungen) eintragen.

Cache automatisch anpassen

Größe: MB

Die Größe des vorgegebenen Cache-Speichers wird für die 4D Datenbank reserviert, unabhängig von den für den Rechner verfügbaren Ressourcen. Diese Einstellung lässt sich in bestimmten Konfigurationen nutzen, oder wenn die Datenbank in Systemen mit unterschiedlichem Speicher verwendet werden soll. In den meisten Fällen bietet der anpassbare Cache eine höhere Leistung.

- **Cache bei 4D Server und 4D Volume Desktop im physikalischen Speicher halten**

Diese Option erzwingt, dass der Cache im physikalischen Speicher (RAM) des Rechners liegt, wenn die Datenbank von 4D Volume Desktop oder 4D Server ausgeführt wird.

Der virtuelle Speicher ist ein Bereich, der auf der Festplatte eingerichtet wird. Da diese Funktionsweise in bestimmten Fällen die Performance von Anwendungen, die mit 4D Server oder 4D Volume Desktop laufen, signifikant beeinträchtigen kann, können Sie diese Option markieren, damit der Cache dauerhaft im tatsächlichen Speicher erhalten bleibt. Dafür muss natürlich die Konfiguration des Rechners, d.h. Größe des RAM-Speichers, Anzahl der gleichzeitig laufenden Programme, etc. an diese Funktionsweise angepasst sein.

Hinweis: Diese Option beeinflusst nicht die Arbeitsweise von Anwendungen, die mit 4D laufen.

- **Daten sichern alle ... Sekunden/Minuten:** Gibt die Zeitspanne zwischen dem Sichern des Daten-Cache an, z.B. dem Schreiben auf die Festplatte.

4D speichert Ihre Daten in regelmäßigen Abständen, Sie können eine beliebige Zeitspanne zwischen 1 und 500 Minuten festlegen. Standardmäßig sind 20 Sekunden vorgegeben. Das Programm speichert Ihre Daten auch jedes Mal, wenn Sie in eine andere Umgebung wechseln oder das Programm verlassen. Sie können auch den Befehl **FLUSH CACHE** aufrufen, um das Speichern jederzeit auszulösen.

Planen Sie die Eingabe großer Datenmengen, sollten Sie ein kurzes Zeitintervall zwischen den Sicherungen definieren. Sollte der Rechner ausfallen, verlieren Sie nur die Daten, die Sie nach der letzten Sicherung eingegeben haben. (wenn die Datenbank ohne Logbuch läuft).

Verlangsamt die Sicherung eines großen Cache die Aktivität Ihres Rechners, sollten Sie ebenfalls ein kürzeres Intervall einstellen. So werden weniger Datensätze gesichert und der Rechner ist schneller wieder verfügbar.

4D zeigt standardmäßig unten links ein kleines Fenster, wenn der Cache auf die Festplatte geschrieben wird. Wollen Sie diese visuelle Anzeige nicht, können Sie auf der **Seite Oberfläche** die Option "**Cache Sichern**"-Fenster deaktivieren.

Auf der Seite Verschieben der **PICTURE LIBRARY LIST** können Sie die Optionen zum Bewegen von Objekten der Datenbank im Designmodus definieren. 4D wendet diese Einstellungen auf Objekte an, die per Drag-and-Drop oder durch Kopieren/Einsetzen in diese Datenbank gesetzt werden, wenn diese als Zieldatenbank verwendet wird.

Standard-Aktion fürs Kopieren von abhängigen Objekten

Diese Optionen bestimmen das Bewegen von abhängigen Objekten. Das sind die Objekte, die mit den zu bewegendenden Formularen verknüpft sind (siehe **Überblick**). Sie können für jeden abhängigen Objekttyp eine Aktion setzen.

Diese Standardaktionen werden automatisch angewendet, wenn beim Bewegen von Objekten keine Konflikte auftreten und die Option **Nur bei Namenskonflikten** ausgewählt ist (siehe nächster Absatz). Sonst werden sie standardmäßig im *Kopieren-Dialog* aufgerufen.

Die Optionen **Ignorieren**, **Erzeugen (bei Bedarf umbenennen)**, **Erzeugen (bei Bedarf ersetzen)** sowie die Aktionen für **Anderes Objekt verwenden** sind für jeden Objekttyp verfügbar. Andere kontextbezogene Aktionen sind verfügbar, wenn das **Dialogfenster Bewegen** angezeigt wird. Es folgt die Beschreibung dieser Optionen:

- **Ignorieren**: Ist dieser Typ zugewiesen, wird das abhängige Objekt nie in die Zieldatenbank kopiert. Im *Kopieren-Dialog* erscheint standardmäßig die Aktion **Nicht erzeugen**.
- **Erzeugen (Bei Bedarf umbenennen)**: Ist dieser Typ zugewiesen, wird das abhängige Objekt immer in die Zieldatenbank kopiert. Im *Kopieren-Dialog* erscheint standardmäßig die Aktion **Erzeugen**, wenn das Objekt in der Zieldatenbank noch nicht existiert.
Bei einem Namenskonflikt mit einem Objekt in der Zieldatenbank erhält der Name des kopierten Objekts eine Nummer, die bei Bedarf hochgezählt wird. In diesem Fall erscheint im *Kopieren-Dialog* die Aktion **Umbenennen**.
- **Erzeugen (Bei Bedarf ersetzen)**: Ist dieser Typ zugewiesen, wird das abhängige Objekt immer in die Zieldatenbank kopiert. Im *Kopieren-Dialog* erscheint standardmäßig die Aktion **Erzeugen**, wenn das Objekt noch nicht in der Zieldatenbank existiert..
Bei einem Namenskonflikt mit einem Objekt in der Zieldatenbank ersetzt das kopierte Objekt das vorhandene Objekt. In diesem Fall erscheint im *Kopieren-Dialog* standardmäßig die Aktion **Umbenennen**.
- **Anderes Objekt verwenden**: Ist diese Option markiert, erscheint der *Kopieren-Dialog* systematisch, und auch, wenn die Option "Nur bei Namenskonflikt anzeigen" gewählt ist. Beim Bewegen müssen Sie ein Objekt in der Zieldatenbank bestimmen, das anstelle des zu kopierenden abhängigen Objekts benutzt werden soll.

Hinweis: Diese Optionen gelten nur für abhängige Objekte. Für die anderen Objekte gilt beim Bewegen die Standardaktion **Erzeugen (Bei Bedarf umbenennen)**.

- **Kopieren-Dialog anzeigen**: Mit dieser Option legen Sie fest, ob der *Kopieren-Dialog* angezeigt wird. Ist die Option **Immer** gewählt, erscheint das Fenster immer, wenn Objekte bewegt werden und ermöglicht so eine genauere Steuerung der Operation. Ist die Option **Nur bei Namenskonflikten** gewählt, erscheint das Dialogfenster nur, wenn beim Bewegen von Objekten Namenskonflikte mit Objekten in der Zieldatenbank auftreten.

Die Optionen dieser Registerkarte ermöglichen, automatische Backups der Datenbank nach Zeitplan einzurichten. Sie können rasch eine Standardeinstellung wählen oder einen eigenen Zeitplan aufstellen.

Im DropDown-Menü **Automatisches Backup** gibt es folgende Optionen:

- **Nie:** Es gibt keinen Zeitplan für automatisches Backup.
- **Stündlich:** Definiert ein automatisches Backup jede Stunde, beginnend mit der nächsten Stunde.
- **Täglich:** Definiert ein automatisches Backup jeden Tag. Sie können dann die Zeit angeben, wann das Backup starten soll.
- **Wöchentlich:** Definiert ein automatisches Backup jede Woche. Es gibt zwei weitere Bereiche, wo Sie Tag und Uhrzeit zum Starten des Backup eintragen können.
- **Monatlich:** Definiert ein automatisches Backup jeden Monat. Es gibt zwei weitere Bereiche, wo Sie Tag des Monats und Uhrzeit zum Starten des Backup eintragen können.
- **Angepaßt:** Hier können Sie individuell auf Ihre Bedürfnisse ausgerichtete automatische Backups einrichten. Wählen Sie diese Option, erscheinen mehrere zusätzliche Eingabebereiche:
 - **Alle X Stunden:** Damit programmieren Sie Backups nach Stunden. Sie können einen Wert zwischen 1 und 24 eingeben.
 - **Alle X Tage um xx:** Damit programmieren Sie Backups nach Tagen. Geben Sie 1 ein, wenn das Backup jeden Tag durchgeführt werden soll. Ist diese Option markiert, müssen Sie auch die Uhrzeit eintragen, zu der das Backup starten soll.
 - **Alle X Wochen um xx:** Damit programmieren Sie Backups nach Wochen. Geben Sie 1 ein, wenn das Backup jede Woche durchgeführt werden soll. Ist diese Option markiert, müssen Sie auch den Wochentag und die Uhrzeit eintragen, zu der das Backup starten soll. Sie können bei Bedarf auch mehrere Wochentage ankreuzen. Sie können z.B. festlegen, dass das Backup zweimal pro Woche läuft, nämlich am Mittwoch und am Freitag.
 - **Alle X Monate, X.Tag um xx:** Damit programmieren Sie Backups nach Monaten. Geben Sie 1 ein, wenn das Backup jeden Monat durchgeführt werden soll. Ist diese Option markiert, müssen Sie auch den Wochentag und die Uhrzeit eintragen, zu der das Backup starten soll.

Hinweise zu Backups in 4D

Eine ausführliche Beschreibung der 4D Backup Mechanismen finden Sie im Kapitel **Backup und Anwendung wiederherstellen**.

Im Client-/Server-Betrieb lassen sich Backup Parameter nur auf dem 4D Server Rechner ausführen.

Einstellungen für Backup.XML setzen

Alle Einstellungen für Backup und Wiederherstellen sind als eigenständige XML Datei verfügbar. 4D verwendet die Daten in dieser Datei zum Anzeigen der Optionen im Dialogfenster Datenbank-Eigenschaften und immer, wenn ein Backup startet.

Sie können die Backup-Einstellungen über die XML Befehle von 4D oder einen beliebigen XML Editor aufrufen und verändern. Auf diese Weise können Entwickler die Backup-Einstellungen per Programmierung steuern, insbesondere in Anwendungen mit einkompilierter 4D Runtime Volume Licence.

Diese Datei hat den Namen **Backup.XML**. Sie wird automatisch an folgende Stelle gesetzt:

DatenbankOrdner/Preferences/Backup/Backup.XML

Dabei ist DatenbankOrdner der Ordner mit der Strukturdatei der Datenbank. Die Unterordner *Preferences/Backup/* werden bei Bedarf automatisch angelegt. Die XML Keys der Backup-Voreinstellungen werden im Handbuch *XML Keys-Backup* beschrieben. Sie können es auf der [Home Page](#) von 4D Doc Center laden.

Über die Optionen dieser Registerkarte bestimmen Sie die Backup-Dateien, das Logbuch und ihre jeweilige Position.

Inhalt

Hier definieren Sie, welche Dateien bzw. Ordner beim nächsten Backup kopiert werden.

- **Datendatei:** Datendatei der Anwendung. Ist diese Option markiert, wird das aktuelle Logbuch der Anwendung – sofern vorhanden – zeitgleich mit den Daten gesichert.
- **Strukturdatei:** Strukturdatei der Anwendung. Diese Option ermöglicht bei kompilierten Datenbanken ein Backup der Datei .4dc.
- **Anwender-Strukturdatei** (optional): Anwender-Strukturdatei der Anwendung, die angepasste Benutzerformulare enthält – sofern vorhanden. Weitere Informationen dazu finden Sie im Kapitel **Benutzerformulare**.
- **Anhänge:** Hier können Sie Dateien bzw. Ordner wählen, die zusammen mit der Datenbank als Backup gesichert werden sollen. Das können Dokumente, Vorlagen von Plug-Ins, Etiketten, Berichte, Bilder, o.ä. sein. Sie können eigene Dateien oder Ordner bestimmen, deren Inhalt vollkommen als Backup gesichert wird. Jedes angefügte Element wird mit dem kompletten Zugriffspfad im Bereich „Anhänge“ aufgeführt.
 - **Löschen:** Entfernt die gewählte Datei aus der Liste der Anhänge.
 - **Neuer Ordner:** Öffnet ein Dialogfenster, um Ordner für das Backup auszuwählen. Beim Wiederherstellen wird dieser Ordner mit seiner internen Struktur wiederhergestellt. Sie können jeden Ordner bzw. jedes Verzeichnis wählen, das an den Rechner angeschlossen ist, mit Ausnahme des Ordners mit den Dateien der Anwendung.
 - **Neues Dokument:** Öffnet ein Dialogfenster, um Dateien dem Backup hinzuzufügen.

Weitere Informationen zu Dateien der 4D Datenbank finden Sie im Abschnitt **Beschreibung der 4D Dateien**.

Backup Zielordner

Hier definieren Sie, wo die Backup-Dateien und Logbücher (sofern vorhanden) gespeichert werden sollen.

Um den Speicherort dieser Dateien anzusehen, klicken Sie in das DropDown-Menü, um den Pfadnamen anzuzeigen.

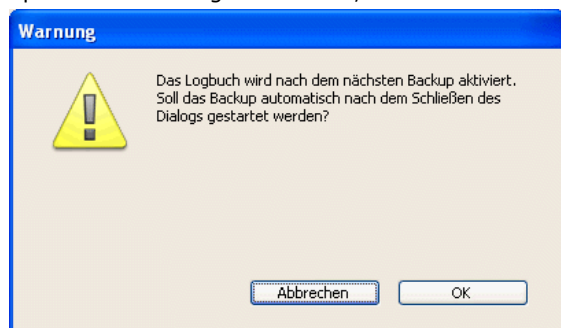
Um den Speicherort dieser Dateien zu ändern, klicken Sie auf die Schaltfläche [...]. Auf dem Bildschirm erscheint ein Auswahldialog, in dem Sie einen Ordner bzw. eine Festplatte zum Ablegen der Backups wählen können. Die Bereiche "Verwendeter Platz" und "Freier Platz" werden automatisch aktualisiert. Sie geben an, wieviel Speicherplatz auf der Festplatte mit dem gewählten Ordner belegt und wieviel frei ist.

Logbuch-Verwaltung

Ist die Option **Benutze Logbuch** aktiv, verwendet die Datenbank ein Logbuch. Der Pfad wird im Bereich darunter angegeben. Die Datenbank lässt sich nicht ohne Logbuch öffnen, wenn diese Option markiert ist.

Es wird automatisch mit der Option **Automatisches Backup** angelegt, die beim Erstellen der Datenbank standardmäßig aktiv ist. Weitere Informationen dazu finden Sie im Abschnitt **OBJECT SET INDICATOR TYPE**. Das Logbuch hat den Namen *DatendateiName.journal* und liegt im gleichen Ordner wie die Strukturdatei.

Aktivieren Sie ein neues Logbuch, muss zuvor ein Backup der Daten in der Datenbank erstellt werden. Markieren Sie diese Option und bestätigen die Seite, erscheint auf dem Bildschirm die Meldung, dass ein Backup erforderlich ist:



Klicken Sie auf die Schaltfläche **OK**, startet sofort das Backup, dann wird das Logbuch aktiviert. Klicken Sie auf **Abbrechen**, wird die Anfrage gesichert und das Erstellen des **Logbuchs** verschoben. Es wird dann erst nach dem nächsten Backup der Datenbank erstellt.

Hinweise zu Backups in 4D

Eine ausführliche Beschreibung der 4D Backup Mechanismen finden Sie im Kapitel **Backup und Anwendung wiederherstellen**.

Im Client-/Server-Betrieb lassen sich Backup Parameter nur auf dem 4D Server Rechner ausführen.

Einstellungen für Backup.XML setzen

Alle Einstellungen für Backup und Wiederherstellen sind als eigenständige XML Datei verfügbar. 4D verwendet die Daten in dieser Datei zum Anzeigen der Optionen im Dialogfenster Datenbank-Eigenschaften und immer, wenn ein Backup startet. Sie können die Backup-Einstellungen über die XML Befehle von 4D oder einen beliebigen XML Editor aufrufen und verändern. Auf diese Weise können Entwickler die Backup-Einstellungen per Programmierung steuern, insbesondere in Anwendungen mit einkompilierter 4D Runtime Volume Licence.

Diese Datei hat den Namen **Backup.XML**. Sie wird automatisch an folgende Stelle gesetzt:

DatenbankOrdner/Preferences/Backup/Backup.XML

Dabei ist DatenbankOrdner der Ordner mit der Strukturdatei der Datenbank. Die Unterordner *Preferences/Backup/* werden bei Bedarf automatisch angelegt. Die XML Keys der Backup-Voreinstellungen werden im Handbuch *XML Keys-Backup* beschrieben. Sie können es auf der [Home Page](#) von 4D Doc Center laden.

Die Optionen dieser Registerkarte bestimmen allgemeinen Einstellungen für Backup und automatische Wiederherstellung der Datenbank.

Allgemeine Einstellungen

Hier bestimmen Sie spezifische Arbeitsweisen beim Ausführen von Sicherungen.

- **Aufheben der letzten X Backup-Dateien:** Damit legen Sie fest, dass die älteren Backup-Dateien gelöscht werden und vermeiden so eine Überbelegung der Festplatte.
Das funktioniert folgendermaßen: Ist das aktuelle Backup komplett, löscht 4D das älteste Archiv, wenn es an derselben Stelle liegt und denselben Namen hat. Sie können auch bestimmen, dass das älteste Archiv vor Ausführen des Backup gelöscht wird, um Platz auf der Festplatte zu sparen.
Ist z.B. als Anzahl 3 definiert, erstellen die ersten drei Backups jeweils die Archive MeineDB-0001, MeineDB-0002 und MeineDB-0003. Beim 4. Backup wird das Archiv MeineDB-0004 erstellt und MeineDB-0001 gelöscht. Standardmäßig ist der Löschmechanismus aktiviert und 4D behält 3 Backup-Sätze.
Wollen Sie diesen Mechanismus deaktivieren, deaktivieren Sie einfach diese Option.
Hinweis: Dieser Parameter betrifft sowohl die Backup-Sätze der Datenbank als auch das Logbuch der Backup-Sätze.
- **Backup nur wenn Datendatei geändert:** Ist diese Option markiert, startet 4D geplante Backups nur, wenn seit dem letzten Backup Daten in der Datenbank hinzugefügt, verändert oder gelöscht wurden. Sonst wird das geplante Backup annulliert und auf das nächste geplante Backup gelegt. Es erscheint keine Fehlermeldung; das Logbuch notiert jedoch, dass das Backup verschoben wurde. Damit sparen Sie Rechnerzeit für Backups von Datenbanken, die überwiegend zum Lesen genutzt werden. Beachten Sie jedoch, dass mit dieser Option keine Änderungen an der Struktur oder angefügten Dateien berücksichtigt werden.
Hinweis: Dieser Parameter betrifft sowohl Backups der Datenbank als auch des Logbuchs.
- **Lösche ältesten Satz nach/vor Backup:** Diese Option wird nur verwendet, wenn die Option "Aufheben der letzten X Backup Dateien" markiert ist. Damit geben Sie an, ob 4D das älteste Archiv vor bzw. nach Ausführen des Backup löscht. Das funktioniert nur, wenn das älteste Archiv weder umbenannt noch verschoben wurde.
- **Wenn Backup fehlschlägt:** Damit legen Sie fest, wie fehlgeschlagene Backups verwaltet werden (Backup nicht möglich). Lässt sich ein Backup nicht ausführen, ermöglicht 4D einen zweiten Versuch.
Hinweis: 4D wertet ein Backup als fehlgeschlagen, wenn die Datenbank zum Zeitpunkt des automatisch geplanten Backups nicht gestartet ist.
 - **Erneut nach planmäßigem Datum und Zeit:** Diese Option ist nur sinnvoll für geplante automatische Backups. Das fehlgeschlagene Backup wird abgebrochen und ein Fehler erscheint.
 - **Erneut nach X Sekunden, Minuten oder Stunden:** Mit dieser Option wird nach der Wartezeit ein neues Backup versucht. So lassen sich bestimmte Umstände umgehen, die ein Backup blockieren können. Sie können die Wartezeit in Sekunden, Minuten oder Stunden angeben. Scheitert auch der 2. Versuch, erscheint ein Fehler. Das wird auch im Statusbereich des letzten Backup und im Logbuch des Backup vermerkt.
 - **Abbruch nach X Versuchen:** Damit legen Sie die maximale Anzahl für gescheiterte Backup-Versuche fest. Ist die max. Anzahl der Versuche erreicht, wird das Backup annulliert und der Fehler 1401 generiert ("Die max. Anzahl der Backup-Versuche ist erreicht; automatisches Backup ist vorübergehend deaktiviert"). In diesem Fall wird ein neues automatisches Backup erst ausgeführt, wenn die Anwendung neu gestartet bzw. ein manuelles Backup erfolgreich ausgeführt wurde.
Dieser Parameter ist hilfreich, um bei einem größeren Problem (das den Eingriff des Administrators erfordert) zu verhindern, dass immer wieder ein Backup-Versuch läuft, der letztendlich die Performance der Datenbank beeinträchtigt. Dieser Parameter ist standardmäßig nicht markiert.

Archive

Hier legen Sie die Einstellungen zum Anlegen des Archivs fest. Sie gelten für die Dateien von Haupt-Backups sowie für Logbücher der Backups.

- **Segmentgröße (Mb)**
Mit 4D können Sie Archive segmentieren, d.h. ein Archiv in mehrere kleinere aufteilen. So können Sie z.B. ein Backup über verschiedene Medien, wie CD, ZIP o.ä. speichern. Beim Wiederherstellen fügt 4D die Segmente automatisch wieder zusammen. Jedes Segment hat den Namen MeineDBxxxx-yyyy].4BK, wobei xxxx die Nummer des Backup ist, yyyy die Nummer des Segments.
Segment Größe ist eine Combobox, in der Sie für jedes Segment des Backup die Größe in Mb festlegen. Sie können eine vordefinierte Größe wählen oder eine eigene Größe zwischen 0 und 2048 angeben. Bei 0 wird nicht segmentiert, das entspricht dem Eintrag **Keine**.
- **Komprimierung**
4D komprimiert die Backups, um Platz zu sparen. Das kann jedoch bei großen Datenmengen den Backup-Prozess beträchtlich verlangsamen.
Es gibt folgende Möglichkeiten:
 - **Keine:** Die Dateien werden nicht komprimiert. Das Backup läuft schneller ab, die Archiv-Dateien sind jedoch um einiges größer.
 - **Schnell** (Standard): Das ist ein Mittelmaß zwischen Geschwindigkeit des Backup und Größe des Archivs.
 - **Kompakt:** Hier wird das Archiv maximal komprimiert. Die Dateien beanspruchen den kleinstmöglichen Platz auf der Festplatte, das Backup läuft jedoch um einiges langsamer ab.

- **Interlacing und Redundanz**

4D erstellt Archive über spezifische Algorithmen, die auf Operationen zur Optimierung (Interlacing) und Sicherheit (Redundanz) beruhen. Dies können Sie gemäß Ihren Anforderungen einstellen. 4D enthält standardmäßig den Wert **Keine**. Weitere Möglichkeiten sind **Niedrig, Mittel** und **Hoch**.

- **Interlacing:** Damit werden Daten in nicht aufeinander folgenden Sektoren gespeichert, um das Risiko bei Beschädigung eines Sektors zu begrenzen. Je höher die Rate, desto besser ist die Sicherheit; Die Datenbearbeitung benötigt jedoch mehr Speicher.
- **Redundanz:** Damit werden die Daten in einer Datei abgesichert, indem dieselbe Information mehrere Male wiederholt wird. Je höher die Redundanzrate, desto besser ist die Sicherheit der Datei. Nachteile sind langsame Speicherung und entsprechend große Dateien.

Automatische Wiederherstellung

Diese Optionen bestimmen die automatischen Mechanismen beim Öffnen einer beschädigten Datenbank.

- **Letztes Backup wiederherstellen, wenn Datenbank beschädigt:** Mit dieser Option startet das Programm automatisch die Wiederherstellung der Daten des letzten gültigen Backup der Datenbank, wenn beim Hochfahren der Datenbank eine Unregelmäßigkeit festgestellt wird (z.B. beschädigte Datei). Der Benutzer selbst muss nichts ausführen; die Operation wird jedoch im Logbuch des Backup vermerkt.
Hinweis: Beim automatischen Wiederherstellen wird nur die Datendatei wiederhergestellt. Soll das auch für die angefügten Dateien oder die Strukturdatei gelten, müssen Sie eine manuelle Wiederherstellung ausführen.
- **Letztes Logbuch integrieren, wenn Datenbank unvollständig:** Ist diese Option markiert, integriert das Programm automatisch das Logbuch, wenn die Datenbank wiederhergestellt oder geöffnet wird.
 - Beim Öffnen einer Datenbank wird das aktuelle Logbuch automatisch integriert, wenn 4D feststellt, dass die im Logbuch gespeicherten Operationen in den Daten nicht vorhanden sind. Das passiert z.B. bei Stromausfall, wenn es Operationen im Daten-Cache gibt, die noch nicht in den Hauptspeicher übertragen wurden.
 - Hat beim Wiederherstellen einer Datenbank das aktuelle Logbuch bzw. Logbuch des Backup dieselbe Nummer wie das Backup, das im gleichen Ordner gespeichert ist, prüft 4D seinen Inhalt. Enthält es Operationen, die nicht in der Datendatei enthalten sind, wird es automatisch integriert.

Auf Benutzerseite erscheint kein Dialogfenster, die Operation läuft vollkommen automatisch ab. Ziel dabei ist, die Verwendung so einfach wie möglich zu machen. Die Operation wird im Backup Journal vermerkt.

Hinweise zu Backups in 4D

Eine ausführliche Beschreibung der 4D Backup Mechanismen finden Sie im Kapitel **Backup und Anwendung wiederherstellen**.

Im Client-/Server-Betrieb lassen sich Backup Parameter nur auf dem 4D Server Rechner ausführen.

Einstellungen für Backup.XML setzen

Alle Einstellungen für Backup und Wiederherstellen sind als eigenständige XML Datei verfügbar. 4D verwendet die Daten in dieser Datei zum Anzeigen der Optionen im Dialogfenster Datenbank-Eigenschaften und immer, wenn ein Backup startet.

Sie können die Backup-Einstellungen über die XML Befehle von 4D oder einen beliebigen XML Editor aufrufen und verändern. Auf diese Weise können Entwickler die Backup-Einstellungen per Programmierung steuern, insbesondere in Anwendungen mit einkompilierter 4D Runtime Volume Licence.

Diese Datei hat den Namen **Backup.XML**. Sie wird automatisch an folgende Stelle gesetzt:

DatenbankOrdner/Preferences/Backup/Backup.XML

Dabei ist DatenbankOrdner der Ordner mit der Strukturdatei der Datenbank. Die Unterordner *Preferences/Backup/* werden bei Bedarf automatisch angelegt. Die XML Keys der Backup-Voreinstellungen werden im Handbuch *XML Keys-Backup* beschrieben. Sie können es auf der [Home Page](#) von 4D Doc Center laden.

Die Registerkarten der Seite **Client-Server** enthalten Parameter zum Einsatz der Anwendung im Client/Server Betrieb. Diese Einstellungen werden natürlich nur berücksichtigt, wenn die Datenbank im remote Modus läuft.

Netzwerk

Beim Starten veröffentlichen

Mit dieser Option geben Sie an, ob die 4D Server Datenbank im Verbindungsdialog in der Liste der veröffentlichten Datenbanken erscheint.

- Ist diese Option markiert (Standard), wird die Datenbank veröffentlicht und erscheint in der Liste der veröffentlichten Datenbanken (Seite **Verfügbar**).
- Ist diese Option nicht markiert, wird die Datenbank nicht veröffentlicht und erscheint nicht in der Liste. Benutzer müssen, um sich anzumelden, die Adresse der Datenbank manuell auf der Seite **Angepasst** des Verbindungsdialogs eingeben.

Hinweis: Verändern Sie diesen Parameter, müssen Sie die Server Datenbank neu starten, damit die Änderung berücksichtigt wird.

Anzeigename

Der Name zum Veröffentlichen einer 4D Server Datenbank lässt sich ändern. Das kann z.B. der Name auf der dynamischen Veröffentlichungsseite **Verfügbar** im Verbindungsdialog sein (siehe Abschnitt **Anmeldung an eine 4D Server Datenbank**). 4D Server verwendet standardmäßig den Namen der Strukturdatei der Datenbank. Sie können einen beliebigen Namen eingeben.

Hinweis: Dieser Parameter wird bei eigenen Client-/Server-Anwendungen nicht berücksichtigt. Im Prinzip meldet sich die Client-Anwendung direkt an die Server Anwendung an, ohne über den Anmeldedialog zu gehen. Der Dialog erscheint jedoch, wenn ein Fehler auftritt; in diesem Fall ist der Publikationsname der Server-Anwendung der Name der kompilierten Datenbank.

Port Nummer

Mit dieser Option können Sie die TCP Port Nummer ändern, unter der 4D Server die Datenbank veröffentlicht. Diese Information wird in der Strukturdatei der Datenbank und auf jedem Client-Rechner gespeichert. 4D Server und 4D im remote Modus verwenden standardmäßig die Port Nummer 19813.

Hinweis: 4D Server verwendet stets zwei Ports zur Kommunikation mit Clients: den o.a. und den darauf folgenden, also 19814. Beide Ports müssen in der Firewall geöffnet werden.

Sie müssen diesen Wert anpassen, wenn Sie mehrere 4D Anwendungen auf demselben Rechner mit dem TCP Protokoll verwenden wollen; in diesem Fall müssen Sie für jede Anwendung eine andere Port Nummer vergeben.

Ändern Sie diesen Wert auf 4D Server bzw. 4D, wird er automatisch auf alle 4D Rechner übertragen, die an die Datenbank angemeldet sind. Um andere nicht angeschlossene Client-Rechner zu aktualisieren, geben Sie bei der nächsten Anmeldung im Verbindungsdialog auf der Seite **Angepasst** die neue Port Nummer ein, getrennt durch Doppelpunkt. Sie wird an die IP-Adresse des Server-Rechners angehängt, getrennt durch Doppelpunkt. Ist die Nummer z.B. 19888, sieht der Eintrag folgendermaßen aus:



Beispiel: Die Nummer könnte lauten 192.121.122.123:19888 (achten Sie auf den Doppelpunkt!)

Hinweis: Auf der Seite **Verfügbar** zur dynamischen Veröffentlichung erscheinen nur die Datenbanken, die auf demselben Port, wie in remote 4D definiert, veröffentlicht werden.

4D Server und Port Nummern

4D Server verwendet drei TCP Port zur Kommunikation zwischen internen Servern und Clients:

- **SQL Server:** standardmäßig 19812 (lässt sich über die Seite "SQL" der Datenbank-Eigenschaften ändern).
- **Application Server:** standardmäßig 19813 (lässt sich in den Einstellungen auf der Seite "Client-Server/Konfiguration" verändern, siehe unten).
- **DB4D Server** (database server): standardmäßig 19814. Diese Port Nummer lässt sich nicht direkt ändern, sie besteht aber immer aus der Port Nummer des Application Server + 1.
Meldet sich ein remote 4D an 4D Server an, verwendet es den TCP Port des Application Server (19813 oder den Port, der im Verbindungsdialog in der IP Adresse nach dem Doppelpunkt ':' angezeigt wird.). Die Anmeldung an andere Server über ihre entsprechenden Ports ist dann automatisch; sie müssen nicht mehr eigens spezifiziert werden.
Beachten Sie, dass die drei TCP Port bei Zugriff über einen Router oder eine Firewall explizit geöffnet werden müssen.

Benutzer-Authentifizierung mit Domain Server

Mit dieser Option können Sie Ihre Client-Server Lösungen unter Windows mit SSO (*Single Sign On* - Einmalanmeldung) ausrüsten. Markieren Sie diese Option, verbindet sich 4D direkt mit der Active Directory Zone des Windows Domain Server und erhält die verfügbaren Tokens der Authentifizierung.

Weitere Informationen dazu finden Sie im Abschnitt **Single Sign On (SSO) unter Windows**.

Dienstprinzipalname

Ist Single Sign On (SSO) aktiviert (siehe oben), müssen Sie dieses Feld ausfüllen, wenn Sie Kerberos als Authentifizierungsprotokoll nutzen wollen.

Weitere Informationen dazu finden Sie im Abschnitt **Single Sign On (SSO) unter Windows**.

Client-Server Verbindungs-Timeout

Im Ablaufbalken legen Sie das Timeout fest, d.h. die Dauer an Inaktivität, nach der die Verbindung zwischen 4D Server und den angeschlossenen Client-Rechnern beendet wird.

Bei **Ohne** gibt es kein Timeout. In diesem Fall erfolgt keine Kontrolle der Client-Aktivität, d.h. ein inaktiver Client bleibt auf unbestimmte Zeit an den Server angeschlossen.

Ist ein Timeout eingestellt, beendet der Server die Verbindung zu einem Client, wenn er innerhalb der vorgegebenen Zeitspanne keine Anfragen von diesem Client erhält.

Client-Server Kommunikation

Client beim Starten anmelden

Ist diese Option markiert, können alle Arbeitsstationen, die an die 4D Server Anwendung angemeldet sind, Methoden im remote Modus ausführen. Weitere Informationen dazu finden Sie im Abschnitt **Serverprozeduren auf Client-Rechnern**.

Verschlüsselte Client-Server Verbindungen

Damit können Sie den gesicherten Modus zur Kommunikation zwischen Server Rechner und Rechnern mit remote 4D aktivieren. Weitere Informationen dazu finden Sie im Abschnitt **Client/Server Verbindungen verschlüsseln**.

'Resources'-Ordner während einer Session aktualisieren

Mit dieser Einstellung können Sie den Aktualisierungsmodus für den Ordner **Resources** auf den angeschlossenen Rechnern global setzen, wenn der Ordner **Resources** während einer Arbeitssitzung verändert wird. (Beim Öffnen einer neuen Arbeitssitzung wird er immer automatisch auf dem remote Rechner synchronisiert). Es gibt drei Möglichkeiten:

- **Nie:** Der lokale Ordner **Resources** wird während der Arbeitssitzung nicht aktualisiert. Die vom Server gesendete Meldung wird ignoriert. Dieser Ordner lässt sich bei Bedarf manuell über den Menüeintrag **Lokale Ressourcen aktualisieren** des Menüs "Aktionen" aktualisieren (siehe unter **Ressourcen Explorer verwenden**).
- **Immer:** Der lokale Ordner **Resources** wird während der Arbeitssitzung automatisch synchronisiert, wenn der Server eine Meldung sendet.
- **Fragen:** Sendet der Server eine Meldung, erscheint auf den Client-Rechnern ein Dialogfenster, das die Änderung meldet. Der Benutzer kann dann die Synchronisation des lokalen Ordners **Resources** bestätigen oder abweisen.

Der Ordner **Resources** bündelt die eigenen Dateien für die Oberfläche der Anwendungen, z.B. Bilder, Übersetzungsdateien. Die automatische oder manuelle Operation ermöglicht, jeden Client zu informieren, wenn sich der Inhalt dieses Ordners geändert hat. Weitere Informationen dazu finden Sie im Abschnitt **Ordner Resources verwalten**.

Hinweis: Diese Einstellmöglichkeit richtet sich an Team-Entwickler, sie ist für Anwender einer Datenbank belanglos.

Öffne Struktur im Modus

Damit definieren Sie den Öffnen-Modus der Datenbankstruktur auf den Client-Rechnern. Standardmäßig ist der Modus **Lesen/Schreiben** eingestellt. Sie können aber auch den Modus **Nur Lesen** wählen, um Änderungen an der Struktur zu verhindern.

Konfiguration Erlauben/Verweigern-Tabelle

Mit dieser Tabelle setzen Sie die Regeln für den Zugriff auf die Datenbank. Das richtet sich nach den IP-Adressen des remote-Rechners. Das sorgt z.B. bei Anwendungen mit sensiblen Daten für mehr Sicherheit.

Hinweis: Diese Tabelle steuert nicht die Web Verbindungen.

Diese Tabelle funktioniert folgendermaßen:

- In der Spalte "Erlauben-Verweigern" wählen Sie im PopUp-Menü die anzuwendende Regel, d.h. **Erlauben** oder **Verweigern**. Um eine Regel hinzuzufügen, klicken Sie auf die Schaltfläche **Hinzufügen**. In der Tabelle erscheint dann eine neue Zeile. Über die Schaltfläche **Löschen** können Sie die aktuelle Zeile entfernen.
- In der Spalte "IP Adresse" setzen Sie die IP-Adresse(n), für welche die Regel gelten soll. Dazu klicken Sie in die Spalte und geben die Adresse in folgender Form ein: 123.45.67.89 (IPv4 Format) oder 2001:0DB8:0000:85A3:0000:0000:AC1F:8001 (IPv6 Format).
Sie können auch nur den Anfang der Adresse eingeben, gefolgt von einem * (Stern). Beispiel: 192.168.* gibt alle Adressen an, die mit 192.168 beginnen.
- Die Anwendung der Regeln richtet sich nach der Reihenfolge der Anzeige in der Tabelle. Bei zwei widersprüchlichen Regeln hat die zuerst angelegte Regel Priorität.
Durch Ändern der aktuellen Sortierung können Sie die Zeilen neu ordnen (Klicken Sie auf den Kopfteil der Spalte, um die Sortierrichtung zu wechseln). Sie können die Zeilen auch per Drag&Drop bewegen.
- Aus Sicherheitsgründen ist die Anmeldung nur an Adressen möglich, die zu einer Regel passen, d.h. enthält die Tabelle eine bzw. mehrere Regeln Verweigern, werden alle Adressen abgewiesen, da keine zu mindestens einer Regel passt. Wollen Sie nur einige Adressen abweisen, andere jedoch zulassen, fügen Sie am Ende der Tabelle die Regel **Erlauben *** hinzu.

Beispiel:

Verweigere 192.168.* (weist alle Adressen ab, die mit 192.168 beginnen)

Erlaube * (lässt jedoch alle anderen Adressen zu)

4D Server verwendet standardmäßig keine Verbindungseinschränkungen, d.h. die erste Zeile der Tabelle enthält die Bezeichnung **Erlaube** und das Zeichen * (alle Adressen sind zugelassen).

Über die Registerkarten der Seite **Web** können Sie verschiedene Aspekte des in 4D integrierten Web Server definieren (Sicherheit, Startup, Verbindungen, Web Services, etc.). Weitere Informationen über die Arbeitsweise des 4D Web Server finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Web Server**, weitere Informationen über 4D Web Services im *Designmodus* im Kapitel **Web Services publizieren und einsetzen**.

Information zum Veröffentlichen

Web Server bei Startup starten

Gibt an, ob der Web Server beim Starten der 4D Anwendung ebenfalls gestartet wird. Weitere Informationen dazu finden Sie im Abschnitt **Web Server konfigurieren und Verbindung verwalten**.

HTTP Port

4D veröffentlicht eine Web Datenbank standardmäßig auf dem regulären Web TCP/IP (HTTP) Port, also Port 80. Wird dieser Port bereits von einem anderen Web Dienst benutzt, müssen Sie den von 4D verwendeten Port für diese Datenbank ändern. Durch Ändern des HTTP Ports können Sie den 4D Web Server unter Mac OS X starten, ohne der Root Benutzer des Rechners zu sein. Weitere Informationen dazu finden Sie im Abschnitt **Web Server konfigurieren und Verbindung verwalten**. Gehen Sie zum Eingabebereich für den **HTTP Port** und geben Sie einen geeigneten Wert an, d.h. eine Nummer, die noch nicht von einem anderen TCP/IP Dienst auf demselben Rechner benutzt wird.

Hinweis: Tragen Sie Null (0) ein, verwendet 4D die standardmäßige HTTP Portnummer 80.

Von einem Web Browser aus müssen Sie diese spezifische HTTP Port Nummer in die Adresse eingeben, mit der Sie die Verbindung zur Web Datenbank aufbauen. Das Ende der Adresse nach dem Doppelpunkt ist die Port-Nummer. Beispiel: Ihre HTTP Port Nummer lautet 8080, Ihre Adresse ist demzufolge "123.4.567.89:8080".

Warnung: Wenn Sie nicht die Standardnummern verwenden, also 80 für HTTP und 443 für HTTPS, achten Sie darauf, dass Sie nicht die Standard Ports anderer Dienste verwenden, die Sie zur gleichen Zeit einsetzen wollen. Wollen Sie z.B. auf Ihrem Web Server Rechner auch das FTP Protokoll verwenden, benützen Sie nicht die Nummern 20 und 21, denn das sind die Standardnummern für dieses Protokoll. Weitere Informationen zu den Standardnummern für TCP/IP Ports finden Sie im Handbuch *4D Internet Commands* unter **Appendix B, TCP Port Numbers**. Port Nummern unter 256 sind für allgemein bekannte Dienste reserviert, Port Nummern zwischen 256 und 1024 für spezifische Dienste auf UNIX Plattformen. Für maximale Sicherheit benutzen Sie als Port Nummer am besten höhere Zahlen, z.B. im zweitausender oder dreitausender Bereich.

IP Adresse für HTTP Anfragen definieren

Sie können die IP Adresse festlegen, über die der Web Server HTTP Anfragen empfangen muss.

Der Server antwortet standardmäßig allen IP Adressen (Option **Alle**), d.h. IPv4 und IPv6 Adressen.

Die DropDown-Liste zeigt automatisch alle auf dem Rechner verfügbaren IPv4 Adressen an. Wählen Sie eine spezifische Adresse, beantwortet der Server nur Anfragen an diese Adresse.

Diese Funktionalität dient für 4D Web Server auf Rechnern mit mehreren TCP/IP Adressen. Das ist für die meisten Internet Host Provider der Fall.

Hinweis: Um Klarheit zu bewahren, erscheinen IPv6 Adressen nicht in der DropDown-Liste "IP Adresse". Zum Filtern von IPv6 Adressen empfehlen wir, die Firewall Einstellungen des Systems entsprechend zu konfigurieren.

Zum Einrichten eines MultiHoming Systems sind auf dem Rechner mit dem Web Server spezifische Konfigurationen erforderlich:

- **Sekundäre IP Adressen auf Mac OS installieren**

Um ein MultiHoming System auf Mac OS einzurichten:

1. Öffnen Sie das Steuerfenster **TCP/IP**.
2. Wählen Sie im PopUp-Menü **Konfiguration** die Option **Manuell**.
3. Legen Sie ein Textdokument mit Namen "Sekundäre IP Adressen" an und sichern es in Ihrem Systemordner im Unterordner **Preferences**.

Jede Zeile von "Sekundäre IP Adressen" besteht aus einer sekundären IP Adresse, einer optionalen Subnetz-Maske und einer dazugehörigen Router-Adresse.

Weitere Informationen dazu finden Sie in der Dokumentation zu Apple.

- **Sekundäre IP Adressen unter Windows installieren**

Um ein MultiHoming System unter Windows einzurichten

1. Wählen Sie folgende Befehlsfolge (oder die Entsprechungen je nach ihrer Windows-Version): Für Windows XP gilt: Menü **Start > Steuerung > Netzwerk und Internet-Verbindungen > Netzwerkverbindung > Local Area Connection** (Eigenschaften) > **Internet Protokoll (TCP/IP)** > Schaltfläche **Eigenschaften** > Schaltfläche **Erweitert**. Auf dem Bildschirm erscheint das Dialogfenster "Erweiterte TCP/IP Einstellungen".
2. Klicken Sie im Bereich "IP Adressen" auf die Schaltfläche **Hinzufügen** und fügen Sie zusätzliche IP Adressen ein. Sie können bis zu 5 verschiedene IP Adressen definieren.

Sie können bis zu 5 verschiedene IP Adressen definieren. Evtl. müssen Sie sich dafür an Ihren Systemadministrator wenden.

HTTPS aktivieren

Zeigt an, ob der Web Server sichere Verbindungen akzeptiert. Weitere Informationen dazu finden Sie im Abschnitt **TLS Protokoll verwenden (HTTPS)**.

HTTPS Port

Damit können Sie die Nummer des TCP/IP Port verändern, welche der Web Server für gesicherte HTTP Verbindungen über TLS (HTTPS Protokoll) verwendet. Standardmäßig ist die Nummer 443 eingetragen.

Es gibt zwei Hauptgründe, diese Port-Nummer zu ändern:

- Aus Sicherheitsgründen: Hacker-Attacken auf Web Server konzentrieren sich in der Regel auf standardmäßige HTTP Ports (80 und 443).
- Damit Standardbenutzer auf Mac OS X den Web Server im gesicherten Modus starten können. Auf Mac OS X sind für TCP/IP Ports, die für Veröffentlichungen im Web reserviert sind (0 bis 1023), bestimmte Zugriffsrechte erforderlich: Nur der Root Benutzer kann eine Verbindung über diese Ports öffnen. Damit auch Standardbenutzer den Web Server starten können, ist eine Lösung, die TCP/IP Port-Nummer zu ändern. Weitere Informationen dazu finden Sie im Abschnitt **Web Server konfigurieren und Verbindung verwalten**. Sie können jeden gültigen Wert übergeben. Unter Mac OS X sollten Sie zur Vermeidung von Zugriffskonflikten einen Wert größer als 1023 übergeben. Weitere Informationen zu Port Nummern finden Sie im oberen Abschnitt **HTTP Port**.

Erlaube Datenbankzugriff per 4DSYNC URLs

Über diese Option auf der Seite "Web/Konfiguration" der Datenbank-Eigenschaften können Sie die Unterstützung von Anfragen mit /4DSYNC URLs steuern. Diese URLs dienen zum Synchronisieren von Daten über HTTP. Weitere Informationen dazu finden Sie unter **URL 4DSYNC/**.

Damit können Sie die spezifische Bearbeitung von Anfragen mit /4DSYNC aktivieren bzw. deaktivieren:

- Ist die Option nicht markiert, werden /4DSYNC Anfrage als standardmäßige Anfragen gewertet und erlauben keine spezifische Bearbeitung. So wird bei einer Synchronisationsanfrage eine Antwort vom Typ "404 - Ressource nicht verfügbar" gesendet.
- Ist die Option markiert, wird der Synchronisationsmechanismus aktiviert. /4DSYNC Anfragen werden dann als spezielle Anfragen gewertet und vom 4D HTTP Server analysiert.

Standardmäßig gilt:

- Diese Option ist **nicht markiert** in Anwendungen, die mit 4D ab Version 13 erstellt wurden .
- Zur Wahrung der Kompatibilität ist diese Option in Anwendungen **markiert**, die aus einer älteren 4D Version konvertiert wurden. Wir empfehlen, sie zu deaktivieren, falls Ihre Anwendung nicht die Funktion HTTP Replikation verwendet.

Die Reichweite dieser Option gilt lokal für die Anwendung. Der Web Server muss erneut gestartet werden, damit sie berücksichtigt wird.

Pfade

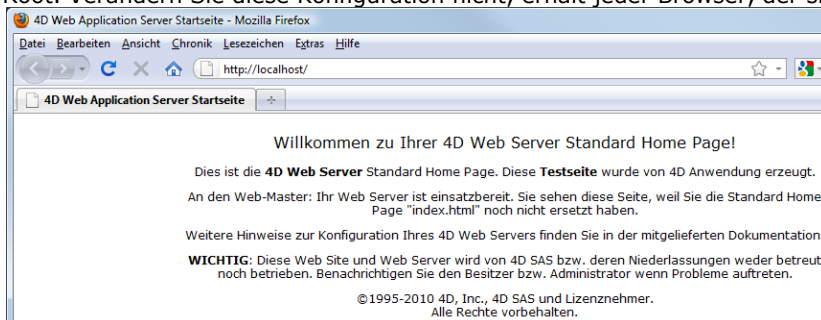
Standard HTML Root

Hier bestimmen Sie einen Standardordner, in dem die Dateien der Web Site abgelegt werden, sowie die hierarchische Ebene auf der Festplatte, ab der die Dateien nicht zugänglich sind. Weitere Informationen dazu finden Sie im Abschnitt **Sicherheit der Verbindung**.

Standard Home Page definieren

Sie können für alle Browser, die sich an die Datenbank anmelden, eine Standard Home Page definieren. Sie kann statisch oder halbdynamisch sein.

Beim ersten Start des Web Servers erstellt 4D eine Home Page mit Namen "index.html" und legt sie in den Ordner HTML Root. Verändern Sie diese Konfiguration nicht, erhält jeder Browser, der sich an den Web Server anmeldet, folgende Seite:



Um die Standard Home Page zu verändern, ersetzen Sie einfach im Root Ordner der Datenbank den Ordner "index.html" durch Ihren eigenen Ordner oder tragen Sie im Eingabebereich den Zugriffspfad in Bezug auf den dafür vorgesehenen Ordner ein.

Der Zugriffspfad muss in Bezug auf den Standard HTML Root Ordner eingerichtet werden.

Um die Plattformunabhängigkeit Ihrer Datenbanken zu gewährleisten, verwendet der 4D Web Server spezielle Schreibregeln für Zugriffspfade:

- Ordner werden durch Schrägstrich ("/") voneinander getrennt
- Der Zugriffspfad darf nicht mit einem Schrägstrich ("/") enden.
- Um in der Ordnerhierarchie eine Ebene höher zu gehen, geben Sie vor dem Ordnernamen zwei Punkte ("..") ein.
- Der Zugriffspfad darf nicht mit einem Schrägstrich ("/") beginnen.

Soll Ihre Standard Homepage zum Beispiel lauten "MyHome.htm" und liegt sie im Ordner "Web", geben Sie "Web/MyHome.htm" ein.

Hinweis: Mit dem 4D Befehl **WEB SET HOME PAGE** können Sie für jeden Web Prozess eine Standard Home Page definieren.

Legen Sie keine Standard Home Page an, wird die **Datenbankmethode On Web Connection** aufgerufen. Sie müssen dann selbst für die Übertragung einer Startseite sorgen.

Cache

Der 4D Web Server hat einen Cache, um angeforderte statische Seiten, GIF Bilder, JPEG Bilder (<128 kb) und Stilvorlagen (.css Dateien) in den Speicher zu laden.

Mit dem Cache steigern Sie die Leistung des Web Servers beim Senden statischer Seiten.

Der Cache wird zwischen allen Web Prozessen aufgeteilt. Seine Größe legen Sie in den Einstellungen der Datenbank fest. Der Cache für statische Seiten ist ab 4D V14 R5 für neue Datenbanken standardmäßig aktiviert. Um ihn zu deaktivieren, deaktivieren Sie die Option **Benutze 4D Web Cache**.

Im Bereich **Seiten Cache Größe** können Sie die Größe des Cache verändern. Sie richtet sich nach Anzahl und Größe der statischen Seiten Ihrer Web Site, sowie nach den jeweiligen Ressourcen der Host Rechner.

Hinweis: Während Sie mit Ihrer Web Datenbank arbeiten, können Sie mit dem 4D Befehl **WEB GET STATISTICS** die Leistung des Cache prüfen. Stellen Sie zum Beispiel fest, dass die Nutzungsrate des Cache fast 100% erreicht, sollten Sie erwägen, die zugewiesene Größe zu erhöhen.

Auch mit den URL /4DSTATS und /4DHTMLSTATS erhalten Sie Information über den Status des Cache. Weitere Informationen dazu finden Sie im Abschnitt **Information über die Web Site**.

Sobald der Cache aktiviert ist, sucht der 4D Web Server nach einer vom Browser angeforderten statischen Seite zuerst im Cache. Findet er die Seite, sendet er sie sofort. Findet er sie nicht, lädt 4D die Seite von der Festplatte und legt sie in den Cache. Ist der Cache voll und wird mehr Platz benötigt, entfernt 4D zuerst die ältesten aufgerufenen Seiten.

Cache leeren

Sie können jederzeit Seiten und Bilder aus dem Cache entfernen, zum Beispiel, wenn Sie eine statische Seite geändert haben und diese wieder in den Cache laden wollen.

Klicken Sie dazu auf die Schaltfläche **Cache löschen**. Der Cache wird dann sofort geleert.

Hinweis: Sie können auch die spezifische URL **/4DCACHECLEAR** verwenden.

Web Prozess

Timeout für interaktive Web-Prozesse

Hier definieren Sie das maximale Timeout zum Schließen inaktiver Web Prozesse auf dem Server.

Maximale gleichzeitige Web Prozesse

Die Option **maximale gleichzeitige Web Prozesse** gibt die Obergrenze für die gleichzeitig auf dem Server geöffneten Web Prozesse an. Sie definiert die maximale Anzahl aller Web Prozesse oder die zum "Pool" an Prozessen gehören. Dieser Parameter verhindert die Überlastung des 4D Server, die bei massiver Anzahl an Anfragen eintreten kann.

Standardmäßig ist der Wert 100 vorgegeben. Sie können jede Zahl zwischen 10 und 32000 eingeben.

Ist die maximale Anzahl konkurrierender Web Prozesse (minus eins) erreicht, erstellt 4D keine neuen Prozesse und sendet für jede neue Anfrage die Meldung "Server nicht verfügbar" (Status HTTP 503 – Service Unavailable).

Hinweis: Sie können die maximale Anzahl Web Prozesse über den Befehl **WEB SET OPTION** festlegen.

Passenden Wert bestimmen

In der Theorie ergibt sich die maximale Anzahl der Web Prozesse aus der Formel: Zugeteilter Arbeitsspeicher/Speichergröße des Web Prozesses (stack size) *

Sie können sich aber auch die Information zu den Web Prozessen ansehen, die der Runtime Explorer anzeigt: aktuelle Anzahl der Web Prozesse und erreichtes Maximum, seit der Web Server in Betrieb ist.

* 4D weist einen Web Prozess als Speichergröße ca. 512 KB für 64-bit Versionen und ca. 256 KB für 32-bit Versionen zu. Diese Werte können je nach Kontext variieren.

Automatische Session-Verwaltung

Aktiviert oder deaktiviert den internen Mechanismus zur automatischen Verwaltung von Benutzersitzungen durch den 4D HTTP Server. Weitere Informationen dazu finden Sie im Abschnitt **Web Sessions verwalten**.

Diese Funktionsweise wird in Anwendungen, die mit 4D v13 oder höher erstellt werden, automatisch aktiviert.

Zur Wahrung der Kompatibilität wird sie jedoch in Anwendungen, die aus früheren 4D Versionen konvertiert wurden, deaktiviert. Sie müssen sie explizit aktivieren, damit Sie die Vorteile der automatischen Verwaltung nutzen können.

Ist diese Option markiert, ist die Option "Web Prozesse erneut verwenden" automatisch markiert (und gesperrt).

Temporären Kontext wiederverwenden (4D in remote Modus)

Diese Option optimiert die Operation des Web Server im remote Modus. Dabei werden Web Prozesse, die zum Verwalten vorheriger Web Anfragen erstellt wurden, wiederhergestellt. Der Web Server von 4D Client benötigt zum Verwalten einer Web Anfrage einen spezifischen Web Prozess; bei Bedarf meldet sich dieser Prozess an den 4D Server Rechner an, um auf die Daten und die Engine der Datenbank zuzugreifen. Er generiert dann über eigene Variablen, Auswahlen, etc. einen temporären Kontext. Dieser Prozess wird vernichtet, wenn die Anfrage abgewickelt ist.

Ist die Option **Web Prozesse erneut verwenden** markiert, behält 4D die spezifischen Web Prozesse auf dem Client-Rechner bei und verwendet sie für nachfolgende Anfragen. Da der Prozess nicht neu aufgebaut werden muss, hat der Web Server eine bessere Performance. Sie müssen jedoch sicherstellen, dass dann die in 4D Methoden verwendeten Variablen systematisch initialisiert werden, damit keine unpassenden Ergebnisse geliefert werden. Außerdem müssen Sie alle aktuellen Auswahlen bzw. Datensätze aus der vorigen Anfrage löschen.

Preemptive Prozesse verwenden

Erlaubt preemptive Web Prozesse in Ihren kompilierten Anwendungen (nur 64-bit Versionen). Ist **Verwende preemptive Prozesse** ausgewählt, wird während dem Kompilieren gewertet, ob Ihr Web-Code, inkl. 4D Tags und Web Datenbankmethoden, preemptive ausführbar ist. Weitere Informationen dazu finden Sie im Abschnitt **Preemptive Web Prozesse verwenden**.

Web Kennwörter

BASIC Modus und DIGEST Modus

In den Einstellungen der Datenbank können Sie das Zugriffssystem für Ihren Web Server einrichten. Es gibt zwei Authentifizierungsmodi: BASIC Modus und DIGEST Modus. Je nach gewähltem Modus wird die Information zu Benutzername und Kennwort unterschiedlich gesammelt und bearbeitet:

- Im BASIC Modus werden vom Benutzer eingegebener Name und Kennwort unverschlüsselt mit der HTTP Anfrage gesendet. Das bietet keine umfassende Systemsicherheit, da diese Information abgefangen und von dritter Stelle weiter verwendet werden kann.
- Der Digest Modus bietet eine höhere Sicherheitsstufe, da die Information durch einen nicht umkehrbaren Prozess, genannt Hashing, bearbeitet wird. Auf diese Weise ist es unmöglich, den jeweiligen Inhalt zu dechiffrieren.

Für den Benutzer ist der Authentifizierungsmodus unsichtbar.

Hinweise:

- Zur Wahrung der Kompatibilität verwenden 4D Datenbanken, die in Version 11 konvertiert wurden, standardmäßig den Basismodus zur Authentifizierung (wenn die Option "Benutze Kennwörter" in der vorigen Version markiert war). Sie müssen den Digest Modus explizit aktivieren.
- Die Digest Authentifizierung ist eine HTTP 1.1. Funktion und wird nicht von allen Browsern unterstützt. So akzeptiert Microsoft Internet Explorer diesen Modus ab Version 5.0. Sendet ein Browser, der diesen Modus nicht unterstützt, eine Anfrage an einen Web Server bei aktivierter Digest Authentifizierung, weist der Server die Anfrage zurück und sendet eine Fehlermeldung an den Browser.

Im Dialogfenster Datenbank-Eigenschaften legen Sie fest, welche Zugriffsoptionen für Ihren Web Server gelten sollen. Wählen Sie dazu unter dem Thema **Web** die Seite **Optionen I**:

H-Listbox - Datenbank-Eigenschaften

Algemein Oberfläche Compiler Datenbank Verschieben Backup Client-Server Web SQL PHP Sicherheit

Konfiguration Optionen (I) Optionen (II) Log (Typ) Log (Backup) Web Services

Cache

Benutze 4D Web Cache

Seiten-Cache Größe: 524288 KB

Web Prozess

Timeout für inaktive Web-Prozesse: Nichts 5 min 15 min 30 min 1 h Unbegrenzt

Maximale gleichzeitige Web Prozesse: 100

Automatische Session Verwaltung

Web Prozesse erneut verwenden

Web Kennwörter

Keine Kennwörter

Kennwörter mit BASIC Protokoll

Verwende 4D Kennwörter

Kennwörter mit DIGEST Protokoll

Allgemeiner Web-Anwender: Designer

Im Bereich "Web Kennwörter" gibt es folgende Optionen:

- **Keine Kennwörter:** Damit wird für Verbindungen an den Web Server keine Authentifizierung durchgeführt. Dann gilt folgendes:
 - Ist die **Datenbankmethode On Web Authentication** vorhanden, wird sie ausgeführt. Zusätzlich zu \$1 und \$2 werden die IP-Adressen von Browser und Server (\$3 und \$4) geliefert, Benutzername und Kennwort (\$5 und \$6) sind leer. In diesem Fall können Sie die Verbindungen anhand der IP-Adresse des Browsers bzw. der angeforderten IP-Adresse des Servers filtern.
 - Ist die **Datenbankmethode On Web Authentication** nicht vorhanden, werden Verbindungen automatisch akzeptiert.

- **Kennwörter mit BASIC Protokoll:** Standard Authentifizierung im Basis Modus. Meldet sich ein Benutzer an den Server an, erscheint ein Dialogfenster auf dem Browser, wo er Benutzername und Kennwort eingibt. Diese beiden Werte sowie die Verbindungsparameter (IP Adresse und Port, URL...) werden an die **Datenbankmethode On Web Authentication** gesendet, die nun ablaufen kann.
Dieser Modus bietet Zugriff auf die Option **Benutze 4D Kennwörter**, so dass Sie an Stelle oder zusätzlich zu Ihrem eigenen Kennwortsystem das Kennwortsystem der 4D Datenbank (wie in 4D definiert) nutzen können.
- **Kennwörter mit DIGEST Protokoll:** Authentifizierung im Digest Modus. Benutzer müssen, analog zum BASIC Modus, Name und Kennwort eingeben. Diese beiden Werte sowie die Verbindungsparameter (IP Adresse und Port, URL...) werden verschlüsselt an die **Datenbankmethode On Web Authentication** gesendet. Sie müssen einen Benutzer mit der 4D Funktion **WEB Validate Digest** authentifizieren.

Hinweise:

- Sie müssen den Web Server neu starten, damit diese Parameter berücksichtigt werden.
- Beachten Sie beim Web Server von 4D Client, dass alle Sites, die Client-Rechner veröffentlichen, auf dieselbe Benutzerliste zugreifen. Die Bestätigung von Benutzern/Kennwörtern erfolgt über die 4D Server Anwendung.

BASIC Modus: Kombination von Kennwörtern und der Datenbankmethode On Web Authentication

In diesem Modus richtet sich das System, das die Verbindungen zum 4D Web Server filtert, nach der Kombination der beiden Parameter:

- Die Option **Web Kennwörter** im Dialogfenster Datenbank-Eigenschaften
- Vorhandensein der **Datenbankmethode On Web Authentication**

Sie erhalten je nach Einstellung unterschiedliche Ergebnisse:

Die Option "Kennwörter mit BASIC Protokoll" ist ausgewählt, "Verwende 4D Kennwörter" ist nicht ausgewählt

- Gibt es die **Datenbankmethode On Web Authentication**, wird sie ausgeführt und alle Parameter werden zurückgegeben. So können Sie die Verbindungen genauer filtern, d.h. nach Benutzername, Kennwort und/oder IP-Adressen der Browser oder Web Server.
- Gibt es die **Datenbankmethode On Web Authentication** nicht, wird die Verbindung automatisch zurückgewiesen. Der Browser erhält eine Meldung, dass die Datenbankmethode nicht vorhanden ist.

Hinweis: Ist der vom Browser gesendete Benutzername ein leerer String und gibt es die **Datenbankmethode On Web Authentication** nicht, erhält der Browser den Kennwortdialog.

Die Optionen "Kennwörter mit BASIC Protokoll " und "Verwende 4D Kennwörter" sind ausgewählt

- Gibt es den vom Browser gesendeten Benutzernamen in der Tabelle der 4D Benutzer und ist das Kennwort korrekt, wird die Verbindung angenommen. Ist das Kennwort falsch, wird die Verbindung zurückgewiesen.
- Gibt es den vom Browser gesendeten Benutzernamen nicht in 4D, sind zwei Ergebnisse möglich:
 - Gibt es die **Datenbankmethode On Web Authentication**, werden die Parameter \$1, \$2, \$3, \$4, \$5 und \$6 zurückgegeben. Sie können dann die Verbindungen nach Benutzername, Kennwort und/oder IP Adresse des Browsers oder Web Servers filtern.
 - Gibt es die **Datenbankmethode On Web Authentication** nicht, wird die Verbindung zurückgewiesen.

DIGEST Modus

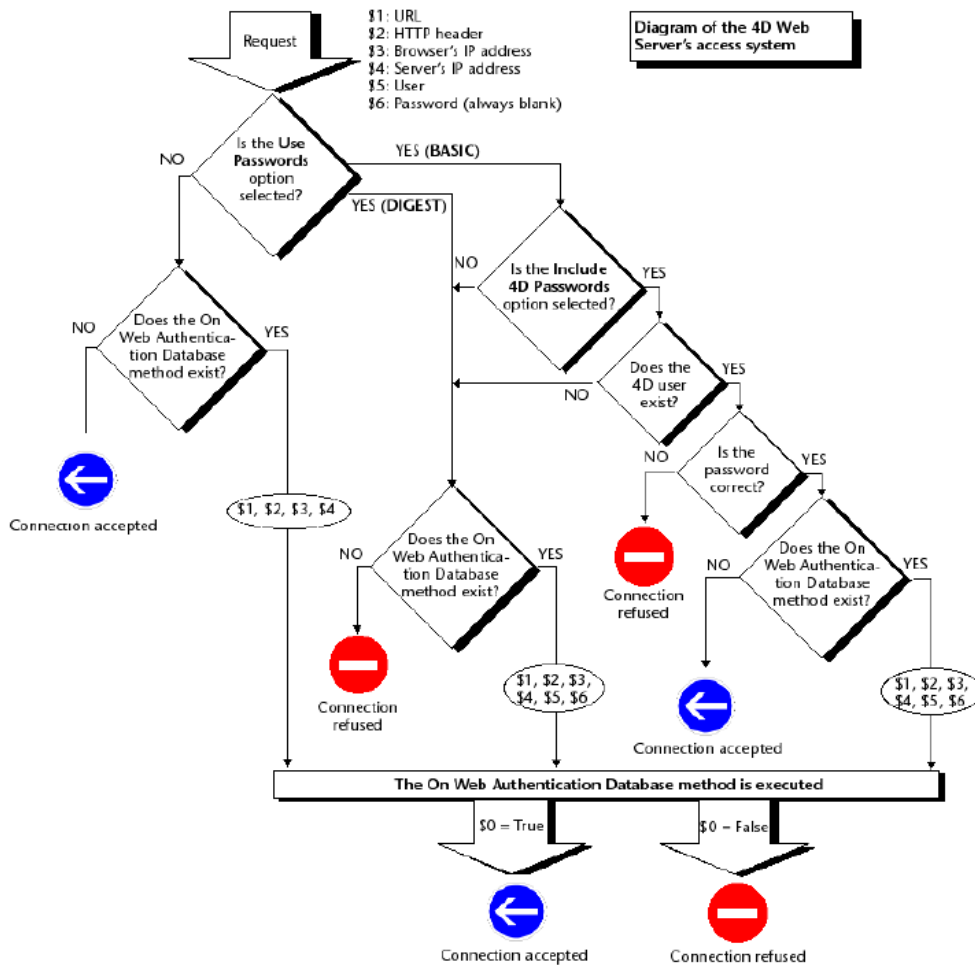
Der DIGEST Modus ist, im Gegensatz zum BASIC Modus, nicht mit den standardmäßigen 4D Kennwörtern kompatibel: Sie können 4D Kennwörter nicht als Web-Kennung verwenden. Bei diesem Modus erscheint die Option "4D Kennwörter einfügen" in Grauschrift.

Sie müssen die Kennung für Web Benutzer selbst verwalten, z.B. über eine Tabelle.

Ist der DIGEST Modus aktiviert, wird der Parameter \$6 (Kennwort) in der **Datenbankmethode On Web Authentication** immer leer zurückgegeben. In diesem Modus läuft diese Information nicht als Klartext (unverschlüsselt) über das Netz.

Deshalb müssen Verbindungsanfragen unbedingt mit der 4D Funktion **WEB Validate Digest** ausgewertet werden

Die Funktionsweise des 4D Web Server Zugriffssystems im Überblick:



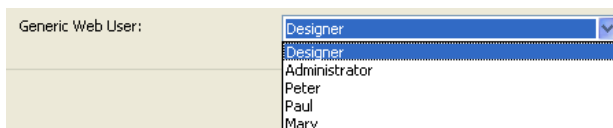
Allgemeiner Web-Anwender

Sie können einen Benutzer, der zuvor in der 4D Kennworttabelle definiert wurde, als allgemeinen Web-Anwender festlegen. In diesem Fall kann jeder Browser, der die Verbindung zur Datenbank herstellt, die diesem generischen Benutzer zugeordneten Zugriffsrechte und Einschränkungen nutzen. So können Sie den Zugriff des Browsers auf die verschiedenen Teile der Datenbank einfach steuern.

Hinweis: Verwechseln Sie diese Option nicht mit dem Steuerungssystem der Web Server Verbindung. Mit "allgemeiner Web Anwender" können Sie den Zugriff des Browsers auf verschiedene Teile der Datenbank einschränken (Tabellen, Menüs, etc.). Das Steuerungssystem der Web Server Verbindung arbeitet mit dem Kennwortsystem und der **Datenbankmethode On Web Authentication**.

Um einen allgemeinen Web-Anwender zu definieren, gehen Sie folgendermaßen vor:

1. Legen Sie im Designmodus im Kennworteditor einen Benutzer an. Sie können dem Benutzer bei Bedarf ein Kennwort zuordnen.
2. Legen Sie in den verschiedenen 4D Editoren die Zugriffsrechte bzw. -einschränkungen für diesen Benutzer fest.
3. Klicken Sie im Dialogfenster Datenbank-Eigenschaften unter dem Thema **Web** auf die Seite **Optionen (I)**. Standardmäßig ist der Designer der allgemeine Web-Anwender, die Browser können auf die gesamte Datenbank zugreifen.
4. Wählen Sie einen Benutzer in der Liste "Allgemeiner Web-Anwender" und bestätigen Sie den Dialog.



Alle Web Browser, die berechtigt sind, sich an die Datenbank anzumelden, können die diesem allgemeinen Web-Anwender zugewiesenen Zugriffsrechte bzw. -einschränkungen nutzen. Das gilt nur dann nicht, wenn die Optionen BASIC Modus und "Verwende 4D Kennwörter" markiert sind und der Benutzer, der die Verbindung herstellt, in der 4D Kennworttabelle nicht vorhanden ist. (siehe unten).

Interaktion mit dem BASIC Protokoll

Die Option "Kennwörter mit BASIC Protokoll" hat keinen Einfluss auf die Funktionsweise des allgemeinen Web-Anwenders. Unabhängig von dieser Option gelten alle Zugriffsrechte bzw. -einschränkungen des "allgemeinen Web-Anwenders" für alle Web Browser, die berechtigt sind, sich an die Datenbank anzuschließen.

Ist dagegen die Option "Verwende 4D Kennwörter" ausgewählt, gibt es zwei Möglichkeiten:

- Es gibt Benutzername und Kennwort NICHT in der Kennworttabelle von 4D. Wird nun die Verbindung von der **Datenbankmethode On Web Authentication** angenommen, gelten die Zugriffsrechte des allgemeinen Web-Anwenders für den Browser.

- Es gibt Benutzername und Kennwort in der Kennworttabelle von 4D. Dann wird der Parameter "allgemeiner Web-Anwender" ignoriert. Der Benutzer meldet sich mit seinen eigenen Zugriffsrechten an.

Text Konvertierung

Erweiterte Zeichen direkt senden

Der 4D Web Server konvertiert die erweiterten Zeichen in dynamischen und statischen Web Seiten standardmäßig nach den HTML Standards, bevor er diese Seiten sendet. Browser können diese Zeichen dann interpretieren.

Sie können den Web Server aber auch so einstellen, dass Zeichen des erweiterten Satzes ohne Konvertierung in HTML Einheiten gesendet werden. Tests haben bei den meisten ausländischen Betriebssystemen eine Beschleunigung ergeben, insbesondere beim japanischen System. Damit der 4D Web Server erweiterte Zeichen direkt senden kann, wählen Sie die Option "Sende erweiterte Zeichen direkt".

Standard Zeichensätze

Markieren Sie die Option **Standardsatz**, um einen Zeichensatz für die Verwendung im 4D Web Server auszuwählen. Standardmäßig wird der Zeichensatz UTF-8 verwendet.

Hinweis: Diese Einstellung wird auch zum Generieren von Schnellberichten im HTML Format verwendet (siehe Abschnitt **Schnellbericht ausführen**).

Keep-Alive Verbindungen

Der Web Server von 4D kann keep-alive Verbindungen verwenden. So können Sie für den Austausch der Elemente zwischen Web Browser und Server eine einzige TCP Verbindung offenhalten. Das spart System-Ressourcen und optimiert die Übertragung.

Diese Option ermöglicht, das Aufrechterhalten von TCP Verbindungen für den Web Server zu aktivieren/deaktivieren. Sie ist standardmäßig aktiviert. Das ist für die meisten Fälle sinnvoll, das es den Austausch beschleunigt. Unterstützt der Web Browser nicht die Option keep-alive, wechselt der 4D Web Server automatisch zu HTTP/1.0.

Die Option keep-alive des 4D Web Server gilt für alle TCP/IP Verbindungen (HTTP, HTTPS). Beachten Sie jedoch, dass keep-alive Verbindungen nicht immer für alle 4D Web Prozesse verwendet werden. In einigen Fällen werden andere optimierte interne Funktionen ausgelöst. Keep-alive Verbindungen sind hauptsächlich für statische Seiten sinnvoll.

Es gibt folgende Optionen:

- **Anzahl Anfragen pro Verbindung:** Damit legen Sie die maximale Anzahl von Anfragen und Antworten fest, die keep-alive nutzen können. Diese Einschränkung verhindert, dass der Server durch eine hohe Anzahl hereinkommender Anfragen überflutet wird – eine von Hackern benutzte Technik. Standardmäßig ist der Wert 100 vorgegeben. Sie können ihn je nach den Ressourcen des Rechners, welcher den 4D Web Server hostet, erhöhen oder verringern.
- **Timeout:** Damit legen Sie die maximale Zeitspanne (in Sekunden) fest, für die der Web Server eine TCP Verbindung ohne Anfragen vom Web Browser offenhält. Ist die Zeit abgelaufen, schließt der Server die Verbindung. Sendet der Web Browser eine Anfrage nach Schließen der Verbindung, wird automatisch eine neue TCP Verbindung angelegt. Dieser Vorgang ist für den Benutzer unsichtbar.

Seite Web/ Log (Typ)

Log-Datei der Verbindung

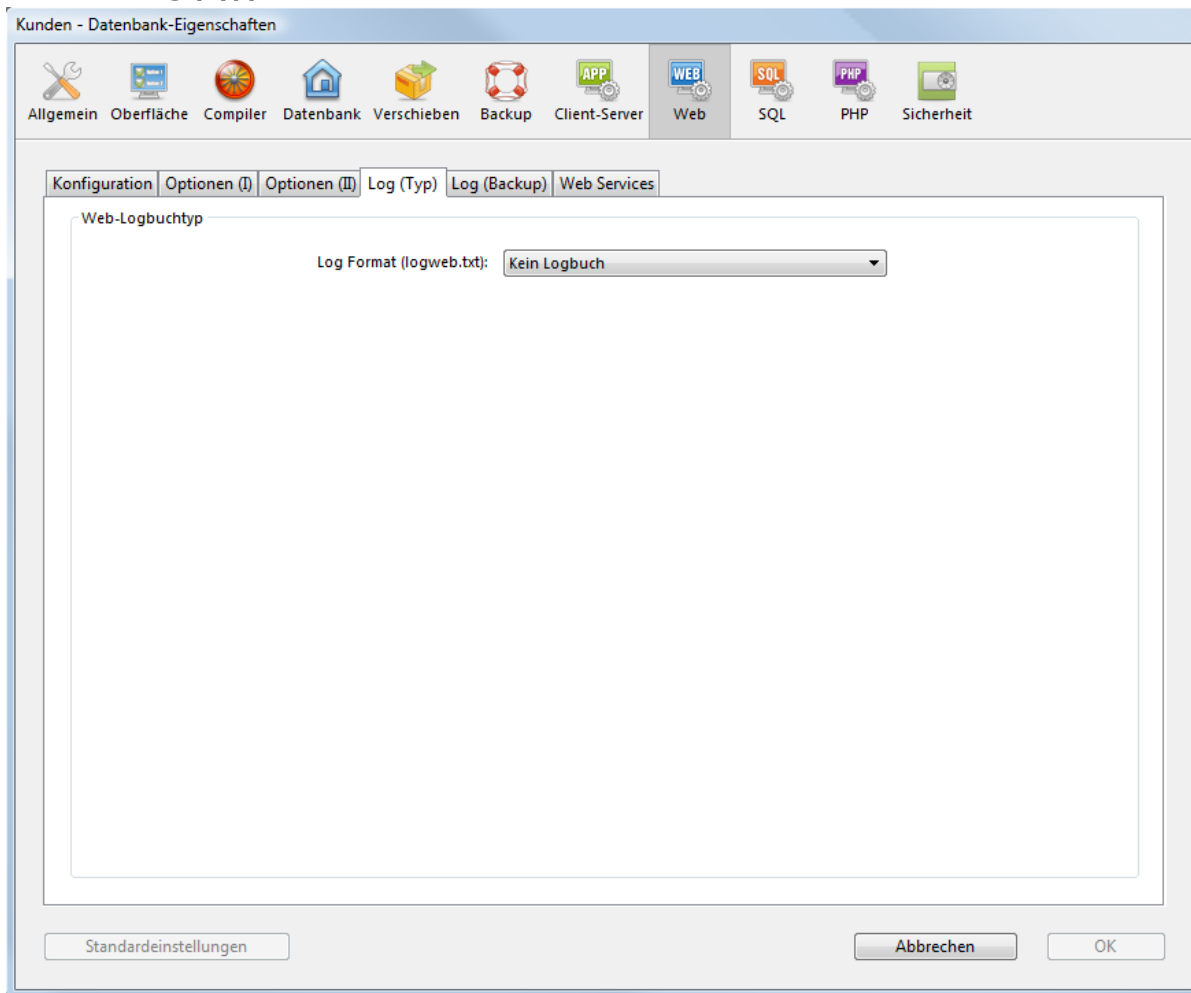
In 4D erhalten Sie eine Log-Datei der Anfragen.

Die Datei hat den Namen "weblog.txt" und wird automatisch abgelegt:

- In 4D im lokalen Modus und 4D Server im Ordner **Logs** neben der Strukturdatei der Anwendung
- In 4D im remote Modus im Unterordner **Logs** des Anwendungsordners der Arbeitsstation (Cache Ordner).

Aktivierung und Format

Die Aktivierung des Logbuchs und die Konfiguration seines Inhalts definieren Sie in den Datenbank-Eigenschaften auf der Seite **Web>Log (Typ)**:



Hinweis: Sie können das Web Logbuch auch per Programmierung über den Befehl **SET DATABASE PARAMETER** (4D v12) bzw. **WEB SET OPTION** (4D v13 und höher) aktivieren und deaktivieren.

Das PopUp-Menü Log Format im oberen Bereich bietet folgende Optionen:

- **Kein Logbuch:** Ist diese Option gewählt, erzeugt 4D kein Web Logbuch.
- **CLF (Common Log Format):** Ist diese Option gewählt, wird das Web Logbuch im Format CLF erstellt. Im Format CLF Format stellt jede Zeile der Datei eine Anfrage dar, wie z.B.:
`host rfc931 user [DD/MMM/YYYY:HH:MM:SS] "query" state length`
Auf jedes Feld folgt ein Leerzeichen, jede Zeile endet mit der Folge CR/LF (Zeichen 13, Zeichen 10).
 - host: IP Adresse des Client (z.B. 192.100.100.10)
 - rfc931: Information, die nicht 4D generiert, ist immer - (Minuszeichen)
 - user: Benutzername wie zugelassen, oder - (Minuszeichen). Leerzeichen in Benutzernamen werden durch _ (Unterstrich) ersetzt.
 - DD: Tag, MMM: Monatsname, abgekürzt mit drei Buchstaben (Jan, Feb,...), YYYY: Jahr, HH: Stunde, MM: Minuten, SS: Sekunden

Datum und Uhrzeit richten sich nach dem jeweiligen Server.

- request: Vom Benutzer gesendete Anfrage (z.B. GET /index.htm HTTP/1.0)
- state: Vom Server zurückgegebene Antwort.
- length: Größe der zurückgegebenen Daten (ohne HTTP Kopfteil) oder 0.

Hinweis: Diese Operationen werden zur Wahrung der Performance vor Übertragen auf die Festplatte im Pufferspeicher in 1 Kb Paketen gesichert. Sie werden auch auf die Festplatte geschrieben, wenn nicht innerhalb der nächsten 5

Sekunden eine Anfrage gesendet wird. Für den Status sind folgende Werte möglich:

200: OK
204: Kein Inhalt
302: Redirektion
304: Nicht geändert
400: Inkorrekte Anfrage
401: Authentifizierung erforderlich
404: Nicht gefunden
500: Interner Fehler

Das CLF Format lässt sich nicht anpassen.

- **DLF (Combined Log Format):** Ist diese Option gewählt, wird das Web Logbuch im Format DLF erzeugt. Das DLF Format ähnelt dem CLF Format und verwendet genau dieselbe Struktur. Es fügt lediglich am Ende jeder Anfrage zwei weitere HTTP Felder hinzu: referer und user-agent.
 - Referer: Enthält die URL der Seite, die auf das angefragte Dokument zeigt.
 - user-agent: Enthält Name und Version von Browser oder Software des Benutzers, der die Anfrage gestartet hat.Das DLF Format lässt sich nicht anpassen.
- **ELF (Extended Log Format):** Ist diese Option gewählt, wird das Web Logbuch im Format ELF erzeugt. Dieses Format ist in der Welt der HTTP Browser weitverbreitet. Sie können damit ausgeklügelte Logbücher für spezielle Zwecke einrichten. Das ELF Format lässt sich also anpassen: Sie können auswählen, welche Felder gespeichert werden und in welcher Reihenfolge sie in die Datei eingefügt werden.
- **WLF (Webstar Log Format):** Ist diese Option gewählt, wird das Web Logbuch im Format WLF erzeugt. Dieses Format wurde speziell für den WebSTAR Server entwickelt. Es ähnelt dem ELF Format, und hat lediglich ein paar zusätzliche Felder. Es lässt sich ebenfalls anpassen.

Felder konfigurieren

Wählen Sie das Format ELF (Extended Log Format) oder WLF (WebStar Log Format), zeigt der Bereich "Web Log Token Auswahl" die für das gewählte Format verfügbaren Felder. Sie müssen jedes Feld auswählen, das in das Logbuch übernommen werden soll.

Hinweis: Sie können dasselbe Feld nicht zweimal auswählen

Nachfolgende Tabelle zeigt die für jedes Format verfügbare Felder in alphabetischer Reihenfolge und beschreibt den Inhalt:

Feld	ELF	WLF	Wert
BYTES_RECEIVED		X	Anzahl der vom Server empfangenen Bytes
BYTES_SENT	X	X	Anzahl der vom Server an den Client gesendeten Bytes
C_DNS	X	X	IP Adresse des DNS (ELF: Feld ist identisch zu C_IP field)
C_IP	X	X	IP Adresse des Client (z.B. 192.100.100.10)
CONNECTION_ID		X	ID Nummer der Verbindung
CS(COOKIE)	X	X	Information über Cookies in der HTTP Anfrage
CS(HOST)	X	X	Host Feld der HTTP Anfrage
CS(REFERER)	X	X	URL der Seite, die auf das angeforderte Dokument zeigt
CS(USER_AGENT)	X	X	Information über Software und Betriebssystem des Client
CS_SIP	X	X	IP Adresse des Server
CS_URI	X	X	URI auf der Anfrage erstellt wurde
CS_URI_QUERY	X	X	Suchparameter der Anfrage
CS_URI_STEM	X	X	Teil der Anfrage ohne Suchparameter
DATE	X	X	DD: Tag, MMM: Abkürzung für Monat mit drei Buchstaben (Jan, Feb, etc.), YYYY: Jahr
METHOD	X	X	HTTP Methode für die Anfrage an den Server
PATH_ARGS		X	CGI Parameter: String nach dem Zeichen "\$"
STATUS	X	X	Vom Server gelieferte Antwort
TIME	X	X	HH: Stunde, MM: Minuten, SS: Sekunden
TRANSFER_TIME	X	X	Vom Server angefragte Zeit zum Erstellen der Antwort
USER	X	X	Benutzername wenn authentifiziert; sonst - (Minuszeichen) Enthält der Benutzername Abstände, werden sie durch Unterstriche ersetzt
URL	X		Vom Client angefragte URL

Hinweis: Datum und Zeit werden in GMT (Greenwich Mean Time) angegeben.

Backup Parameter

In den Datenbank-Eigenschaften auf der Seite **Web>Log (Backup)** können Sie automatische Backup Parameter für das Web Logbuch definieren:

Sie müssen zuerst durch Anklicken des entsprechenden Optionsfeldes das Backup-Intervall, also Tage, Wochen, etc. oder die max. Dateigröße festlegen. Dann definieren Sie den gewünschten Zeitpunkt für das Backup.

- **Kein Backup:** Damit ist ein planmäßiges Backup deaktiviert.
- **Alle X Stunden:** Damit wird ein Backup nach der definierten Stundenanzahl durchgeführt. Sie können Werte von 1 bis 24 eingeben.
 - **Start um:** Hier legen Sie die Zeit fest, zu der das erste Backup starten soll.
- **Alle X Tage um X:** Damit wird ein Backup nach der definierten Tagesanzahl durchgeführt. Tragen Sie 1 ein, wenn es täglich durchgeführt werden soll. Sie müssen auch die Zeit festlegen, zu der das erste Backup starten soll.
- **Alle X Wochen, Wochentag, um X:** Damit wird ein Backup nach der definierten Wochenanzahl durchgeführt. Tragen Sie 1 ein, wenn es wöchentlich durchgeführt werden soll. Sie müssen auch den Wochentag und die Zeit festlegen, zu der das Backup jeweils starten soll. Bei Bedarf können Sie auch mehrere Wochentage auswählen. Sie können z.B. zwei wöchentliche Backups festlegen, einmal am Mittwoch, einmal am Freitag.
- **Alle X Monate, X. Tag um X:** Damit wird ein Backup nach der definierten Monatsanzahl durchgeführt. Tragen Sie 1 ein, wenn es monatlich durchgeführt werden soll. Sie müssen auch den Tag und die Zeit festlegen, zu der das Backup jeweils starten soll.
- **Alle X MB:** Damit wird ein Backup nach der festgelegten Größe des aktuellen Web Logbuchs durchgeführt. Das Backup startet automatisch, wenn die Datei die definierte Größe erreicht. Sie können die Grenze 1, 10, 100 oder 1000 MB eintragen.

Hinweis: Ist ein planmäßiges Backup festgelegt und war der Web Server zum vorgesehenen Termin nicht gestartet, betrachtet 4D beim nächsten Start das Backup als gescheitert und führt die entsprechenden Operationen aus, so wie sie in den Datenbank-Eigenschaften festgelegt wurden.

Über die Optionen dieser Seite können Sie Web Services für die 4D Datenbank aktivieren und konfigurieren, sowohl zum Veröffentlichen (Server Seite) als auch zum Abonnieren (Client Seite).

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Web Services publizieren und einsetzen**.

Auf Server

Dieser Bereich enthält verschiedene Optionen, um 4D als einen „Server“ für Web Services einzusetzen, z.B. um Projektmethoden als Web Services zu veröffentlichen.

- **Web Service Abfragen erlauben:** Mit dieser Option aktivieren Sie die Veröffentlichung von Web Services. Ist diese Option inaktiv, verweigert 4D SOAP Anfragen und erzeugt kein WSDL — selbst wenn die Methoden die Eigenschaft **Zugang per Web Service** haben. Ist diese Option markiert, erstellt 4D die WSDL Datei.
 - **Web Service Name:** Hier können Sie den vorgegebenen Namen des Web Service ändern. Er dient zur Unterscheidung der Dienste auf Ebene des SOAP Server (wenn der Server mehrere Web Services veröffentlicht) und in den Verzeichnissen der Web Services. 4D verwendet standardmäßig den Namen A_WebService.
 - **Web Services Namespace:** Hier können Sie den vorgegebenen Namensraum der von 4D veröffentlichten Web Services verändern. Jeder im Internet veröffentlichte Web Service muss einmalig sein. Diese Einmaligkeit wird durch XML Namensräume sichergestellt. Ein Namensraum ist eine willkürliche Zeichenfolge, um einen Satz von XML Tags als einmalig zu identifizieren. In der Regel beginnt ein Namensraum mit der URL der Firma, also `http://meinefirma.com/meinnamensraum`. Die URL muss in diesem Fall weiter nichts besonderes enthalten; wichtig ist, dass die verwendete Zeichenfolge einmalig ist.
4D verwendet standardmäßig den Namensraum: `http://www.4d.com/namespace/default`.
- Hinweis:** In Übereinstimmung mit dem XML Standard für Tag-Namen dürfen die Zeichenfolgen weder Leerzeichen enthalten, noch mit einer Nummer beginnen. Außerdem empfehlen wir, zur Vermeidung von Konflikten nur Zeichen des Latin Zeichensatzes zu verwenden ([A-Za-z0-9._] | '-')*.

Auf Client

Dieser Bereich enthält verschiedene Optionen, um 4D als „Client“ für Web Services einzusetzen, also im Netzwerk veröffentlichte Dienste zu nutzen.

- **Methoden Präfix für Assistent:** Hier können Sie die Vorsilbe ändern, die 4D automatisch dem Namen der Proxy-Methode voranstellt, die der Web Services Assistent erzeugt. Proxy Projektmethoden bilden eine Verknüpfung zwischen dem Programm 4D und dem Server für Web Services.
4D verwendet standardmäßig die Vorsilbe „proxy_“.

Diese Seite enthält Optionen, um 4D Mobile Zugriffe (via REST) für 4D Anwendungen zu aktivieren und zu steuern. Ist der Zugriff über 4D Mobile aktiviert, können Wakanda Applikationen direkt auf die Daten Ihrer 4D Anwendung zugreifen. Weitere Informationen dazu finden Sie im Handbuch **4D Mobile**.

4D Mobile Services aktivieren

Aus Sicherheitsgründen reagiert 4D standardmäßig nicht auf 4D Mobile Anfragen (via REST). Sie müssen die Option **Aktiviere 4D Mobile Service** markieren, damit solche Anfragen bearbeitet werden und Wakanda / 4D Mobile eingerichtet werden kann.

Hinweis: 4D Mobile Services verwenden den 4D HTTP Server. Sie müssen also sicherstellen, dass der 4D Web Server bzw. 4D Server gestartet ist.

Die Meldung "Achtung, prüfen Sie die Zugriffsberechtigungen" erscheint, um Sie darauf aufmerksam zu machen, dass durch Aktivieren von 4D Mobile Services die Datenbankobjekte standardmäßig frei zugänglich sind, solange die 4D Mobile Zugriffe nicht gesteuert werden.

Zugriff Lesen/Schreiben

Diese Option definiert die Gruppe 4D Benutzer, die berechtigt ist, über die Wakanda Applikation die Verbindung zu 4D Server herzustellen.

Dieses Menü zeigt standardmäßig **<Jeder>** an, d.h. 4D Mobile Zugriffe sind für alle Benutzer offen.

Haben Sie eine Gruppe definiert, können nur 4D Benutzer, die zu dieser Gruppe gehören, über eine 4D Mobile Anfrage auf 4D zugreifen -- insbesondere, um über die Methode *mergeOutsideCatalog()* eine Session auf 4D Server zu öffnen. Wird ein Account verwendet, das nicht zu dieser Gruppe gehört, gibt 4D an den Absender der Anfrage einen Authentifizierungsfehler zurück.

Diese Einstellung greift nur, wenn folgende Bedingungen erfüllt sind:

- Das 4D Kennwortsystem muss aktiviert sein, d.h. der Designer muss ein Kennwort haben (siehe **Kennwortsystem aktivieren**),
- Die **Datenbankmethode On 4D Mobile Authentication** darf nicht definiert sein. Da diese Methode vorrangig ist, berücksichtigt 4D dann nicht die in den Datenbank-Eigenschaften festgelegten Zugriffsrechte.

Auf dieser Seite legen Sie die Standardeinstellungen beim Starten und die Zugriffsrechte für den integrierten SQL Server sowie die Optionen für die SQL Engine fest. Weitere Information dazu finden Sie im Handbuch *SQL Reference* im Kapitel **Erweiterte Command Line Interface (CLI)**.

Einstellungen zum Veröffentlichen des SQL Server

Hier legen Sie für den integrierten SQL Server die Standardeinstellungen beim Starten und die Zugriffsrechte fest, sowie die SQL Engine Optionen:

- Über die Option **Automatisch starten** können Sie den SQL Server beim Starten des Programms mitstarten.
- **TCP Port**: Der 4D SQL Server antwortet standardmäßig auf dem TCP Port 19812. Wird er bereits von einem anderen Dienst genutzt oder benötigen Ihre Verbindungsparameter eine andere Einstellung, können Sie den von 4D SQL Server verwendeten TCP Port ändern.
Hinweis: Übergeben Sie 0, verwendet 4D die standardmäßige TCP Port Nummer, z.B. 19812.
- **IP Adresse**: Sie können die IPv4 Adresse des Rechners setzen, auf dem der SQL Server SQL Anfragen bearbeiten soll. Der Server antwortet standardmäßig für alle IP Adressen (Option **Alle**), d.h. IPv4 und IPv6 Adressen. Die DropDown-Liste "IP Adresse" enthält automatisch alle auf dem Rechner vorhandenen IPv4 Adressen. Wählen Sie eine bestimmte Adresse, beantwortet der Server nur Anfragen, die an diese Adresse gesendet werden. Das dient für 4D Anwendungen, die auf Rechnern mit mehreren TCP/IP Adressen gehostet werden.
Hinweise:
 - Um Klarheit zu bewahren, erscheinen IPv6 Adressen nicht in der DropDown-Liste "IP Adresse". Zum Filtern von IPv6 Adressen empfehlen wir, die Firewall Einstellungen des Systems entsprechend zu konfigurieren.
 - Auf dem Client müssen IP Adresse und TCP Port des SQL Server, an den sich das Programm anmeldet, in der Definition der ODBC Datenquelle korrekt definiert werden.
- **SSL aktivieren**: Diese Option gibt an, ob der SQL Server das SSL Protokoll zum Bearbeiten von SQL Verbindungen aktivieren soll. Ist dieses Protokoll aktiviert, müssen Sie an das Ende der IP Adresse des SQL Server die Kombination ":ssl" anhängen, wenn Sie eine Verbindung mit dem Befehl **SQL LOGIN** öffnen. Standardmäßig verwendet der SL Server interne Dateien für SSL Schlüssel und Zertifikat. Sie können aber auch eigene Elemente verwenden. Dazu kopieren Sie ihre eigenen Dateien *key.pem* und *cert.pem* an folgende Stelle: MeineDatenbank/Preferences/SQL („Meine Datenbank“ ist der Ordner bzw. das Paket der Datenbank).
- **Flash Player Anfragen erlauben**: Damit können Sie festlegen, dass der 4D SQL Server Flash Player Anfragen unterstützt. Diese Operation basiert auf der Datei "socketpolicy.xml," die im Ordner Preferences der Datenbank vorhanden sein muss (Preferences/SQL/Flash/). Flash Player benötigt diese Datei, um cross-domain Verbindungen bzw. Verbindungen über sockets von Flex (Web 2.0) Anwendungen zuzulassen. In bisherigen 4D Versionen musste diese Datei manuell hinzugefügt werden. Ab jetzt wird die Aktivierung über die Option **Anfragen erlauben** automatisch ausgeführt. Aktivieren Sie diese Option, werden Flash Player Anfragen angenommen und - falls erforderlich - wird eine generische Datei "socketpolicy.xml" für die Datenbank angelegt. Deaktivieren Sie diese Option, wird die Datei "socketpolicy.xml" deaktiviert (umbenannt). Alle Flash Player Anfragen, die der SQL Server dann empfängt, werden abgewiesen. Beim Öffnen der Datenbank ist die Option aktiv oder inaktiv, je nachdem, ob im Ordner Preferences der Datenbank eine aktive Datei "socketpolicy.xml" liegt.
Hinweis: Über den 4D Befehl **SQL SET OPTION** können Sie die vom SQL Server verwendete Verschlüsselung zum Bearbeiten externer Anfragen setzen.

SQL Access Control for the default schema

For security reasons, it is possible to limit actions that external queries sent to the SQL server can perform in the 4D database.

This can be done at two levels:

- At the level of the type of action allowed,
 - At the level of the user carrying out the query.
- These settings can be made on the **SQL** page of the Database Settings.

Note: You can also use the **Datenbankmethode On SQL Authentication** to control in a custom way any external access to the 4D internal SQL engine.

The parameters set in this dialog box are applied to the default schema. The control of external access to the database is based on the concept of SQL schemas (see the section). If you do not create custom schemas, the default schema will include all the tables of the database. If you create other schemas with specific access rights and associate them with tables, the default schema will only include the tables that are not included in custom schemas.

You can configure three separate types of access to the default schema via the SQL server:

- **"Read Only (Data)"**: Unlimited access to read all the data of the database tables but no adding, modifying or removing of records, nor any modification to the structure of the database is allowed.
- **"Read/Write (Data)"**: Read and write (add, modify and delete) access to all the data of the database tables, but no modification of the database structure is allowed.
- **"Full (Data and Design)"**: Read and write (add, modify and delete) access to all the data of the database tables, as well as modification of the database structure (tables, fields, relations, etc.) is allowed.

You can designate a set of users for each type of access. There are three options available for this purpose:

- **<Nobody>**: If you select this option, the type of access concerned will be refused for any queries, regardless of their origin. This parameter can be used even when the 4D password access management system is not activated.
- **<Everybody>**: If you select this option, the type of access concerned will be allowed for all queries (no limit is applied).
- **Group of users**: This option lets you designate a group of users as exclusively authorized to carry out the type of access concerned. This option requires that 4D passwords be activated. The user at the origin of the queries provides their name and password when connecting to the SQL server.

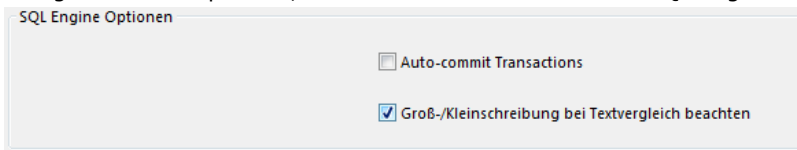
WARNING: Each type of access is set independently from the others. More specifically, if you only assign **Read Only** type access to one group this will not have any effect since this group as well as all the others will continue to benefit from **Read/Write** access (assigned to **<Everybody>** by default). In order to set a **Read Only** type access, you also need to configure the **Read/Write** access.

WARNING: This mechanism is based on 4D passwords. In order for the SQL server access control to come into effect, the 4D password system must be activated (a password must be assigned to the Designer).

Note: An additional security option can be set at the level of each 4D project method. For more information, please refer to the "Available through SQL option" paragraph in the section.

SQL Engine Optionen

Hier gibt es zwei Optionen, um die Funktionsweise der 4D SQL Engine zu verändern.



- **Auto-commit Transactions:** Mit dieser Option können Sie Auto commit in der 4D SQL Engine aktivieren. Dieser Modus sorgt für referentielle Datenintegrität. Ist die Option aktiv, werden alle Suchläufe mit **SELECT, INSERT, UPDATE** und **DELETE** (SIUD), die noch nicht innerhalb einer Transaktion ausgeführt werden, automatisch in eine ad-hoc Transaktion eingesetzt. So ist garantiert, dass die Suchen vollständig ausgeführt oder bei einem Fehler vollständig annulliert werden. Suchen, die bereits in einer Transaktion laufen, werden dabei nicht berücksichtigt (individuelle Verwaltung der Datenintegrität).
Ist diese Option nicht markiert, werden keine automatischen Transaktionen erzeugt. Davon ausgenommen sind Suchläufe mit **SELECT... FOR UPDATE** (siehe SQL Befehl *SELECT*). Diese Option ist standardmäßig nicht markiert. Über den 4D Befehl **SET DATABASE PARAMETER** können Sie die Option auch per Programmierung verwalten.
Hinweis: **Auto commit** gilt nur für lokale Datenbanken, die über die 4D SQL Engine durchsucht werden. Bei externen Datenbanken wird **Auto commit** über remote SQL Engines verwaltet
- **Groß-/Kleinschreibung bei Textvergleich beachten:** Mit dieser Option können Sie die Berücksichtigung von Groß-/Kleinschreibung und Akzenten in SQL Anfragen verändern. Diese Option ist standardmäßig aktiviert, d.h. die SQL Engine unterscheidet beim Vergleichen von Strings zwischen Groß- und Kleinschreibung und Akzenten. Das gilt insbesondere für Sortieren und Suchen. Beispiel: "ABC"="ABC" aber "ABC" # "Abc und abc # âbc."
In bestimmten Fällen, z.B. um die Funktionsweise der SQL Engine an die 4D Engine anzupassen, sollen beim Vergleichen von Strings die Groß- und Kleinschreibung und Akzente nicht berücksichtigt werden ("ABC"="Abc"="âbc"). Dazu deaktivieren Sie einfach diese Option in den Datenbank-Eigenschaften.
Über den 4D Befehl **SET DATABASE PARAMETER** können Sie die Option auch per Programmierung verwalten.

Auf der Seite PHP können Sie Optionen wählen, um in 4D PHP Skripte direkt auszuführen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **PHP Skripte in 4D ausführen**.

Interpreter

- **IP Adresse und Port Nummer**

4D liefert standardmäßig einen PHP Interpreter, kompiliert in FastCGI. Bedingt durch die interne Architektur werden an den PHP Interpreter gerichtete Anfragen zur Ausführung an eine spezifische HTTP Adresse gesendet. 4D verwendet standardmäßig die Adresse 127.0.0.1 und den Port 8002. Hier können Sie die Adresse bzw. den Port ändern, wenn diese z.B. bereits von einem anderen Dienst verwendet werden oder auf demselben Rechner mehrere Interpreter eingerichtet werden sollen.

Beachten Sie, dass die HTTP Adresse auf demselben Rechner wie 4D liegen muss.

- **Externer Interpreter**

Sie können auch einen externen PHP Interpreter verwenden, Er muss jedoch in FastCGI kompiliert sein und auf dem gleichen Rechner wie 4D liegen. Weitere Informationen dazu finden Sie im Abschnitt "Anderen PHP Interpreter oder andere php.ini Datei verwenden" unter **PHP Skripte in 4D ausführen**.

Markieren Sie diese Option, damit 4D beim Ausführen einer PHP Anfrage keine Verbindung mit dem internen Interpreter startet. Beachten Sie, dass Sie mit dieser Konfiguration die Ausführung und Steuerung des externen Interpreters selbst verwalten müssen.

4D Server: Diese Einstellungen werden von 4D Server und Rechnern mit remote 4D gemeinsam genutzt. Von daher ist es nicht möglich, einen externen Interpreter auf dem Server Rechner und gleichzeitig den internen Interpreter auf den Client Rechnern bzw. umgekehrt zu verwenden. Verwendet der Server einen externen Interpreter auf Port 9002, müssen die Client Rechner einen Interpreter auch auf diesem Port verwenden.

Optionen

Diese Optionen betreffen die automatische Verwaltung des mit 4D gelieferten PHP Interpreters. Sie sind inaktiv, wenn die Option **Externer Interpreter** markiert ist.

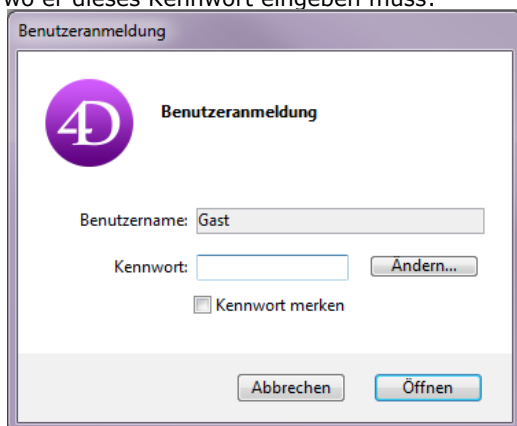
- **Anzahl Prozesse:** Der PHP Interpreter von 4D steuert eine Reihe von Prozessen zur Systemausführung, genannt "Kindprozesse". Zur Optimierung werden standardmäßig bis zu fünf Kindprozesse gleichzeitig gestartet. Über diese Option können Sie die Anzahl der Kindprozesse je nach Ihren Bedürfnissen verändern. Weitere Informationen dazu finden Sie im Abschnitt "Architektur" unter **PHP Skripte in 4D ausführen**.
Hinweis: Auf Mac OS nutzen alle Kindprozesse den gleichen Port. Unter Windows verwendet jeder Kindprozess eine spezifische Port Nummer. Als erste Nummer dient die Nummer, welche der PHP Interpreter setzt, die anderen Kindprozesse erhöhen diese Nummer jeweils. Ist der Standardport z.B. 8002 und starten Sie 5 Kindprozesse, verwenden diese die Port Nummern 8002 bis 8006.
- **Interpreter nach X Anfragen neu starten:** Hier legen Sie die max. Anzahl Anfragen fest, die der PHP Interpreter von 4D akzeptiert. Ist die Höchstzahl erreicht, startet der PHP Interpreter neu. Weitere Informationen dazu finden Sie in der Dokumentation von FastCGI-PHP.
Hinweis: Über dieses Dialogfenster werden die Parameter standardmäßig für alle angemeldeten Rechner und für alle Sitzungen festgelegt. Über die 4D Befehle **SET DATABASE PARAMETER** und **Get database parameter** lassen sich diese Parameter auch für jeden Rechner und jede Sitzung separat ändern bzw. lesen. Dabei sind Parameter, die über den 4D Befehl **SET DATABASE PARAMETER** geändert wurden, für die aktuelle Sitzung vorrangig.

Diese Seite enthält Optionen zum Datenzugriff und zur Sicherheit Ihrer Datenbank.

Hinweis: Einen allgemeinen Überblick über die Sicherheitsvorkehrungen in 4D finden Sie im [4D Security guide](#).

Zugriffskontrolle

- **Design-Zugriff:** Gibt der im DropDown-Menü angegebenen Gruppe Zugriff auf die Designumgebung. Beachten Sie folgendes:
 - Richten Sie eine Zugriffsgruppe in der Designumgebung ein, können Sie im Importeditor die Option **Erzeuge Tabelle** deaktivieren. Weitere Informationen dazu finden Sie im Abschnitt **Daten aus Datei importieren**.
 - Designer und Administrator haben immer Zugriff auf die Designumgebung, auch wenn sie nicht explizit in der jeweiligen Zugriffsgruppe aufgeführt sind.Weitere Informationen zu Benutzern und Gruppen finden Sie im Kapitel **Benutzer und Gruppen**.
- **Filtern von Befehlen und Projektmethoden im Formeleditor und in 4D Write Pro Dokumenten:** Aus Sicherheitsgründen beschränkt 4D standardmäßig den Zugriff auf Befehle, Funktionen und Projektmethoden im **Formeleditor** im Anwendungsmodus und in 4D Write Pro Dokumenten: Hier lassen sich nur solche 4D Funktionen und Projektmethoden verwenden, die explizit über den Befehl **SET ALLOWED METHODS** deklariert wurden. Über folgende Optionen können Sie dieses Filtern komplett oder teilweise entfernen.
 - **Aktivieren für alle (Standardeinstellung):** Der Zugriff auf Befehle, Funktionen und Projektmethoden ist für alle Benutzer im Anwendungsmodus eingeschränkt, inkl. Designer und Administrator.
 - **Deaktivieren für Designer und Administrator:** Diese Option gewährt nur für Designer und Administrator unbegrenzten Zugriff auf 4D Befehle und Methoden im Formeleditor und in 4D Write Pro Dokumenten. Damit können Sie allgemeinen Zugriff auf die Befehle und Methoden setzen bei gleichzeitiger Steuerung der ausgeführten Operationen. Dieser Modus lässt sich in der Entwicklungsphase verwenden, um alle Formeln, Berichte, usw. uneingeschränkt zu testen. Während der Ausführung kann die Option dazu dienen, sichere Lösungen einzurichten, die den Zugriff auf Befehle und Methoden nur vorübergehend erlauben. Dazu verändern Sie den Benutzer über den 4D Befehl **CHANGE CURRENT USER**, bevor Sie ein Dialogfenster aufrufen oder einen Druckvorgang starten, für die der volle Zugriff auf die Befehle erforderlich ist. Ist die spezifische Operation ausgeführt, kehren Sie wieder zum ursprünglichen Benutzer zurück.
Hinweis: Wurde mit der ersten Option voller Zugriff aktiviert, hat diese Option keine Auswirkung.
 - **Deaktivieren für alle:** Diese Option deaktiviert die Steuerung von Benutzeraktionen im Formulareditor und in 4D Write Pro Dokumenten. Ist sie markiert, haben Benutzer Zugriff auf alle 4D Befehle und Plug-Ins, sowie auf alle Projektmethoden (außer, sie sind nicht sichtbar).
 - **Hinweis:** Diese Option ist vorrangig zum Befehl **SET ALLOWED METHODS**. Ist sie aktiv, führt dieser Befehl nichts aus.
- **Standardbenutzer:** Wurde ein Benutzer als Standardbenutzer definiert, hat jeder Benutzer beim Öffnen oder Anmelden an die Datenbank die für den Standardbenutzer definierten Rechte und Einschränkungen. Der Standardbenutzer muss keinen Namen eingeben. Wurde ihm kein Kennwort zugewiesen, erscheint der Kennwortdialog nicht mehr. Die Datenbank öffnet sich direkt. Diese Option vereinfacht den Zugriff auf die Datenbank beim Verwalten eines komplexen Datenkontrollsystems.
 - Wurde dem Standardbenutzer ein Kennwort zugeordnet, erscheint beim Öffnen der Datenbank ein Dialogfenster, wo er dieses Kennwort eingeben muss:



- Bei Standardbenutzer ohne Kennwort erscheint dieses Dialogfenster nicht.
- **Hinweis:** Sie können die Anzeige des Kennwortdialogs "erzwingen", wenn der Modus "Standardbenutzer" aktiv ist, z.B. um sich als Administrator oder Designer anzumelden. Dazu öffnen oder melden Sie sich mit gedrückter **Umschalttaste** an die Datenbank an.
- **Benutzerliste in Kennwortdialog anzeigen:** Ist diese Option markiert, müssen Benutzer ihre Namen in der Benutzerliste wählen und ihr Kennwort im Kennwortdialog eingeben. Ist diese Option inaktiv, muss der Benutzer selbst Name und Kennwort eingeben. Weitere Informationen dazu finden Sie im Abschnitt **Überblick über das Zugriffssystem**.
 - **Benutzerliste alphabetisch anzeigen** (nur aktiv, wenn die vorige Option markiert ist): Mit dieser Option erscheint die Benutzerliste im Dialogfenster zur Eingabe der Kennwörter in alphabetischer Reihenfolge.

- **Die Anwender können ihre Kennwörter ändern:** Ist diese Option markiert, erscheint im Kennwortdialog die Schaltfläche **Ändern**. Klickt der Benutzer darauf, öffnet sich ein Dialogfenster, wo er sein Kennwort ändern kann. Weitere Informationen dazu finden Sie im Abschnitt **Kennwortsystem pflegen**. Sie können die Schaltfläche **Ändern** auch ausblenden, so dass Benutzer ihre Kennwörter nicht ändern können. Dazu deaktivieren Sie einfach diese Option.

Optionen

- **Benutzer-Einstellungen in externer Datei:** Damit können Sie Benutzereinstellungen extern darstellen. Ist diese Option aktiviert, werden die Datenbank-Eigenschaften in drei Dialogfenster aufgeteilt: **Struktureinstellungen**, **Benutzereinstellungen** und **Benutzereinstellungen für Datendatei**. Weitere Informationen dazu finden Sie im Abschnitt **Benutzereinstellungen verwenden**.
- **Führe "On Host Database Event" Methode der Komponente aus:** Die **Datenbankmethode On Host Database Event** vereinfacht die Initialisierung und Backup Phasen für 4D Komponenten. Aus Sicherheitsgründen müssen Sie die Ausführung dieser Methode in jeder Host Datenbank explizit autorisieren. Dazu müssen Sie diese Option markieren, sie ist standardmäßig nicht markiert.
Ist diese Option markiert, gilt folgendes:
 - 4D Komponenten werden geladen
 - Jede **Datenbankmethode On Host Database Event** der Komponente (sofern vorhanden) wird von der Host Datenbank aufgerufen
 - Der Code der Methode wird ausgeführtIst diese Option nicht markiert, gilt folgendes:
 - 4D Komponenten werden geladen, müssen aber die Initialisierung und Backup Phasen selbst verwalten
 - Der Entwickler der Komponente muss die Komponentenmethoden, die die Host Datenbank während dieser Phasen aufrufen muss, veröffentlichen (Startup und Shutdown)
 - Der Entwickler der Host Datenbank muss die passenden Methoden der Komponente zur rechten Zeit aufrufen (muss in der Dokumentation der Komponente abgedeckt sein)

Die Seite Kompatibilität bietet Parameter zur Wahrung der Kompatibilität mit älteren 4D Versionen. Beachten Sie, dass eine Reihe der Optionen abhängig von der 4D Version angezeigt werden, mit der die Originaldatenbank erstellt wurde: 2004.x, v11, v12 usw. sowie die in dieser Datenbank geänderten Einstellungen.

Hinweis: Diese Seite erscheint nicht in Anwendungen, die mit der aktuellen 4D Version erstellt wurden (nicht-konvertierte Anwendungen).

- **Felder in Dialogfenstern eingebbar:** In früheren Versionen von 4D war es nicht möglich, Werte über Felder in Dialogfenstern einzugeben. Sie werden z.B. über den 4D Befehl **DIALOG** angezeigt. Diese Einschränkung ist seit 4D 2004 aufgehoben. Sie können das vorige Verhalten beibehalten, insbesondere wenn Ihre Datenbank in Dialogfenstern Felder zum Anzeigen von Daten verwendet. Diese Option ist für konvertierte Datenbanken, die mit einer Version vor 2004 erstellt wurden, standardmäßig aktiv, für Datenbanken, die in Version 2004 oder neuer erstellt wurden, inaktiv.
- **Optionsfelder nach Namen gruppieren:** In früheren Versionen von 4D war koordiniertes Verhalten einer Gruppe von Optionsfeldern möglich, wenn die zugewiesenen Variablen denselben Anfangsbuchstaben hatten, z.B. m_button1, m_button2, m_button3, etc.. (Koordiniert = Es lässt sich immer nur ein Optionsfeld in der Gruppe auswählen). Das wurde ab 4D 2004 wie folgt geändert: Ein Satz Optionsfelder arbeitet koordiniert, wenn er im Formulareditor gruppiert ist. Weitere Informationen dazu finden Sie im Abschnitt **Optionsfelder und Optionsbilder**. Dieser Modus gilt für Optionsfelder, 3D Optionsfelder und Optionsfelder vom Typ Bild. Aus Kompatibilitätsgründen wird in konvertierten Datenbanken standardmäßig der alte Modus beibehalten. Wollen Sie den neuen Modus verwenden, deaktivieren Sie diese Option. Datenbanken, die in Version 2004 oder neuer erstellt wurden, verwenden die neue Arbeitsweise.
- **Formular für jeden Datensatz während PRINT SELECTION neu laden:** In früheren Versionen von 4D wurde das Formular, welches über den 4D Befehl **PRINT SELECTION** beim Drucken verwendet wird, für jeden Datensatz erneut geladen. So ließen sich alle Objekteinstellungen, die der Entwickler unter Umständen per Programmierung verändert hatte, über das Formularereignis [On printing detail](#) erneut initialisieren. Diese Funktionsweise wurde ab 4D 2004 zur Optimierung der Performance aufgehoben. Der 4D Entwickler muss jetzt selbst die gewünschten Einstellungen in der Formularmethode reinitialisieren – das ist identisch mit der Arbeitsweise von Listenformularen mit dem Formularereignis [On display detail](#). Über diese Option können Sie die bisherige Arbeitsweise beibehalten. Datenbanken, die in Version 2004 oder neuer erstellt wurden, verwenden die neue Arbeitsweise.
- **Verwende nicht neue Kontext-Referenzmethode:** Ist diese Option nicht markiert (Standardeinstellung), setzt der 4D Web Server die Kontextnummer in der URL der gesendeten HTML Dokumente. In älteren Versionen (Option markiert) sendete der 4D Web Server die Kontextnummer für jeden Eintrag einer Seite an den Browser, was die Bearbeitung verlangsamte. Die Option muss aus Kompatibilitätsgründen unter Umständen aktiv sein. Beachten Sie, dass eine Änderung dieser Einstellung erst nach Neustart der Datenbank in Kraft tritt.
- **"/" von unbekanntem URLs entfernen:** In früheren 4D Versionen wurden unbekannte URL, d.h. URL, die weder zu einer vorhandenen Seite, noch zu einer 4D spezifischen URL gehören, in den Datenbankmethoden *On Web Authentication* und *On Web Connection* (\$) zurückgegeben und begannen nicht mit dem Zeichen "/". Diese Vorgehensweise wurde in 4D 2004 entfernt. Sollten Sie Algorithmen eingebaut haben, die auf diesem Sonderfall basieren und wollen Sie das bisherige Verhalten beibehalten, deaktivieren Sie diese Option.
- **Drag & Drop von außerhalb von 4D verweigern:** 4D erlaubt ab Version 11 im Anwendungsmodus Drag&Drop für Auswahlen, Objekte und externe Dateien, z.B. Bilddateien. Diese Möglichkeit muss der Code der Datenbank jedoch unterstützen. In konvertierten Datenbanken aus früheren Versionen können Störungen auftreten, wenn der vorhandene Code nicht entsprechend angepasst wird. Um das zu verhindern, markieren Sie diese Option. Dann lassen sich externe Objekte nicht per Drag&Drop in 4D Formulare legen. Beachten Sie, dass sich externe Objekte mit dem Attribut **Automatisches Drop** weiterhin in Objekte ziehen lassen, sofern die Anwendung die Daten interpretieren kann (Text oder Bild). Weitere Informationen dazu finden Sie im Abschnitt **Drag and Drop**.
- **QUERY BY FORMULA auf Server ausführen und Execute ORDER BY FORMULA auf Server ausführen:** Ab 4D Version 11 werden die Such- und Sortierbefehle „nach Formel“ auf dem Server ausgeführt und nur das Ergebnis an den Client-Rechner zurückgegeben. Das gilt für die 4D Befehle **QUERY BY FORMULA**, **QUERY SELECTION BY FORMULA** und **ORDER BY FORMULA**. Werden Variablen direkt in der Formel aufgerufen, wird die Suche mit dem Wert der Variablen auf dem Client-Rechner durchgeführt. So wird zum Beispiel die Anweisung

```
<strong>QUERY BY FORMULA</strong>([aTable];[aTable]aField=theValue)
```

Dieses Prinzip gilt dagegen nicht für Formeln mit Methoden, die ihrerseits Variablen aufrufen: In diesem Fall wird der Wert der Variablen auf dem Server berechnet.

In konvertierten Datenbank kann diese neue Funktionsweise vorhandene Algorithmen beeinträchtigen. Folglich werden in diesem Kontext die Befehle weiterhin auf dem Client-Rechner ausgeführt. Um die Vorteile des neuen Algorithmus in konvertierten Datenbanken zu nutzen, können Sie einfach diese Optionen markieren.











Hinweis: Sie können diese Option auch über den 4D Befehl **SET DATABASE PARAMETER** setzen.

- Die Formel lässt sich nicht in die einzelnen Elemente in Form von {Feld ;Vergleichsoperator ;Wert} aufbrechen
- Es werden zwei Felder derselben Tabelle miteinander verglichen.
- o Sie können diese Option auch über den 4D Befehl **SET DATABASE PARAMETER** setzen.
- **Verschachtelte Transaktionen erlauben:** Diese Option unterstützt mehrstufige Transaktionen. 4D akzeptiert ab Version 11 verschachtelte Transaktionen auf einer unbegrenzten Anzahl von Ebenen. Da diese Option jedoch die Funktionsweise von Datenbanken älterer Versionen beeinträchtigen kann, ist sie in konvertierten Datenbanken standardmäßig inaktiv, d.h. hier sind Transaktionen auf eine Ebene begrenzt. Um in einer konvertierten Datenbank Transaktionen auf mehreren Ebenen durchzuführen, müssen Sie diese Option markieren. Diese Option ist standardmäßig nicht markiert. Sie gilt spezifisch für eine Datenbank.
Hinweis: Diese Option hat keine Auswirkung auf Transaktionen, die mit der SQL-Engine von 4D ausgeführt werden. SQL-Transaktionen sind immer verschachtelt.
- **Unicode Modus:** Aktiviert bzw. deaktiviert den Unicode Modus für die aktuelle Datenbank. Im Unicode Modus verwalten Datenbank-Engine und Programmiersprache Zeichenketten in Unicode nativ. Im Kompatibilitätsmodus wird von der Programmiersprache der ASCII Zeichensatz verwendet. Mit dieser Option lässt sich die Kompatibilität konvertierter Datenbanken aufrechterhalten. Sie ist für Datenbanken, die mit 4D v11 oder höher erstellt wurden, standardmäßig markiert, für konvertierte Datenbanken nicht markiert.
Hinweise:
 - o Ändern Sie diese Option, wird sie erst nach Neustart der Anwendung berücksichtigt.
 - o Sie gilt immer für eine Datenbank. Im interpretierten Modus können Sie demnach eine Unicode Datenbank zusammen mit Komponenten im Ascii Kompatibilitätsmodus haben, oder umgekehrt.
 - o Sie können den Modus Unicode auch mit dem 4D Befehl **SET DATABASE PARAMETER** festlegen.
 Weitere Informationen zur Unterstützung von Unicode finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **EXPORT TEXT**.
- **Verwende Punkt und Komma als Platzhalter in Zahlenformaten:** Ab 4D Version 11 basieren Zahlenformate automatisch auf den landesspezifischen Parametern des Systems (siehe "Zahlenformate" unter **Anzeigeformate**). 4D ersetzt die Zeichen "." und "," in Zahlenformaten automatisch mit den Trennzeichen für Tausend und Dezimal, wie es im Betriebssystem definiert wurde. Punkt und Komma dienen als Platzhalter in den Formaten 0 oder #. In bisherigen 4D Versionen haben Anzeigeformate für Zahlen nicht die landesspezifischen Parameter des Betriebssystems berücksichtigt. So ist das Format "####,##0.00" ein für das amerikanische Betriebssystem gültiges Format. Das führt in einem deutschen oder schweizer System zu einem inkorrekten Ergebnis. In konvertierten Datenbanken ist diese Funktionsweise zur Wahrung der Kompatibilität nicht aktiviert. Um sie zu nutzen, müssen Sie diese Option markieren.
- **Automatische Zuweisung von Variablen:** In bisherigen 4D Versionen hat der Web Server automatisch den Wert von Variablen kopiert, die über ein Web Formular oder eine URL in 4D Prozessvariablen gesendet wurden. Im interpretierten Modus wurde der Wert jeder empfangenen Variable, direkt in eine 4D Prozessvariable mit demselben Namen kopiert; im kompilierten Modus mussten die Variablen zuvor in einer Projektmethode COMPILER_WEB deklariert werden. Dieser Mechanismus ist ab 4D v13.4 veraltet und in neuen Anwendungen nicht mehr verfügbar. Zur Wahrung der Kompatibilität wird er in konvertierten Anwendungen noch beibehalten. Sie können ihn aber auch deaktivieren. Wir empfehlen jetzt, die dafür vorgesehenen Befehle **WEB GET VARIABLES** oder **WEB GET BODY PART** zu verwenden.
- **Verwende legacy Netzwerk-Schicht:** 4D Applikationen enthalten ab Version 14 R5 eine neue Netzwerk-Schicht mit Namen *ServerNet*, um die Kommunikation zwischen 4D Server und Rechnern mit remote 4D (Clients) zu steuern. Die bisherige Netzwerk-Schicht ist überholt, wird jedoch zur Wahrung der Kompatibilität mit vorhandenen Datenbank beibehalten. Mit dieser Option können Sie die bisherige Netzwerkschicht je nach Ihren Bedürfnissen in Ihren konvertierten 4D Server Anwendungen aktivieren oder deaktivieren, z.B. beim Migrieren Ihrer Client Applikationen (siehe Abschnitt **Netzwerk und Client-Server Optionen**). Diese Option ist in konvertierten Datenbanken standardmäßig aktiviert und für neu angelegte Datenbanken automatisch deaktiviert. Beachten Sie, dass die Änderung erst nach dem Neustart der Anwendung berücksichtigt wird. Auch Client-Anwendungen, die sich mit der neuen Netzwerk-Schicht anmelden können, müssen neu gestartet werden. Zum Verwenden von *ServerNet* ist mindestens die Client-Version von 4D v14 R4 erforderlich, siehe Abschnitt **Netzwerk und Client-Server Optionen**.
Hinweis: Diese Option lässt sich mit dem Befehl **SET DATABASE PARAMETER** auch per Programmierung steuern.
- **Sichere Methode in Unicode:** Damit können Sie in 4D Methoden Code-Strings in Unicode sichern. In 4D Versionen vor v15 wurden Code Strings (Formeln, Variablen- und Methodennamen, Kommentare, etc.) von 4D Methoden über die aktuelle lokale Codierung gespeichert. Das konnte zu Problemen führen, insbesondere wenn Entwickler aus verschiedenen Ländern den 4D Code gemeinsam nutzten: Wird z.B. 4D Code mit französischen Akzenten oder deutschen Umlauten an einen englischen Entwickler gesendet, können diese Sonderzeichen verloren gehen. Größere Probleme treten bei Code auf, der mit einer japanischen Version geschrieben wird. Solche Probleme entfallen bei Methoden, die in Unicode gesichert werden, und ermöglicht den korrekten Austausch von 4D Code mit spezifischen landessprachlichen Zeichen. Wir empfehlen, die Option Unicode Modus für Methoden in Ihren Anwendungen sobald wie möglich zu aktivieren, insbesondere wenn Sie auf internationaler Ebene arbeiten.
Hinweise:
 - o Diese Funktionalität gilt für die Sprache selbst und ihre Interpretation. Einige Fenster des 4D Editors, wie die Eigenschaftenliste, verwenden noch die aktuelle lokale Codierung und zeigen so u.U. manche Strings nicht korrekt an. Das behindert jedoch nicht die Ausführung von Code.
 - o Sie können die Option jederzeit aktivieren/deaktivieren. Die Änderung gilt nur für anschließend gesicherte Methoden.
- **Verwende Objektnotation, um auf Objekteigenschaften zuzugreifen (Unicode nötig):** Damit können Sie die Zeichen ".", "[" und "]" in Ihrem Code als Symbole der Objektnotation zum Definieren von Token Teilen verwenden -- und nicht in Tabellen-, Feld-, Methoden- oder Variablennamen. Wollen Sie Objektnotation verwenden, müssen Sie diese Option für Anwendungen aktivieren, die vor 4D v17 erstellt wurden. Denn in allen früheren 4D Releases waren die Symbole "." "]" in Namen erlaubt. Aktivieren Sie diese Option für Ihre konvertierte Anwendung, machen Sie ihren Code mit Objektnotation kompatibel. Wir empfehlen, die Kompatibilität Ihres Code über das MSC zu prüfen (siehe **Seite Prüfen**). Weitere Informationen dazu finden Sie im Abschnitt **Objektnotation verwenden**.
Hinweis: Für Objektnotation muss die Anwendung im Unicode-Modus arbeiten, d.h. die Optionen **Unicode Modus** und **Sichere Methode in Unicode** müssen markiert sein.
- **Verwende Datumstyp statt ISO Datumsformat in Objekten:** Damit können Sie Datumsangaben in Objektattributen als Datumstyp anstatt als Text im ISO-Format speichern. In bisherigen 4D Versionen ließ sich ein Datum in

Objektattributen nur als String abspeichern. Ab 4D v16 R6 können Sie diese neue Option markieren, um das Datum in Objektattributen als Datumstyp zu speichern. Dies gilt nur für ein neu eingegebenes Datum, zuvor im ISO-Format gespeicherte Datumsangaben bleiben erhalten. Dieses Feature können Sie auch per Programmierung über den Befehl **SET DATABASE PARAMETER** und die Selektoren Date_type, String_type with time zone und String_type without time zone steuern.

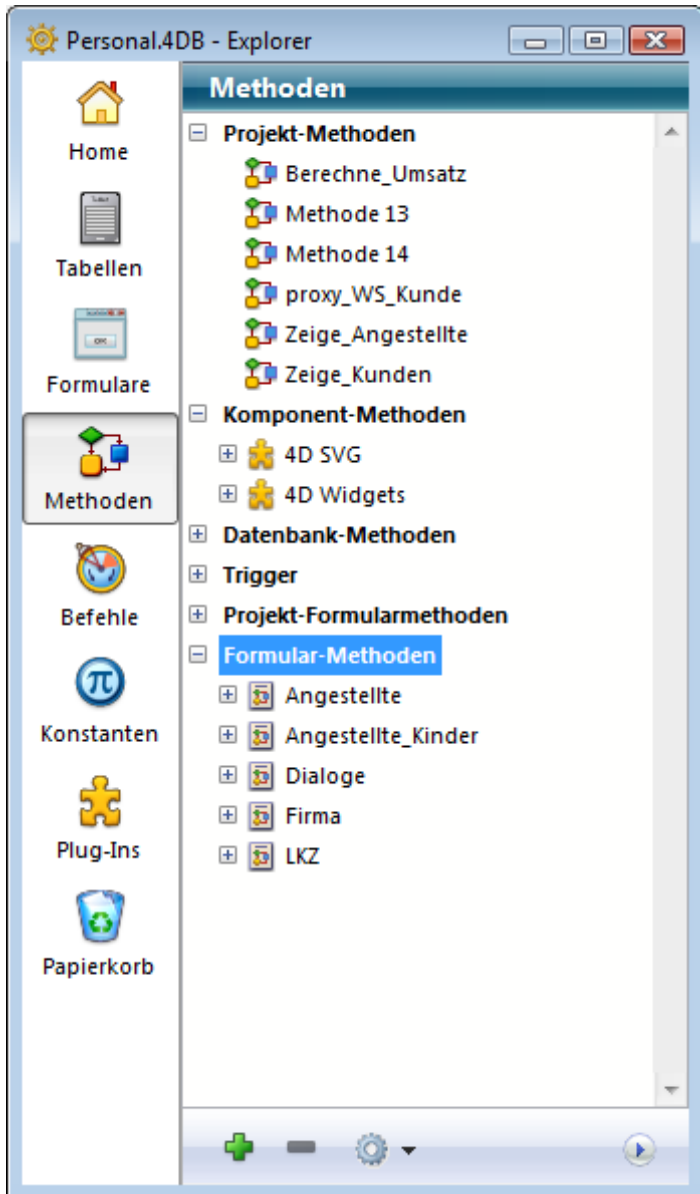
- **Benutze neue Architektur für Anwendungen-Weitergabe:** Diese Option ist für Anwendungen ab 4D v15 R4 standardmäßig markiert und in konvertierten Anwendungen standardmäßig deaktiviert. Wollen Sie die neuen Mechanismen nutzen, müssen sie diese explizit markieren. Sie muss auf dem Rechner gesetzt werden, der die Endanwendung generiert. Weitere Informationen dazu finden Sie in den Abschnitten **Zuletzt geöffnete Datendatei** und **Verbindungsprozess für eingebundene Clients**.

Explorer

-  Überblick und Funktionsweise
-  Seite Home
-  Seite Tabellen
-  Seite Formulare
-  Seite Methoden
-  Seite Befehle
-  Seite Konstanten
-  Seite Plug-Ins
-  Seite Papierkorb
-  Kommentare verwenden

Überblick und Funktionsweise

Der Explorer ist ein Fenster in der Designumgebung, über das Sie rasch auf die verschiedenen Strukturobjekte zugreifen können. Es erscheint, wenn Sie im Menü **Design** den Befehl **Explorer** wählen oder in der Werkzeugleiste auf das Icon **Explorer** klicken.



Über die Schaltflächen am linken Rand gelangen Sie auf die einzelnen Seiten im Designmodus, das sind Tabellen, Formulare, Methoden, Befehle und Konstanten der 4D Programmiersprache und Plug-Ins.

Die Seiten Home und Papierkorb bieten spezifische Funktionen.

Öffnen Sie eine Seite, erscheinen die dazugehörigen Objekte in Form einer hierarchischen Liste. Der Bereich unter der Liste enthält Schaltflächen zum Hinzufügen und Entfernen sowie ein DropDown-Menü mit Optionen.

Objekt umbenennen

Wenn Sie einen Ordner, eine Tabelle, ein Feld, Formular oder eine Methode umbenennen wollen, halten Sie unter Windows die **Alt-Taste**, auf Macintosh die **Wahltaste** gedrückt und klicken auf den Namen des Objekts. Sie können auch zweimal auf das Objekt klicken, damit der Text eingebbar wird. Warten Sie einen Moment zwischen den beiden Klicks, damit Sie keinen Doppelklick auslösen!

Der Text lässt sich jetzt bearbeiten.




Führen Sie die Änderungen durch und klicken dann außerhalb des Textbereiches, um die Änderung zu speichern.

Hinweis: Ändern Sie den Namen eines Formulars, werden Methoden und Formulare, die es enthalten, ungültig. Sie müssen jedes Objekt aktualisieren, damit sie wieder korrekt funktionieren.

Der Explorer zeigt die Objekte in alphabetischer Reihenfolge an. Wenn sich die Reihenfolge durch den neuen Namen ändert, sortiert 4D die Liste neu, sobald Sie außerhalb des Eingabebereiches klicken.

4D Server: Der Objektname wird auf dem Server verändert, wenn Sie außerhalb des Eingabebereichs für den Namen klicken. Ändern mehrere Benutzer gleichzeitig den Namen, wird der zuletzt geänderte Namen verwendet. Sie können auch einen Objekteigentümer bestimmen, so dass nur bestimmte Benutzer den Namen verändern können. Weitere Informationen über Zugriffsrechte finden Sie im Kapitel **Benutzer und Gruppen**.

Vorschau ein- oder ausblenden

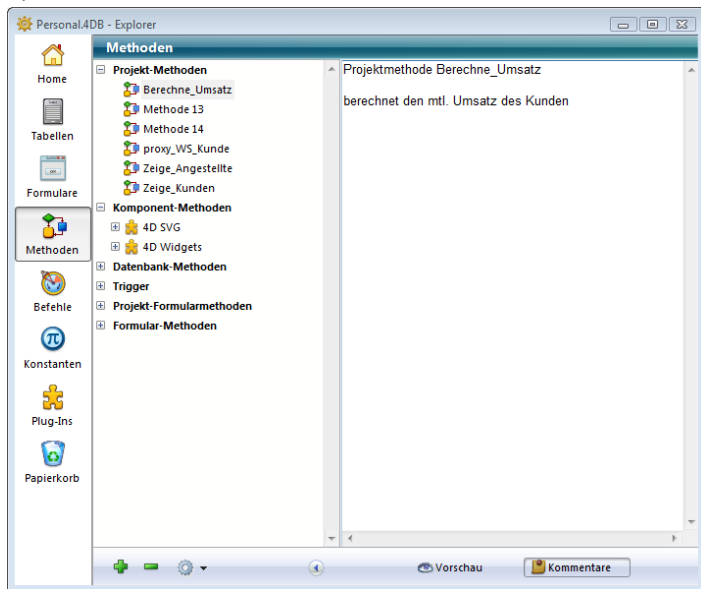
Sie können den Vorschaubereich des Explorers ein- oder ausblenden. Klicken Sie auf das Icon  am unteren Rand, erscheint im rechten Bereich eine Vorschau der Tabellendarstellung, Formulare, Methoden, Dokumentation zu Befehlen und Konstanten sowie Informationen zu den installierten Komponenten und Plug-Ins. In der Vorschau können Sie auch zu bestimmten Strukturobjekten Kommentare eingeben und ansehen.

Um die Vorschau wieder auszublenden, klicken Sie erneut auf dieses Icon.

Kommentare anzeigen

Mit 4D können Sie folgenden Strukturobjekten Kommentare zuordnen: Ordner, Tabellen, Datenfelder, Formulare, Methoden (Datenbank-Projekt-, Formularmethoden und Trigger). Diese Kommentare lassen sich über den Explorer anzeigen bzw. verändern. Dazu muss der Vorschaubereich eingeblendet sein.

Um Kommentare zu einem Objekt anzuzeigen, wählen Sie das Objekt im linken Bereich des Explorers aus und klicken auf das Optionsfeld *Kommentar* unter der Vorschau. Der Kommentar erscheint in der Vorschau.



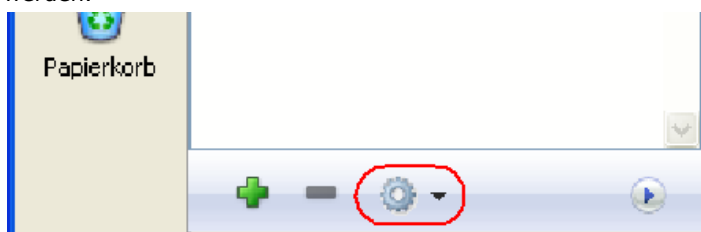
Weitere Informationen dazu finden Sie im Abschnitt **Kommentare verwenden**.

Drag and Drop verwenden

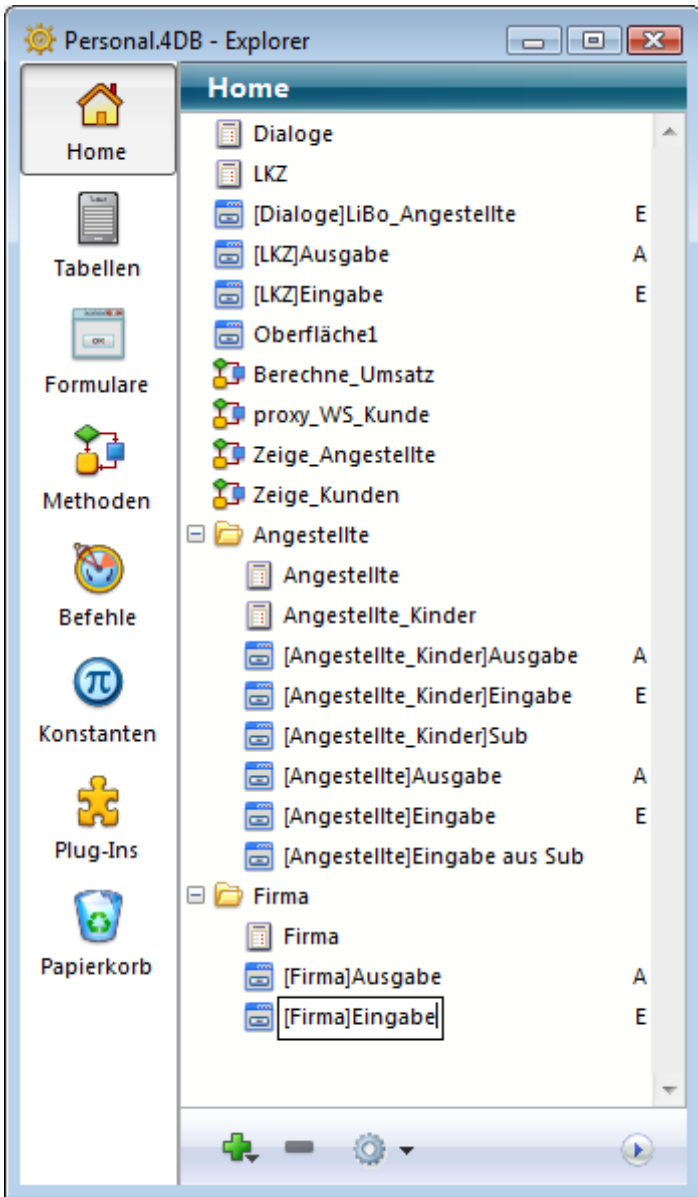
Drag&Drop setzen Sie in vielen Fällen ein, um im Editorfenster ein Strukturobjekt hinzuzufügen. Sie können z.B. einem Formular ein Datenfeld anfügen, indem Sie bei gedrückter Maustaste einen Feldnamen von der Tabellenseite des Explorers in ein Formular im Formulareditor ziehen. Wenn Sie mit dem Methodeneditor arbeiten, können Sie per Drag&Drop einer Methode die Namen von Tabellen, Formularen, Datenfeldern, Projektmethoden, Konstanten und Befehlen anfügen. Die folgenden Abschnitte dieses Kapitels beschreiben die einzelnen Seite des Explorers und geben spezifische Informationen über die Drag&Drop Möglichkeiten für die jeweilige Seite.

Kontextmenüs und Menü Optionen

Auf den Seiten Home, Formulare, Methoden und Papierkorb können Sie Kontextmenüs und das DropDown-Menü **Optionen** unter der Liste verwenden. Sie bieten zusätzliche Funktionen, die in den nächsten Abschnitten dieses Kapitels beschrieben werden.



Auf der Seite Home definieren und verwenden Sie die Ordner von Objekten in 4D Explorer.



Ähnlich wie die Ordner in Windows und Mac OS Betriebssystemen enthalten 4D Ordner verschiedene Objekte, die nach eigenen Kriterien gruppiert werden, in der Regel nach Funktionalität und nicht nur nach Typ. Sie können z.B. im Ordner Meldungen alle Objekte gruppieren, die zum Meldungssystem in Ihrer Anwendung gehören.

Sie können Objekte vom Typ Tabelle, Formular und Projektmethode in Ordner legen. Auch Unterordner auf mehreren Ebenen sind möglich.

Alle selbst erstellten Objekte der Struktur, also Tabellen, Formulare und Projektmethoden erscheinen immer auf der Seite Home. Sie werden auch auf den anderen Seiten des Explorer aufgelistet.

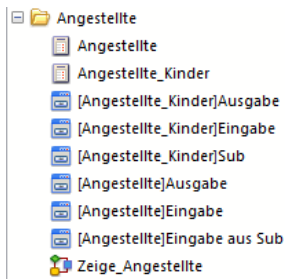
Auf dieser Seite können Sie Ordner oder Unterordner hinzufügen, umbenennen, bewegen und Ordner bzw. Objekte neu organisieren. Sie können auch direkt Tabellen, Formulare und Projektmethoden erstellen. Wie auf anderen Seiten des Explorer können Sie eine Tabelle, ein Formular oder eine Projektmethode per Doppelklick auf den Namen im Fenster des entsprechenden Editors öffnen oder per Drag-and-Drop einfügen. Ebenso können Sie Ordner und Objekte umbenennen oder löschen und jedes Objekt schon beim Erstellen einem Ordner zuweisen.

Auf dieser Seite können Sie im Designmodus auch verschiedene Objekttypen, wie Methoden, Formulare, etc., in Editoren der Datenbank oder in andere Datenbanken ziehen. Die je nach Objekttyp möglichen Drag-and-Drop Optionen werden in den nachfolgenden Abschnitten beschrieben. Weitere Informationen zu Drag-and-Drop zwischen Datenbanken finden Sie im Abschnitt **Objekte bewegen**.

4D Server: Alle Client-Rechner, die mit derselben Anwendung verbunden sind, arbeiten mit derselben Ordnerkonfiguration. Führt ein Client eine Änderung auf der Seite **Home** durch, gilt sie automatisch und sofort für alle angeschlossenen Client-Rechner.

Ordner oder Unterordner erstellen

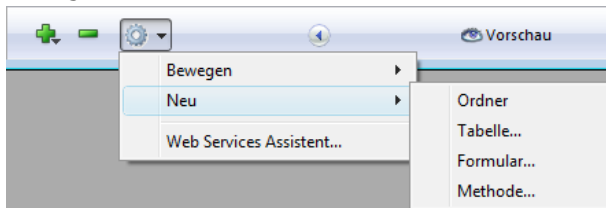
Ordner sind nützlich, um Datenbanken nach Funktionalität oder eigenen Kriterien wie Datum oder Sprache zu organisieren. Sie vereinfachen auch den Zugriff auf die Objekte über den Explorer. Ordner sind auch in Form hierarchischer Listen im Methodeneditor von 4D verfügbar. So lassen sich Objekte beim Schreiben von Methoden einfacher aufrufen:



Über Ordner können Sie auch die Anzeige der Tabellen im Struktureditor filtern. Weitere Informationen dazu finden Sie im Abschnitt **Tabellen nach Ordner ein-/Ausblenden**.

Standardmäßig liegen alle Objekte auf der obersten Ebene. Sie können jederzeit auf der Seite Home des Explorer Ordner bzw. Unterordner hinzufügen.

Dazu wählen Sie im DropDown-Menü Optionen unter der Liste bzw. im Kontextmenü des Bereichs (rechte Maustaste) den Eintrag **Neu>Ordner**:



Sie können auch am unteren Rand auf das Pluszeichen klicken und dann im DropDown-Menü den Eintrag **Neuer Ordner** wählen.

Der Ordner oder Unterordner wird sofort angelegt, als Standardname wird "Ordner_+Ordernummer" vergeben. Sie können den Namen beliebig ändern, er muss jedoch einmalig sein (unabhängig von der Hierarchie-Ebene) und darf max. 31 Zeichen enthalten.

- Um einen Ordner zu anzulegen, stellen Sie sicher, dass beim Hinzufügen kein Element ausgewählt ist,
- Um einen Unterordner zu erstellen, klicken Sie vor dem Hinzufügen auf den Ordner, in dem Sie den Unterordner hinzufügen möchten.

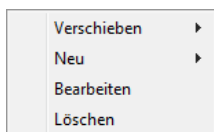
Sie können dann per Drag&Drop oder über Befehle zum Bewegen ganz einfach einen Ordner in einen Unterordner und umgekehrt umwandeln.

Über den Eintrag **Neu>** im Kontextmenü der Seite Home können Sie nicht nur Ordner, sondern auch Tabellen, Formulare und Projektmethoden erstellen. Wählen Sie einen der Untereinträge aus, erscheint direkt das entsprechende Fenster zum Erstellen des Objekts.

Das Kontextmenü der Seite Home und das Menü Optionen am unteren Rand bieten je nach ausgewähltem Objekttyp weitere Funktionalitäten. Die jeweils verfügbaren Optionen werden in den entsprechenden Abschnitten dieses Kapitels beschrieben.

Objekt löschen

Über den Eintrag **Löschen** im Kontextmenü löschen Sie die Auswahl. Enthält sie einen Ordner bzw. Unterordner, werden alle darin enthaltenen Elemente gelöscht.



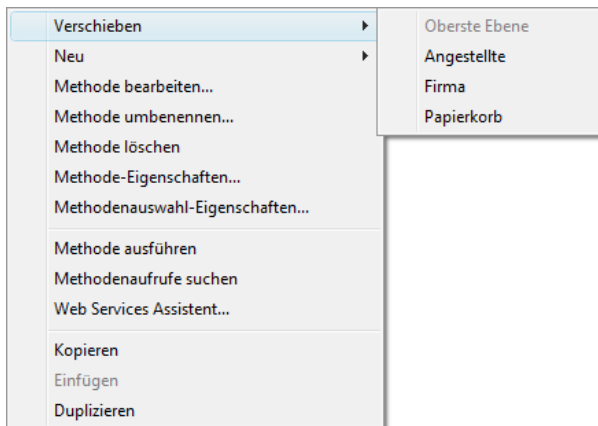
Sie können jede Art von Element löschen, also Ordner, Unterordner, Tabellen, Formulare und Projektmethoden sowie mehrfache Auswahlen. Gelöschte Objekte verschwinden aus Listen und Editoren der Anwendung.

Gelöschte Elemente werden in den Papierkorb des Explorer gelegt. Weitere Informationen dazu finden Sie auf der **Seite Papierkorb**.

Sie können auch ein oder mehrere Objekte auf der Seite Home des Explorer auswählen und dann im Kontextmenü (rechte Maustaste) bzw. dem Menü **Optionen** unter der Liste den Eintrag **Verschieben>Papierkorb** wählen.

Verschieben

Mit dem Eintrag **Verschieben** des Kontextmenüs auf der Seite Home können Sie den Inhalt eines Ordners ändern, ohne vorher den Zielordner aufzuklappen. Das Kontextmenü erscheint, wenn Sie mit der rechten Maustaste auf das Objekt, den Ordner oder die Auswahl klicken. Das DropDown-Menü zeigt alle vorhandenen Ordner und Unterordner als mögliche Ziele an:

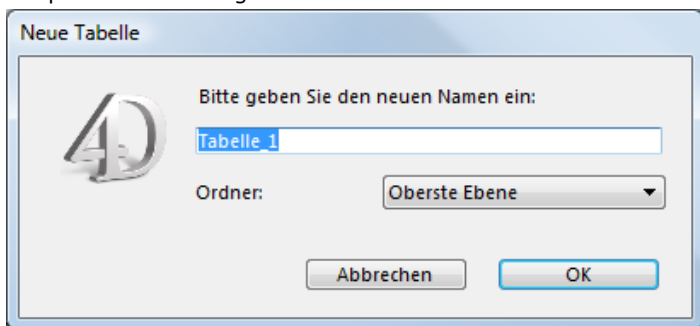


Wählen Sie den Ordner aus, in welchen Sie die Auswahl legen möchten. Wählen Sie den Eintrag **Oberste Ebene**, wird die Auswahl nicht in einen Ordner, sondern auf die oberste Ebene gelegt. Dieser Eintrag erscheint in Grauschrift, wenn Sie versuchen, einen Ordner in die erste Ebene zu legen, der dort bereits vorhanden ist. Wählen Sie **Papierkorb**, wird die Auswahl gelöscht (siehe voriger Abschnitt).

Objekt beim Erstellen in Ordner legen

Sie können ein Objekt schon beim Erstellen in einen Ordner legen. Das jeweilige Dialogfenster zum Erstellen von Tabellen, Formularen bzw. Projektmethoden ermöglicht, im DropDown-Menü Ordner den gewünschten Ordner auszuwählen. Objekte werden standardmäßig auf die oberste Ebene oder in den zuletzt im Explorer gewählten Ordner gelegt.

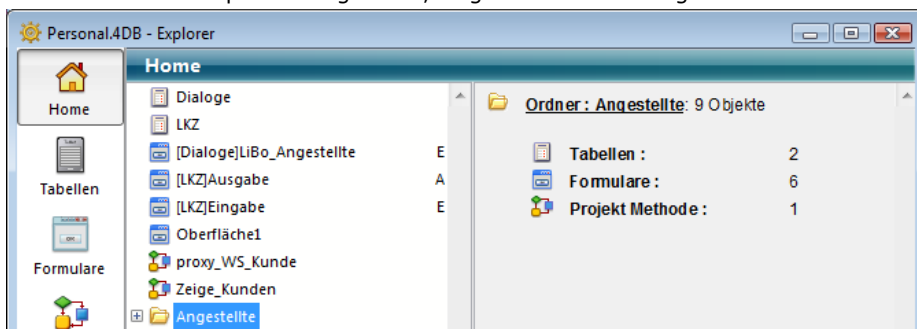
Beispiel für das Dialogfenster zum Erstellen einer Tabelle:



Sie können natürlich einmal angelegte Objekte auf der Seite Home im Explorer jederzeit in einen anderen Ordner legen.

Information zum Ordner

Ist ein Ordner im Explorer ausgewählt, zeigt die Vorschau folgende Informationen an:



- **Ordner: Ordnername:** Anzahl der Objekte innerhalb des Ordners
- **Tabelle(n):** Anzahl der Tabellen innerhalb des Ordners
- **Formular(e):** Anzahl der Formulare innerhalb des Ordners
- **Projektmethode(n):** Anzahl der Projektmethoden innerhalb des Ordners
- **Ordner:** Anzahl der Ordner (Unterdordner) innerhalb des Ordners

Ist ein Objekttyp nicht in einem Ordner vorhanden, erscheint er nicht in der Vorschau. Ist der Ordner leer, zeigt der Bereich nur **Ordner: Ordnername** an. Sind mehrere Ordner bzw. Objekte ausgewählt, gilt die angezeigte Information für das aktuelle Element, d.h. das zuletzt angeklickte Element.

Seite Tabellen


Die Seite Tabellen listet alle Tabellen und Datenfelder der Anwendung auf. Sie verwenden diese Seite alternativ zum Strukturfenster, um auf Tabellen- und Feldeigenschaften zuzugreifen. Wenn eine Tabelle geöffnet ist, werden die Datenfelder der Tabelle gezeigt.

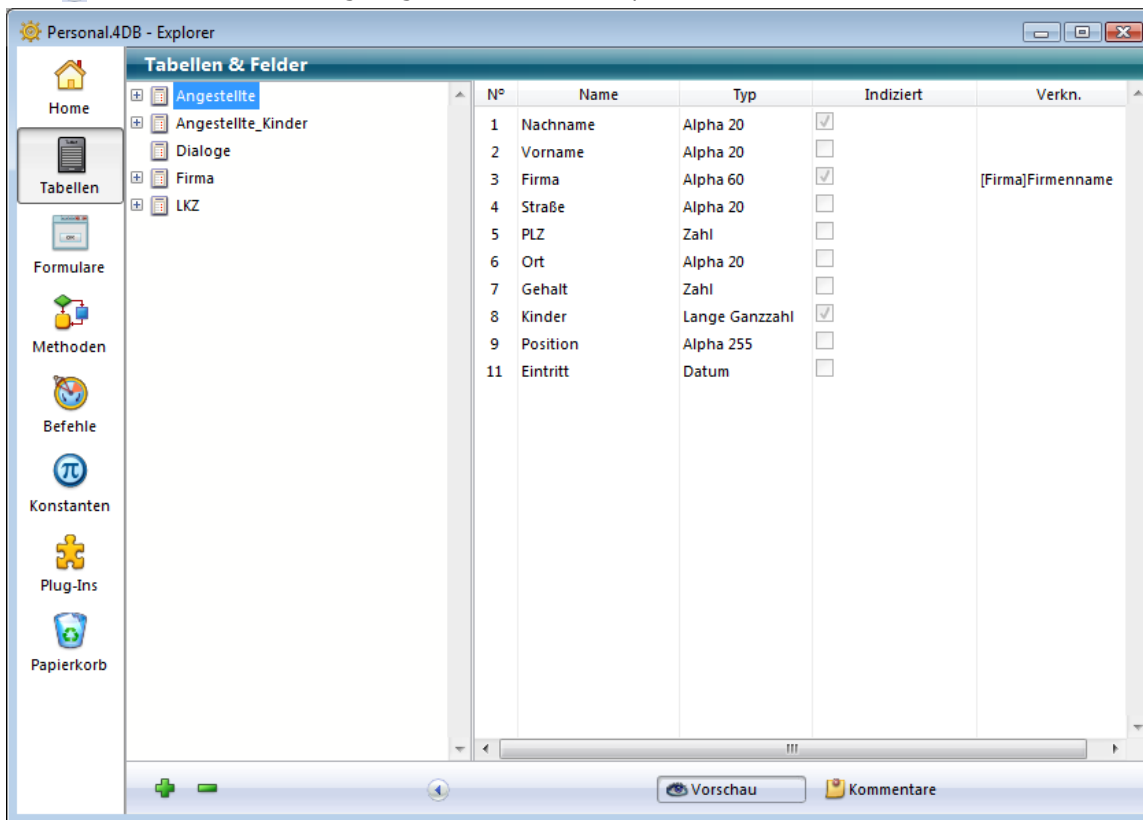
Der Feldtyp wird durch ein Icon links vom Namen angegeben. Um das Fenster mit den Feldeigenschaften zu öffnen, doppelklicken Sie auf einen Feldnamen. Weitere Informationen dazu finden Sie im Abschnitt [Get database parameter](#).

Drag and Drop verwenden

Sie fügen einem Formular ein Datenfeld hinzu, indem Sie bei gedrückter Maustaste den Feldnamen von der Tabellenseite in das Formular ziehen. Sie fügen einer Methode einen Tabellennamen oder einen Feldnamen hinzu, indem Sie bei gedrückter Maustaste den Namen in die Methode ziehen. Auf diese Weise erscheint der Name mit der richtigen Syntax. Wenn Sie z.B. das Datenfeld "Vorname" in eine Tabelle [Kunden] ziehen, erscheint es im Methodeneditor als "[Kunden] Vorname".

Vorschau

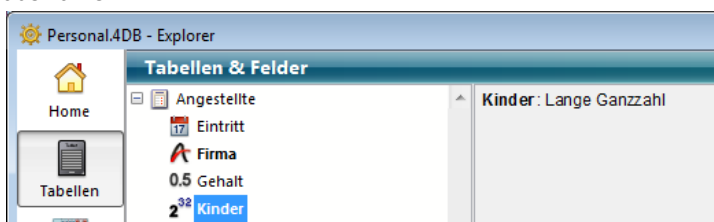
Sie können auch eine Vorschau der Tabelle erhalten. Hierzu wählen Sie eine Tabelle und klicken am unteren Rand auf das Icon , falls sie noch nicht angezeigt wird. Hier ein Beispiel für eine Vorschau der Tabelle:



N°	Name	Typ	Indiziert	Verkn.
1	Nachname	Alpha 20	<input checked="" type="checkbox"/>	
2	Vorname	Alpha 20	<input type="checkbox"/>	
3	Firma	Alpha 60	<input checked="" type="checkbox"/>	[Firma]Firmenname
4	Straße	Alpha 20	<input type="checkbox"/>	
5	PLZ	Zahl	<input type="checkbox"/>	
6	Ort	Alpha 20	<input type="checkbox"/>	
7	Gehalt	Zahl	<input type="checkbox"/>	
8	Kinder	Lange Ganzzahl	<input checked="" type="checkbox"/>	
9	Position	Alpha 255	<input type="checkbox"/>	
11	Eintritt	Datum	<input type="checkbox"/>	

Dieser Bereich ist nur zur Information; Sie können hier keine Werte verändern. Es zeigt Nummer, Name und Typ der Felder für die ausgewählte Tabelle, sowie zugewiesene Indizes und Verknüpfungen. Für jedes Feld, von dem eine Verknüpfung ausgeht (Viele-Feld) wird in der Spalte Verkn. das Zielfeld (Eine-Feld) angezeigt.

Sie können auch den Typ eines Datenfeldes anzeigen. Er erscheint in der Vorschau, wenn Sie den Namen in der Liste auswählen:



N°	Name	Typ	Indiziert	Verkn.
8	Kinder	Lange Ganzzahl	<input checked="" type="checkbox"/>	

Tabellenansicht im Strukturfenster ansehen

Sie erhalten eine Tabellenansicht, wenn Sie auf den Tabellennamen doppelklicken. 4D zentriert dann die Ansicht im Strukturfenster nach dieser Tabelle und zeigt seine Eigenschaften im Inspektorfenster an. Das ist besonders hilfreich bei einer Anwendung mit vielen Tabellen und erspart Ihnen das Scrollen im Strukturfenster zu einer bestimmten Tabellenansicht

Tabelle oder Feld hinzufügen


Über das Icon  am unteren Rand können Sie eine Tabelle oder ein Datenfeld hinzufügen.

Um eine Tabelle zu erstellen, wählen Sie einen Tabellennamen – stellen Sie dabei sicher, dass kein Objekt ausgewählt ist – und klicken auf das Pluszeichen. Auf dem Bildschirm erscheint der Standarddialog zum Hinzufügen von Tabellen. Weitere Informationen dazu finden Sie im Abschnitt **Tabellen erstellen und ändern**.

Um ein Datenfeld zu erstellen, wählen Sie ein vorhandenes Datenfeld und klicken auf das Pluszeichen. Das Inspektor-Fenster erscheint und bietet das Erstellen in der ausgewählten Tabelle an. Weitere Informationen dazu finden Sie im Abschnitt

Datenfeld anlegen.

Tabelle löschen

Über das Icon  am unteren Rand können Sie eine Tabelle löschen. Wählen Sie eine Tabelle und klicken auf das Minuszeichen, wird sie aus den 4D Editoren entfernt und ist nicht mehr verwendbar. Auch die zugeordneten Formulare und Methoden werden gelöscht.

Hinweis: Ein Datenfeld lässt sich nicht über das Minuszeichen löschen.

Tatsächlich wird die Tabelle nicht endgültig gelöscht, sondern in den Papierkorb gelegt und lediglich aus dem Programm ausgeblendet. Weitere Informationen dazu finden Sie im Abschnitt **Seite Papierkorb**. Die Tabelle wird nur endgültig gelöscht, wenn Sie den Papierkorb leeren.

Sie können eine Tabelle permanent aus dem Struktureditor löschen (siehe Abschnitt **Tabelle löschen** oder über die SQL Programmiersprache).

Die Seite Formulare enthält die beiden Listen **Projektformulare** und **Tabellenformulare**. Die Liste der Tabellenformulare zeigt die Tabellen der Anwendung und die dazugehörigen Formulare.

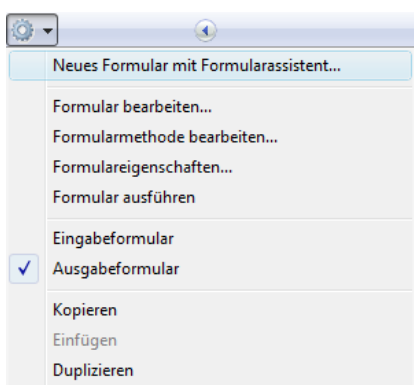
Projektformulare sind unabhängige Formulare, die keiner Tabelle zugeordnet sind. Sie sind besonders hilfreich beim Entwickeln von Oberflächen und Komponenten.

Tabellenformulare sind einer bestimmten Tabelle zugeordnet. Sie können automatische Funktionsweisen und weitere Funktionen zur Datenbearbeitung nutzen. Weitere Informationen dazu finden Sie im Abschnitt **Tabellenformulare und Projektformulare**.

Neues Formular erstellen

Es gibt zwei Möglichkeiten, ein neues Formular (Projekt- oder Tabellenformular) zu erstellen:

- **Über den Formularassistenten:** Dazu wählen Sie den Titel *Projektformulare* oder die Tabelle bzw. ein Formular in der Tabelle aus und dann im Menü **Optionen** am unteren Rand den Eintrag **Neues Formular mit Formularassistent:**



Es erscheint der Formularassistent, um ein neues Formular anzulegen. Weitere Informationen dazu finden Sie im Abschnitt **Formular über Formularassistent erstellen**.

- **Über ein leeres Formular:** Dazu wählen Sie den Titel *Projektformulare* oder die Tabelle bzw. ein Formular in der Tabelle aus und klicken am unteren Rand auf das Pluszeichen **+**. Auf dem Bildschirm erscheint ein Dialogfenster, in dem Sie Name und Ordner des Formulars definieren. Klicken Sie auf die Schaltfläche **Bestätigen**, erscheint im Formulareditor ein leeres Formular. Weitere Informationen dazu finden Sie im Abschnitt **Leeres Formular erstellen**.

Formular bearbeiten

Im Explorer können Sie ein bestehendes Formular direkt öffnen und verändern. Es gibt zwei Möglichkeiten:

- Doppelklicken auf den Namen des Formulars
- Markieren Sie das Formular und wählen im Kontextmenü der Seite oder im Menü **Optionen** am unteren Rand den Eintrag **Formular bearbeiten**.

Weitere Informationen dazu finden Sie im Kapitel **Formulare erstellen**.

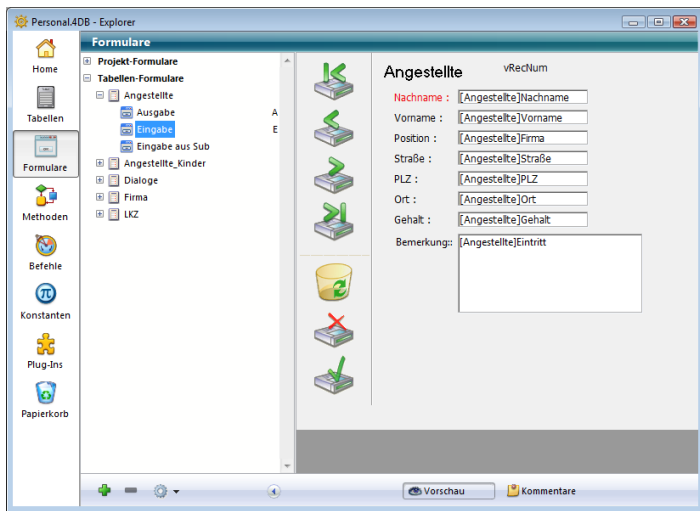
Formular löschen

Um ein Formular zu löschen, wählen Sie den Namen in der Liste aus und klicken auf das Icon **🗑** am unteren Rand oder wählen Sie im Kontextmenü den Eintrag **Formular löschen**. Sie können das Formular auch per Drag&Drop in den Papierkorb ziehen.

Sie können ein Formular nicht löschen, wenn es das standardmäßige Eingabe- bzw. Ausgabeformular in der Tabelle ist.

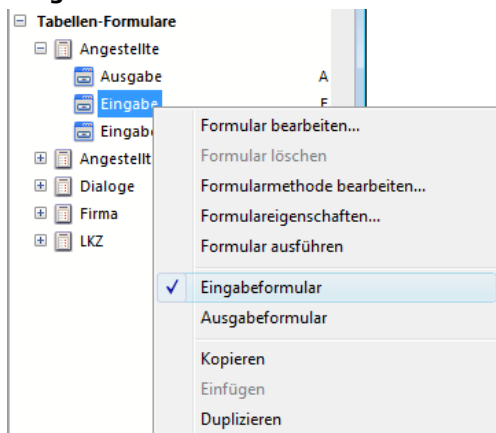
Vorschau

Um die Vorschau einzublenden, klicken Sie am unteren Rand auf das Icon **👁**. Markieren Sie das gewünschte Formular, damit dessen Vorschau im rechten Teil erscheint.



Aktuelles Eingabe- bzw. Ausgabeformular bestimmen

Im Explorer können Sie für jede Tabelle ein aktuelles Eingabe- und Ausgabeformular festlegen. Klicken Sie dazu in der hierarchischen Liste auf den gewünschten Formularnamen, und wählen im Kontextmenü den Eintrag **Eingabeformular** bzw. **Ausgabeformular**:



Weitere Informationen dazu finden Sie im Abschnitt [Eingabe- und Ausgabeformulare festlegen](#).

Formularmethode ändern

Sie können eine Formularmethode auch direkt auf der Seite Formulare des Explorers öffnen. Wählen Sie dazu im Kontextmenü oder im Menü **Optionen** am unteren Rand den Eintrag **Formularmethode bearbeiten**. Sie wird im Methodeneditor geöffnet.

Formulareigenschaften definieren

Im Explorer können Sie auf die wichtigsten Formulareigenschaften zugreifen. Klicken Sie dazu auf den Namen des gewünschten Formulars und wählen im Kontextmenü der Seite oder im Menü **Optionen** am unteren Rand den Eintrag **Formulareigenschaften**. Weitere Informationen dazu finden Sie im Abschnitt [Formulareigenschaften \(Explorer\)](#).

Formular ausführen

Sie können ein Formular in seinem Kontext (Liste der Datensätze für Listenformular, aktueller Datensatz für Detailformular) im Fenster zum Anzeigen der Datensätze ausführen. Wählen Sie dazu im Kontextmenü der Seite bzw. im Menü **Optionen** am unteren Rand den Eintrag **Formular ausführen**. Weitere Informationen dazu finden Sie im Abschnitt [Werkzeugleiste verwenden](#).

Drag and Drop verwenden

Sie können einen Formularnamen durch Ziehen bei gedrückter Maustaste in einer Methode hinzufügen. Hierbei erscheint der Formularnamen mit der korrekten Syntax. Ziehen Sie z.B. das Formular "Eingabe" in die Tabelle [Firmen], erscheint es in der Methode als "[Firmen] Eingabe".

Sie fügen einem Formular ein Unterformular hinzu, indem Sie bei gedrückter Maustaste den Namen des Listenformulars von der Formularseite des Explorers in das offene Formular des Formulareditors ziehen. Sie fügen ein Eingabe-Unterformular hinzu, wenn Sie die **Umschalttaste** gedrückt halten und bei gedrückter Maustaste den Namen eines Eingabeformulars von der Formularseite des Explorers in den Unterformular-Bereich im Formular ziehen. Weitere Informationen dazu finden Sie im Kapitel [Unterformulare und Widgets](#).

Formular duplizieren oder kopieren

Sie können vorhandene Formulare über die entsprechenden Befehle des Kontextmenüs oder des Menüs **Optionen** am unteren Rand kopieren und einfügen, sowie duplizieren. Auf diese Weise können Sie rasch und problemlos mehrere Formulare mit denselben Merkmalen entwickeln.





Ist am Zielort bereits ein gleichnamiges Formular vorhanden, fügt 4D an den Namen automatisch eine Nummer an bzw. erhöht sie bei Bedarf.

Die Seite Methoden zeigt die Projektmethoden, Datenbankmethoden und Trigger sowie die Projekt- und Tabellen-Formularmethoden für die Anwendung. Die Methodentypen sind nach Kategorie gruppiert.

Hinweis: Sind in der Anwendung auch Komponenten mit gemeinsam genutzten Methoden installiert, enthält die Seite zusätzlich die Kategorie Komponent-Methoden. Sie zeigt die installierten Komponenten und die gemeinsam genutzten Methoden. Weitere Informationen dazu finden Sie im Abschnitt **Komponenten anzeigen**.

Neue Methode erstellen

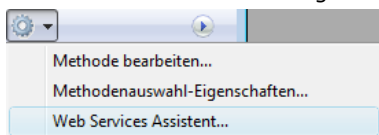
Dieser Abschnitt beschreibt, wie Sie die verschiedenen Methodenarten im Explorer erstellen.

- **Projektmethoden:** Um eine neue Projektmethode zu erstellen, wählen Sie in der hierarchischen Liste den Eintrag Projektmethode oder eine vorhandene Projektmethode aus und klicken am unteren Rand auf das Icon . Sie können eine Projektmethode auch durch Duplizieren oder Kopieren anlegen (siehe unten).
- **Komponent Methoden:** Diese Kategorie erscheint, wenn Komponenten mit gemeinsam genutzten Methoden in der Datenbank installiert sind. Sie zeigt die installierten Komponenten mit den dazugehörigen Methoden. Weitere Informationen dazu finden Sie im Abschnitt **Komponenten anzeigen**.
- **Datenbankmethoden:** Sie können keine neuen Datenbankmethoden erstellen. Sie können jedoch in einer leeren Datenbankmethode Programmiercode hinzufügen. Hierzu erweitern Sie den Eintrag Datenbankmethode und doppelklicken auf die gewünschte Datenbankmethode.
- **Trigger:** Um einen Trigger zu erzeugen, erweitern Sie in der hierarchischen Liste den Eintrag Trigger und markieren die gewünschte Tabelle. Doppelklicken Sie auf den Tabellennamen oder klicken Sie am unteren Rand auf das Icon .
- **Projekt-Formularmethoden:** Um eine neue Methode für ein Projektformular zu erstellen, öffnen Sie in der hierarchischen Liste den Eintrag Projekt-Formularmethoden, erweitern die gewünschte Tabelle und markieren dann das gewünschte Formular. Doppelklicken Sie auf den Formularnamen oder klicken Sie am unteren Rand auf das Icon .
- **Tabellen-Formularmethoden:** Um eine neue Methode für ein Tabellenformular zu erstellen, öffnen Sie in der hierarchischen Liste den Eintrag Tabellen-Formularmethoden, erweitern die gewünschte Tabelle und markieren dann das gewünschte Formular. Doppelklicken Sie auf den Formularnamen oder klicken Sie am unteren Rand auf das Icon .
- **4D Mobile Methoden:** Hier erscheinen Tabellen, die in 4D Mobile veröffentlicht sind und alle dazugehörigen Projektmethoden. Weitere Informationen dazu finden Sie im Abschnitt **Die 4D Datenbank konfigurieren** des Handbuchs *4D Mobile*.

Hinweis: Sie können für diese Methoden auch im Kontextmenü der Seite oder im DropDown-Menü Optionen am unteren Rand den Eintrag **Methode bearbeiten** wählen.

Methode mit Web Services Assistent erstellen

Sie können den Web Services Assistenten über die Seite Methoden aufrufen. Wählen Sie dazu im DropDown-Menü Optionen am unteren Rand den Eintrag Web-Services Assistent.



Auf dem Bildschirm erscheint der Web Services Assistent, bereit zum Abrufen von Web Services und zum Erstellen entsprechender Proxy-Methoden. Weitere Informationen dazu finden Sie im Abschnitt **Web Service in 4D abonnieren**.


Methode löschen

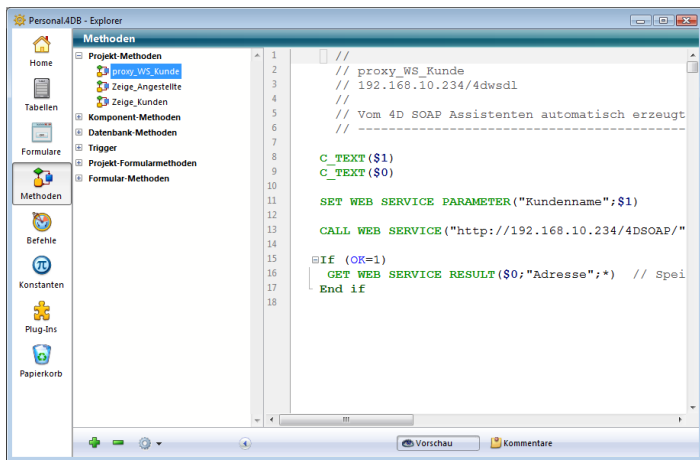
Sie können jede Methode löschen. Wählen Sie dazu die gewünschte Methode aus und klicken am unteren Rand auf das Icon  oder wählen Sie im Kontextmenü der Seite den Eintrag **Methode löschen**.

Der Löschvorgang richtet sich nach dem Methodentyp:

- Datenbankmethode, Trigger und Formularmethode: Der Inhalt der Methode wird entfernt, die Methode ist standardmäßig wieder leer. Eine Meldung warnt Sie, dass der Inhalt definitiv gelöscht wird.
- Projektmethode: Die Methode wird aus der Methodenliste entfernt und in den Papierkorb gelegt. Sie können diese daraus wieder entnehmen oder definitiv entfernen, wenn Sie den Papierkorb leeren. Weitere Informationen dazu finden Sie im Abschnitt **Seite Papierkorb**.

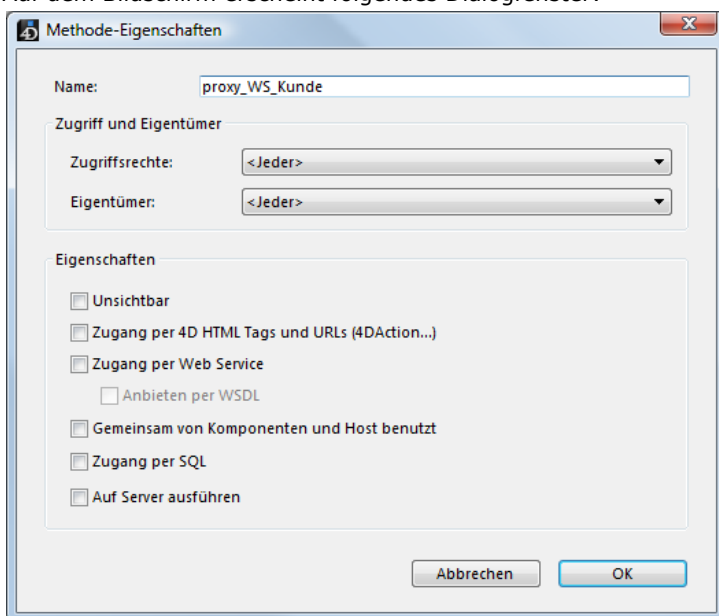
Vorschau einer Methode

Klicken Sie am unteren Rand auf das Icon , um den Vorschaubereich anzuzeigen und markieren Sie die gewünschte Methode. Der Inhalt der Methode erscheint wie im Methodeneditor. Sie können den gesamten Inhalt oder Teile aus dem Vorschaubereich auswählen, kopieren oder per Drag&Drop bewegen.



Eigenschaften der Projektmethoden

Sie können die Eigenschaften von Projektmethoden im Explorer anzeigen und verändern. Markieren Sie dazu eine Methode und wählen im Kontextmenü der Seite oder am unteren Rand im Menü Optionen den Eintrag **Methode-Eigenschaften**. Auf dem Bildschirm erscheint folgendes Dialogfenster:



Hier können Sie den Name und Eigenschaften der Methode verändern. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Projektmethoden**.

Methodenauswahl-Eigenschaften

Über den Befehl **Methodenauswahl-Eigenschaften** des Kontextmenüs oder des Menüs Optionen am unteren Rand (gilt nur für Projektmethoden) lässt sich ein Attribut (Unsichtbar, Zugang per 4D HTML Tags und URLs (4DACTION...), etc.) für alle oder Teile der Projektmethoden der Anwendung in einer einzigen Operation setzen. Weitere Informationen dazu finden Sie im Abschnitt **Methodeneigenschaften als Gruppe bearbeiten**.

Methode ausführen

Sie können die Ausführung einer Datenbankmethode bzw. Projektmethode im Explorer starten. Markieren Sie dazu die Methode und wählen im Kontextmenü der Seite oder im Menü Optionen am unteren Rand den Eintrag **Methode ausführen**. Die Methode wird im Standardmodus ausgeführt. Weitere Informationen dazu finden Sie im Abschnitt **Methoden ausführen**.

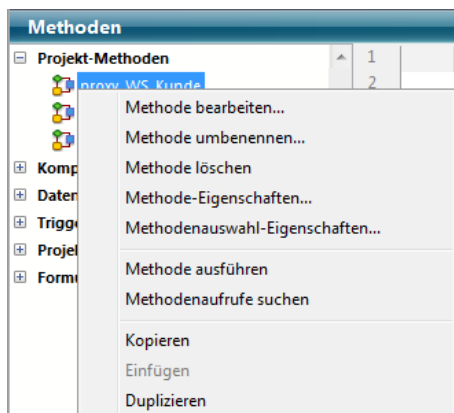
Projektmethoden duplizieren oder kopieren

Sie können Projektmethoden direkt aus der Methodenliste im Explorer kopieren und einfügen, sowie duplizieren. Auf diese Weise können Sie eine Methode in der gleichen Anwendung duplizieren oder von einer Anwendung in eine andere kopieren. Um eine Methode zu kopieren bzw. zu duplizieren, markieren Sie die entsprechende Methode und wählen im Kontextmenü der Seite oder im Menü Optionen am unteren Rand den entsprechenden Eintrag und dann im gleichen Menü den Befehl **Einfügen**.

Beim Duplizieren oder wenn beim Einfügen am Zielort bereits eine gleichnamige Methode vorhanden ist, erhält der Name der hinzugefügten Methode eine Nummer, z.B. *Meinmethode1*. Diese Nummer wird bei Bedarf erhöht, also *Meinmethode2*, *Meinmethode3*, etc.

Nach Aufrufen suchen

Sie können im Explorer nach der Liste der Objekte suchen, die eine bestimmte Projektmethode, andere Methoden oder Menüs aufrufen. Dazu rufen Sie im Kontextmenü der Seite oder im Menü **Optionen** am unteren Rand den Eintrag **Methodenaufrufe suchen** auf:



Hinweis: Der Befehl **Methodenaufrufe suchen** ist auch im Menü **Methode** des **WA SET URL FILTERS** verfügbar. Das Ergebnis erscheint in einem neuen Fenster. Weitere Informationen dazu finden Sie im Abschnitt **Suchergebnis**.

Abhängigkeiten suchen

Sie können jetzt automatisch nach Abhängigkeiten von Projektmethoden suchen, d.h. nach anderen Projektmethoden und Formularen, die direkt oder indirekt von diesen Methoden verwendet werden. Dazu markieren Sie eine Projektmethode im Explorer und wählen im Kontextmenü oder im Menü Optionen auf dieser Seite den Eintrag **Abhängigkeiten suchen**.

Alle Abhängigkeiten zu dieser Methode erscheinen dann in einem standardmäßigen Fenster **Suchergebnis**. Sie können sie in den jeweiligen Editoren öffnen oder auswählen und per Drag and Drop bewegen. Auf diese Weise können Sie mühelos komplette Funktionalitäten zwischen Ihren Datenbanken übertragen.

Sie können auch mehrere Projektmethoden gleichzeitig markieren und **Abhängigkeiten suchen** auswählen. In diesem Fall werden die Abhängigkeiten all dieser Methoden im Ergebnisfenster angezeigt.

Analyse der Abhängigkeiten

Um abhängige Projektmethoden zu finden, durchläuft 4D den Code der gesuchten Methode:

- direkte Aufrufe im Code
- die Namen von Methoden, die in einem der folgenden Befehle als Parameter übergeben wurden:
 - **Open window** (Parameter *Kontrollmenü-Box*)
 - **ON ERR CALL**
 - **ON EVENT CALL**
 - **New process**
 - **EXECUTE ON CLIENT**
 - **SET MENU ITEM METHOD**
 - **EXECUTE METHOD**

Sie müssen einen Parameternamen übergeben, der eine Konstante vom Typ Text ist und zum Namen einer gültigen ausführbaren Methode passt.

Um ein abhängiges Formular im Code zu finden, sucht 4D nach einem Formularnamen, der in einer der folgenden Befehle als Parameter übergeben wurde:

- **Print form**
- **DIALOG**
- **FORM SET OUTPUT**
- **FORM SET INPUT**
- **PRINT SETTINGS**
- **FORM GET PROPERTIES**
- **Open form window**

Der Parameter *Tabelle* des Formulars wird erkannt, muss aber nicht explizit angegeben werden.

In folgenden Fällen findet 4D unter Umständen keine Abhängigkeit:

```
//Beispiel für nicht gefundene Abhängigkeit
//Methode über Referenz
v:="myhandler"
ON ERR CALL(v)
//Zeiger auf eine Tabelle
DIALOG(Table(1)->,$formname)
// use of DEFAULT TABLE
DEFAULT TABLE([myTable])
DIALOG("myForm") // "myForm" wird als Projektformular gewertet
// Eine Funktion über FN in SQL Code aufrufen
Begin SQL
```

```
SELECT Film_Title, {FN How_Many_Actors(ID) AS NUMERIC}  
...
```

Seite Befehle

Die Seite **Befehle** zeigt alle in 4D enthaltenen Befehle, unterteilt in Kategorien. Sie ist dasselbe wie die Befehlsliste im Methodeneditor.

Über diese Seite können Sie auf die Online-Dokumentation zu den Befehlen zugreifen und in der Vorschau erscheint die Syntax der Befehle.

Zugriff auf Online Dokumentation

Doppelklicken Sie auf einen Befehlsnamen, erscheint die dazugehörige HTML Seite in Ihrem Browser.

Die HTML Dokumentation kann auf einer CD-Rom, einer DVD oder auf Ihrer Festplatte liegen bzw. direkt von der Web Site von 4D Deutschland kommen. Von wo aus die Seite geladen wird, richtet sich nach dem festgelegten Parameter in den Einstellungen auf der Seite **Allgemein>Dokumentation**. Weitere Informationen dazu finden Sie im Abschnitt **Seite Allgemein**. 4D stellt standardmäßig die Verbindung zur Web Site von 4D Doc Center her.

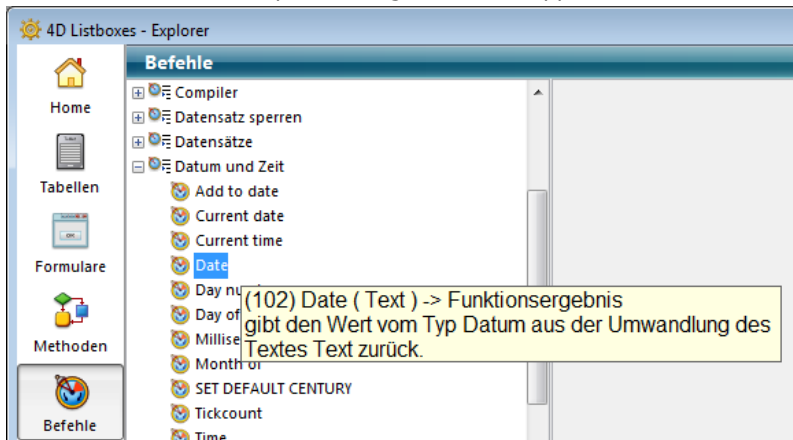
Befehlssyntax anzeigen

Ist die Vorschau eingblendet, zeigt sie die Nummer des ausgewählten Befehls und die Beschreibung der Syntax.

Hinweis: Die Nummer des Befehls wird von der 4D Funktion **Command name** verwendet.

Wollen Sie die Syntax nicht anzeigen, klicken Sie unter Windows mit gedrückter **Alt-Taste**, auf Mac OS mit gedrückter **Wahltaste** auf den Befehlsnamen.

Der Listenbereich des Explorers zeigt auch Hilfetipps mit Nummer und Syntax der Befehle.



Diese Hilfetipps sind die gleichen wie im Bearbeitungsbereich des Fensters **Methodeneditor**.

Drag and Drop verwenden

Sie können einen Befehl per Drag&Drop in eine Methode einfügen. Wählen Sie diesen im Explorer aus und ziehen ihn in den Methodeneditor. Der Befehl wird dann mit seiner Syntax eingefügt.

Die Seite **Konstanten** enthält eine hierarchische Liste aller Konstanten, die in den Methoden verwendet werden können. Sie können den Wert der markierten Konstanten entweder im Vorschaubereich sehen, wenn er aufgeklappt ist, oder über Hilfetipps, die im Listenbereich erscheinen.

Die angezeigten Konstanten können von 4D, einem Plug-In oder eigenen Ressourcen stammen. Weitere Informationen zur Verwendung von Konstanten in Methoden finden Sie im Handbuch *4D Programmiersprache* unter **Konstanten**.

Drag and Drop verwenden

Sie verwenden in Ihren Methoden häufig Konstanten. Anstatt sie zu schreiben, können Sie diese einer Methode vom Explorer aus hinzufügen. Markieren Sie die gewünschte Konstante in der Liste und ziehen sie in den Methodeneditor. Sobald der Methodeneditor diese Zeile des Programmcodes analysiert, wird die Konstante standardmäßig unterstrichen.

Seite Plug-Ins

Die Seite Plug-Ins listet alle Plug-Ins mit den dazugehörigen Befehlen auf, die in der Anwendung installiert sind.

Die Befehle sind nach Themen sortiert. Sie können diese per Drag&Drop in Ihre Methoden einfügen. In der Vorschau erscheinen zusätzliche Informationen sowie die Position der aktiven Dateien.

Hinweis: Weitere Informationen zum Installieren von Plug-Ins in Ihre Anwendung finden Sie im Abschnitt [Plug-Ins oder Komponenten installieren](#).

Die Seite Papierkorb im Explorer sammelt alle Ordner, Tabellen, Objekte von Formularen oder Projektmethoden, die in der Anwendung gelöscht wurden.

Der Papierkorb in 4D ist – wie der Papierkorb unter Windows bzw. auf Mac OS – ein Element zur Zwischenlagerung. So wird verhindert, dass Einträge aus Versehen definitiv gelöscht werden.

Objekte im Papierkorb erscheinen nicht mehr in den 4D Editoren oder Menüs, sie lassen sich weder ändern, verwenden noch bewegen. Folglich lassen sich andere ähnliche Objekte mit demselben Namen erstellen. Objekte im Papierkorb sind wiederverwendbar, so lange der Papierkorb nicht geleert wurde (siehe unten).

Sie können die Objekte in der Vorschau der Seite Papierkorb anzeigen. Außerdem bleiben Kommentare, die den gelöschten Objekten zugeordnet sind, sichtbar.

4D Server: Der Inhalt des Papierkorbs ist für alle Client-Rechner gleich.

Objekte löschen

Sie können Objekte aus den Seiten Home (Ordner), Formulare und Methoden löschen. Dazu wählen Sie das Objekt und dann im Kontextmenü (rechte Maustaste) des Explorers den Eintrag **Löschen**. Sie können es auch in den Papierkorb ziehen oder am unteren Rand auf das Icon klicken.

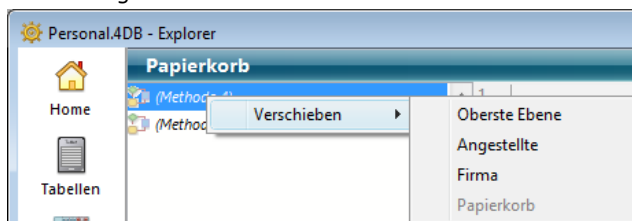
Auf der Seite **Home** lässt sich der Papierkorb auch als Zielort im Kontextmenü **Verschieben >** auswählen.

Sie können auch eine Tabelle in den Papierkorb schieben. Die Tabelle verschwindet dann aus den Editoren der Datenbank, inkl. Struktureditor. Beim Entfernen einer Tabelle werden die dazugehörigen Formulare automatisch mitentfernt.

Formulare mit folgenden Attributen können Sie nicht löschen: E (aktuelles Eingabeformular), A (aktuelles Ausgabeformular) oder B (Formular für Ein- und Ausgabe); außer die Tabelle dieses Formulars wurde gelöscht. Ebenso wenig können Sie Ordner löschen, die ein nicht-löschbares Objekt enthalten.

Objekt wiederherstellen

Sie können ein Objekt aus dem Papierkorb jederzeit wiederherstellen, solange dieser nicht geleert wurde. Dazu wählen Sie das Objekt und ziehen es auf das Icon der gewünschten Seite im linken Fensterteil oder wählen im Kontextmenü der Liste den Eintrag **Verschieben >**:



Das DropDown-Menü listet die Ordner auf, die in der Datenbank vorhanden sind. Weitere Informationen dazu finden Sie im Abschnitt **Seite Home**. Enthält die Datenbank keine Ordner, ist nur der Eintrag **Oberste Ebene** verfügbar. Wählen Sie einen Befehl aus diesem Menü, wird die Objektauswahl wiederhergestellt und in den gewählten Ordner gesetzt. Wählen Sie **Oberste Ebene**, wird die Auswahl in keinem Ordner, sondern auf der ersten Ebene positioniert.

Haben ein oder mehrere Objekte der Datenbank denselben Namen wie andere Objekte, die Sie wiederherstellen wollen, zeigt 4D eine Meldung. Sie können dann folgendes ausführen:

- Das Objekt in der Anwendung ersetzen (Formulare, Methoden, etc.)
- Das wiederherzustellende Objekt mit *Name_1* umbenennen (Tabellen).

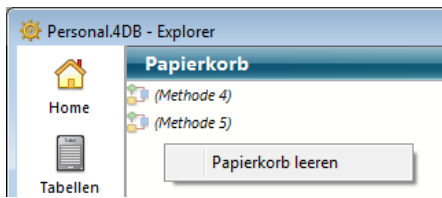
Die Schaltfläche **Ja** benennt das wiederherzustellende Objekt um oder ersetzt das vorhandene Objekt. Die Schaltfläche **Ja für alle** benennt alle Objekte der Auswahl um oder ersetzt sie. Klicken Sie auf **Nein** oder **Nein für alle**, werden die Objekte der Auswahl nicht wiederhergestellt.

Nur das angegebene Objekt wird wiederhergestellt. Formulare, die mit einer Tabelle gelöscht wurden, werden nicht automatisch mit der Tabelle wiederhergestellt.

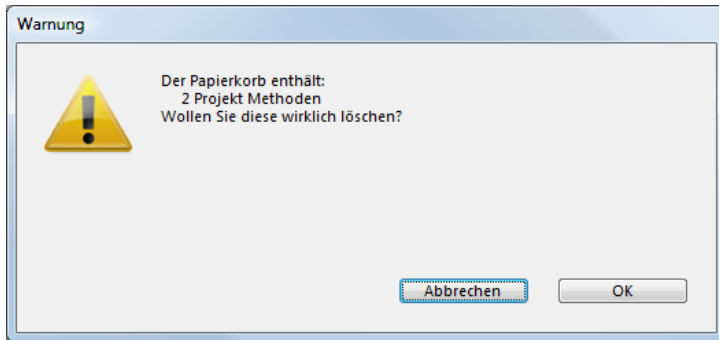
Papierkorb leeren

Gelöschte Objekte bleiben standardmäßig im Papierkorb, bis er geleert wird. Sie können den Inhalt des Papierkorbs jederzeit löschen, z.B. vor dem Komprimieren, um die Größe der Strukturdatei zu verringern. Alle Objekte im Papierkorb werden beim Leeren definitiv aus der Datenbank gelöscht.

Um den Papierkorb zu leeren, wählen Sie im Kontextmenü des Explorers (rechte Maustaste) bzw. im Menü Optionen am unteren Rand den Eintrag **Papierkorb leeren** auf:



Eine Meldung auf dem Bildschirm gibt die Anzahl und Art der löschbaren Objekte im Papierkorb an. Sie können die Operation bestätigen oder abbrechen:



Klicken Sie auf die Schaltfläche **OK**, um alle Objekte im Papierkorb zu löschen.

Kommentare verwenden

Im Explorer können Sie den Objekten Ihrer Datenbank Kommentare zuordnen. Das ist besonders hilfreich für Anwendungen, an denen mehrere Programmierer entwickeln und Ausdruck einer guten Programmierung. Ihre Kommentare erscheinen im Explorer in der Vorschau.

Sie können folgenden Objekten Kommentare zuordnen:

- Ordner
- Methoden (Datenbank-, Komponent-, Projekt-, Formarmethoden und Trigger)
- Formulare
- Tabellen
- Felder
- Plug-Ins

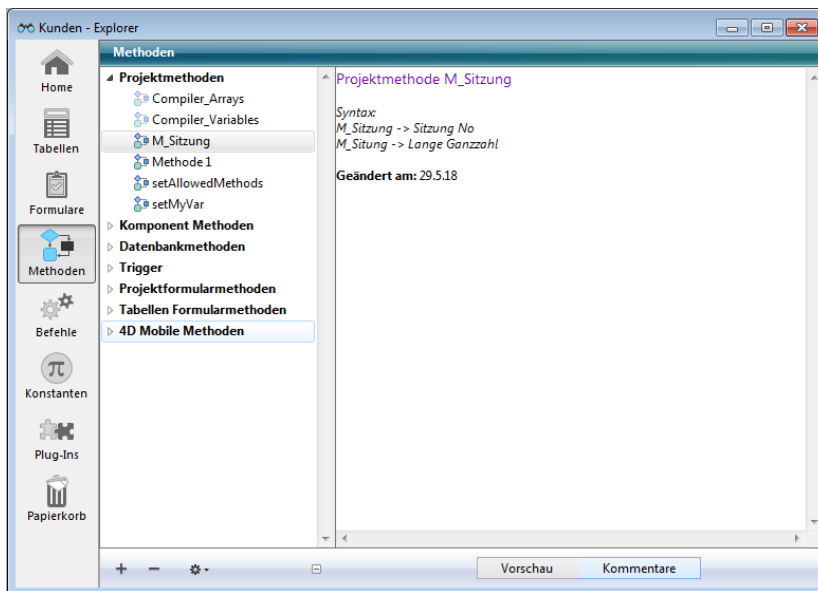
Ein 4D Kommentar lässt sich als Text eingeben, den Sie jederzeit im Explorer anzeigen und ändern können. Der Text kann unterschiedliche Schriftstile, Farben, u.v.m. haben. Ein Kommentar kann ein Objekt beschreiben oder informieren, wie das Objekt in der Datenbank funktioniert. Die angelegten Kommentare werden in der Struktur der Datenbank gespeichert. Sie können außerdem **automatische Kommentare** erstellen, d.h. 4D gibt automatisch Kommentare ein, wenn ein Objekt erstellt oder geändert wird.

Hinweis zur Kompatibilität: Ab 4D Version 12 erscheinen Kommentare, die im Kopfteil der Methode mit den Zeichen // eingefügt wurden, als Hilfetipps, wenn in einer anderen Methode auf diese Methode verwiesen wird (siehe Abschnitt "Hilfetipps verwenden" unter **WA SET EXTERNAL LINKS FILTERS**). Diese Funktionalität ist besonders hilfreich zum Dokumentieren von Benutzermethoden. Sie ist jedoch nicht kompatibel mit Kommentaren, die im Explorer gesetzt werden. Wollen Sie Kopfteile von Methoden auf diese Weise verwenden, geben Sie keine Kommentare im Explorer ein.

Kommentar einem Objekt zuordnen

Sie können Kommentare im Explorer erstellen, ansehen und ändern. Um einen Kommentar hinzuzufügen, wählen Sie bei aktivem Vorschaubereich ein Objekt in der Liste des Explorers aus und klicken auf die Schaltfläche **Kommentare** unter der Vorschau.

Jetzt erscheint anstelle der Vorschau der Kommentarbereich. Hier können Sie Text eingeben bzw. ändern.



4D Server: Das Icon zum Sperren in der linken unteren Ecke des Vorschaubereichs gibt an, ob der Kommentar derzeit im Netzwerk von einem anderen Benutzer geändert wird. In diesem Fall ist der Bleistift durchgestrichen. Der Kommentar kann dann nur gelesen, aber nicht verändert werden.

Der Text wird gesichert, sobald Sie außerhalb des Eingabebereichs klicken. Sie können pro Objekt bis zu 32 KB (32 700 Zeichen) eingeben. Der Kommentar kann dann nur gelesen, aber nicht verändert werden.

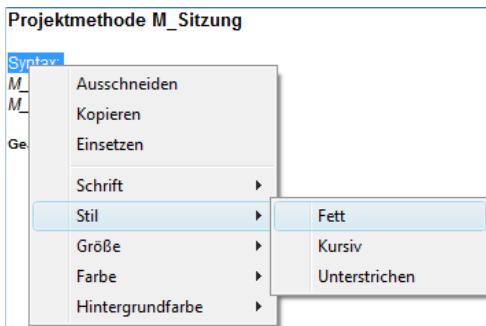
Im Kommentarbereich können Sie die Standardbefehle für Textverarbeitung, also Kopieren, Einfügen, Alles auswählen, etc. über das Kontextmenü oder Tastaturkürzel verwenden. Sie können auch wie in jedem normalen Textbereich über Tastaturkürzel im Text navigieren.

Stil für Kommentare ändern

Sie können Kommentare mit Fett- oder Kursivschrift gestalten und Schriftart und -größe verändern oder eine Farbe zuweisen. Gehen Sie folgendermaßen vor:

1. Wählen Sie im Kommentarbereich den entsprechenden Text aus.

2. Klicken Sie mit der rechten Maustaste in den Bereich.
Auf dem Bildschirm erscheint ein hierarchisches PopUp-Menü:



3. Wählen Sie die gewünschten Stilelemente aus.

Automatische Kommentare einfügen

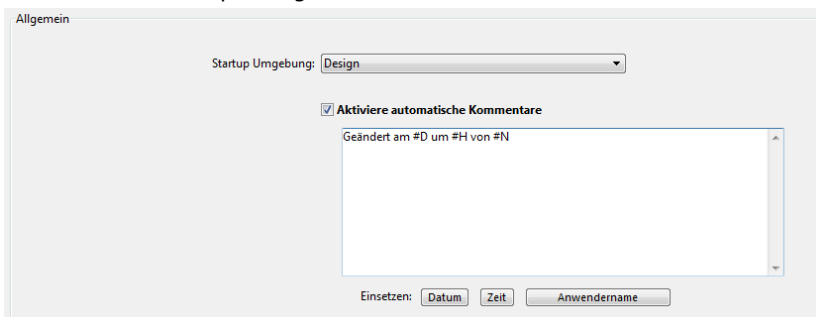
Für Methoden und Formulare der Datenbank können Sie ein automatisches Kommentarsystem aktivieren. Ist dieses System aktiv, wird jeder Methode bzw. jedem Formular, das in der Datenbank erstellt oder geändert wird, automatisch ein Kommentar zugeordnet. Das können statischer Text sein, z.B. „Geändert von“ oder Variablen, z.B. aktuelles Datum, aktueller Benutzername (wie im 4D Kennwortsystem festgelegt).

Automatische Kommentare werden in den Datenbank-Eigenschaften definiert (siehe **Aktiviere automatische Kommentare**). Sie können diese, genauso wie Standardkommentare, über den Explorer betrachten.

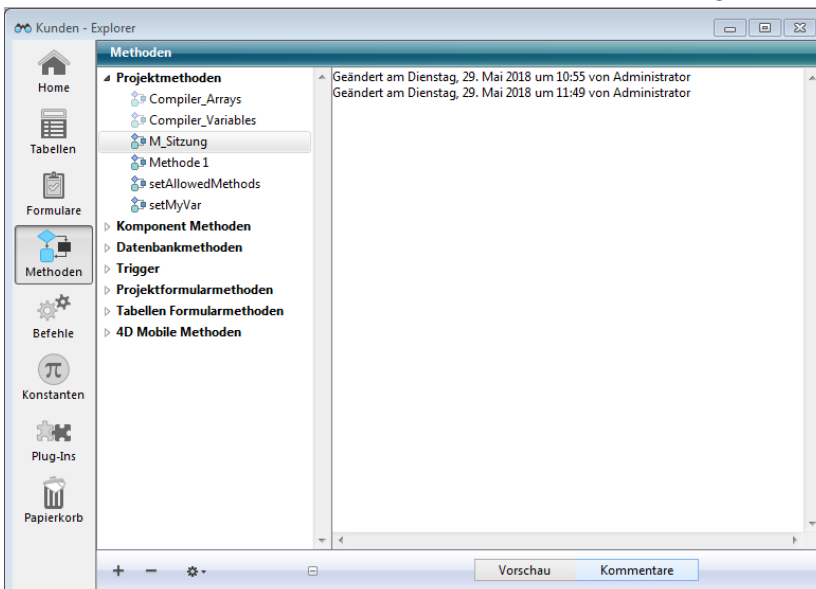
Gehen Sie folgendermaßen vor:

1. Wählen Sie im Menü **Design>Datenbank-Eigenschaften** oder klicken Sie in der Werkzeugleiste am Ende auf das Icon **Eigenschaften**.
2. Markieren Sie auf der Seite **Allgemein** die Option **Automatische Kommentare aktivieren**.
3. Geben Sie im darunterliegenden Bereich Text ein, der automatisch eingefügt werden soll.
Sie können statischen Text eingeben und Variablen über die Schaltflächen **Datum**, **Zeit** und **Anwendername** unter dem Bereich. Sie können die Variablen auch direkt eingeben:
#D für Datum
#H für Zeit
#N für den aktuellen Anwender

Geben Sie zum Beispiel folgende Werte ein:








Der automatische Kommentar wird für Kommentare zu allen neuen Methoden und Formularen hinzugefügt, sowie zu allen vorhandenen Methoden und Formularen, die nach der Aktivierung automatischer Kommentare geändert werden:



Hinweis: Hat Ihre Datenbank kein Kennwortsystem, gibt #N den String Designer zurück.

4D Server: Jede Arbeitsstation mit Zugriff auf die Datenbank-Eigenschaften kann automatische Kommentare ändern. Sie können diese Parameter auch vom Server-Rechner aus ändern. Jede Änderung in den automatischen Kommentaren wird sofort von allen Arbeitsstationen berücksichtigt, sobald ein Objekt geändert und die dazugehörigen Kommentare bestätigt wurden.

Runtime Explorer

-  Zugriff auf den Runtime Explorer
-  Seite Überwachen
-  Seite Prozess
-  Seite Break
-  Seite Catch

Zugriff auf den Runtime Explorer

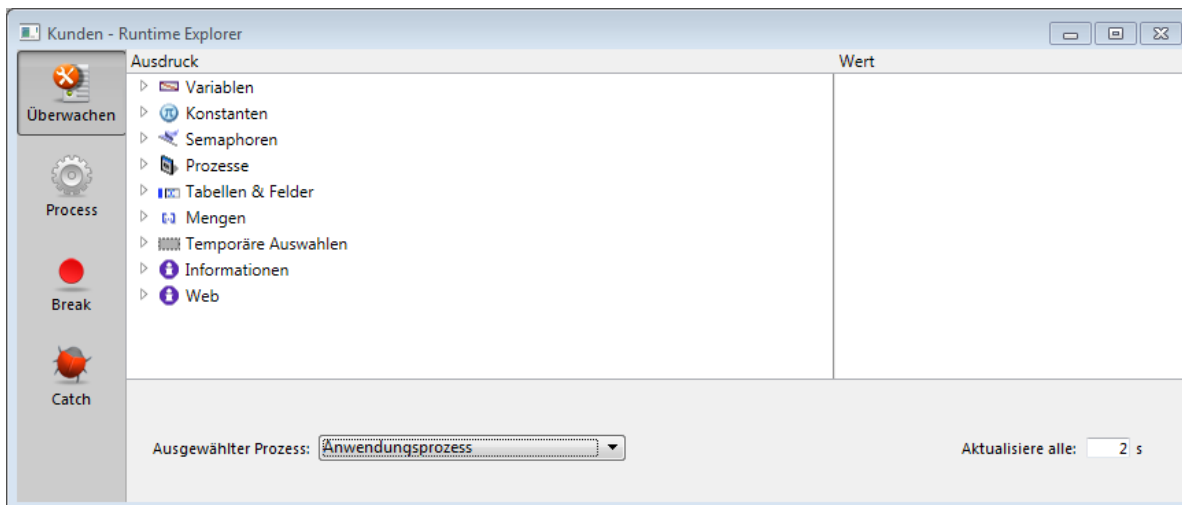
Über das Fenster Runtime Explorer können Sie das Verhalten der verschiedenen Strukturelemente in Ihrer Datenbank ansehen und prüfen, ob die verfügbaren Ressourcen korrekt arbeiten. Der Runtime Explorer ist besonders nützlich in der Phase der Datenbankentwicklung und -analyse.

Das Fenster lässt sich in der Design- und Anwendungsumgebung, im kompilierten oder interpretierten Modus anzeigen. In kennwortgeschützten Datenbanken können nur **Designer und Administrator** auf das Fenster Runtime Explorer zugreifen.

Der Runtime Explorer lässt sich auf zwei Arten anzeigen: In einem Standardfenster oder in einem Palettenfenster. Das Palettenfenster bleibt auch bei anderen offenen Fenstern im Vordergrund.

- Um den Runtime Explorer in einem Standardfenster anzuzeigen, wählen Sie im Menü **Start** den Eintrag **Runtime Explorer**.
- Um den Runtime Explorer in einem Palettenfenster anzuzeigen:
 - Drücken Sie unter Windows die **Strg-Taste+Umschalttaste+F9**,
 - auf Mac OS die **Befehlstaste+Umschalttaste+F9**
 - oder wählen Sie bei gedrückter Umschalttaste im Menü **Start>Runtime Explorer**.

Das Fenster Runtime Explorer hat vier Seiten, die Sie über die Icons am linken Rand öffnen können: **Überwachen**, **Prozess**, **Break** und **Catch**.



Seite Überwachen

Die Seite **Überwachen** ist ein Debugger. Sie zeigt Information über die Ausführung von Code in der Anwendung und dem am unteren Rand gewählten Prozess:

- **Ausgewählter Prozess:** Diese DropDown Liste enthält alle Prozesse *, die in der Datenbank ausgeführt werden. Hier können Sie den/die Prozesse auswählen, die Sie beobachten wollen.
- **Aktualisiere alle:** Hier tragen Sie ein, nach wieviel Sekunden die Information auf der Seite aktualisiert wird.

(* Im preemptive Modus ausgeführte Prozesse sind in dieser Liste nicht verfügbar (siehe unter **Preemptive 4D Prozesse**).

Die Spalte "Ausdruck" zeigt die Namen der Objekte und Ausdrücke. Die Spalte "Wert" zeigt den aktuellen Wert der Objekte und Ausdrücke. Sie können die verschiedenen Spalten im Verhältnis zueinander anpassen. Klicken Sie dazu auf die Trennungslinie und ziehen sie nach rechts oder links.

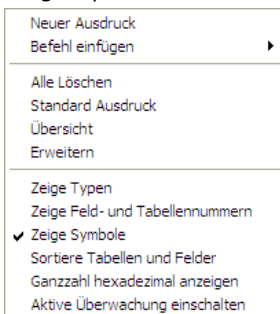
Wollen Sie den Wert eines Objekts ändern, markieren Sie ihn in der rechten Spalte. Das ist natürlich nur möglich, wenn der Wert veränderbar ist. Die hierarchische Liste enthält folgende Einträge:

- **Variablen**
- **Current Form Values**
- **Konstanten**
- **Semaphoren**
- **Prozesse**
- **Tabellen und Felder**
- **Mengen**
- **Temporäre Auswahlen**
- **Information**
- **Web**

Diese Gruppen sind identisch mit den im 4D Debugger angezeigten Gruppen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Bereich Überprüfung**.

Wollen Sie einen Ausdruck oder eine Gruppe löschen, wählen Sie die entsprechende Zeile aus und drücken die Taste **Löschen** oder **Rückschritt**.

Einige Operationen können Sie auch über das Kontextmenü ausführen:



Sie können auch einen **Neuen Ausdruck**, einen **4D Befehl** hinzufügen, oder globale Aktionen durchführen wie **Alles löschen**, Zeige alle **Standardausdrücke**, **Alle Untergruppen schließen** oder **Alle Gruppen erweitern**.

Hinweis: Einen neuen Ausdruck können Sie auch durch Doppelklick in die Spalte **Ausdruck** hinzufügen.

Im unteren Teil des Kontext-Menüs gibt es weitere Anzeigeeoptionen:

- **Zeige Typen** blendet in der Liste Tabellen & Namen die Feldtypen neben den Feldnamen ein oder aus.
- **Zeige Feld- und Tabellennummern** blendet in der Liste Tabellen & Namen die Feldnummern neben den Feldnamen ein oder aus. Für jedes Feld wird folgendes Format angewandt: [TabellenNum]FeldNum.
- **Zeige Symbole** blendet die Objektsymbole in der hierarchischen Liste ein oder aus.
- **Sortiere Tabellen und Felder** sortiert die Liste der Tabellen und Felder alphabetisch. Standardmäßig erscheinen diese Objekte in der Reihenfolge, wie sie erstellt wurden.
- **Ganzzahl hexadezimal anzeigen** zeigt die Variablen vom Typ Ganzzahl oder Lange Ganzzahl in hexadezimaler Form an.
- **Aktive Überwachung einschalten** zeigt zusätzliche Information zu Zeitplan und Kommunikation über Netzwerk. Hier können Sie die interne Aktivität der Anwendung überwachen. Bedenken Sie jedoch, dass diese Option den Ablauf von Prozessen verlangsamt.

Auf der Seite **Prozess** können Sie die verbrauchte CPU Zeit sehen sowie verschiedene Informationen zu den in der Anwendung erstellten Prozessen. Für Prozesse gibt es auch verschiedene Werkzeuge; davon ausgenommen sind Prozesse des Systems.

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Einführung in Prozesse**.

Hinweis zu 4D Server: Die Seite **Prozess** des Runtime Explorer auf einem remote 4D steuert die Prozesse dieses Clients. Die **Seite Prozesse** des Verwaltungsfenster von 4D Server steuert alle Prozesse auf allen Client-Rechnern, die an den Server angemeldet sind.
























Information über die Prozesse

Die Seite zeigt für jeden Prozess folgende Informationen:

- Prozesstyp (Icon)
- Prozessnummer (Prozess ID, identisch mit der Prozessnummer). Sie verwenden diese Prozessnummer zum Identifizieren eines spezifischen Prozesses in Befehlen und Funktionen).
- Prozessname
- Aktueller Status des Prozesses
- Ausführungszeit in Sekunden, die der Prozess seit dem Starten beansprucht hat.
- Vom Prozess verbrauchte CPU Zeit in Prozent.

Prozesstypen

Jedem Prozess ist ein Icon zugeordnet; Farbe und Form kennzeichnen die Art des Prozesses

-  Anwendungsserver
-  SQL Server
-  DB4D Server (Datenbank-Engine)
-  Web Server
-  SOAP Server
-  Geschützter 4D Client Prozess (Prozess für Entwicklungsumgebung eines Client)
-  4D Client Hauptprozess (Hauptprozess eines Client), Spiegelprozess eines 4D Client auf dem Server
-  4D Client Basisprozess (Prozess existiert parallel zu einem 4D Client Prozess; preemptive Prozess zur Kontrolle des jeweiligen 4D Client Hauptprozesses)
-  Reserve Prozess (vorheriger oder zukünftiger 4D Client Prozess)
-  SQL Server Worker Prozess
-  HTTP Server Worker Prozess
-  4D Client Prozess (auf einem Client gestarteter Prozess)
-  Serverprozedur (von einem angemeldeten Client gestarteter Prozess auf dem Server)
-  Web Methode (z.B. über 4DACTION gestartet)
-  SOAP Methode (über einen Web Dienst gestartet)
-  Web Methode (preemptive)
-  Logger
-  TCP Connection Listener
-  TCP Session Manager
-  Anderer Prozess
-  Worker Prozess (kooperativ)
-  Serverprozedur (preemptive Prozess)
-  Worker Prozess (preemptive)

Hinweis: Jeder 4D Client Hauptprozess und sein „Zwilling“ 4D Client Basisprozess werden zusammengelegt, wenn die Option *Prozesse gruppiert anzeigen* markiert ist.

Prozessnummer

Jeder Prozess hat eine einmalige Nummer (in Klammern), die der Reihenfolge seiner Erstellung während der Arbeitssitzung entspricht. Beim Starten der Anwendung werden die ersten Nummern den automatisch von 4D angelegten Prozessen zugewiesen. Ihre Nummern variieren je nach den beim Start ausgeführten Servern.

Starten Sie einen eigenen Prozess, wird er unter den Standardprozessen als nächster Prozess aufgelistet oder nimmt den Platz eines abgebrochenen Prozesses ein. Zum Beispiel werden Prozess 7 und 8 ausgeführt. Wird Prozess 7 abgebrochen, wird der als nächster startende Prozess Prozess 7.

Hinweis: Prozesse werden automatisch nach der Ausführung abgebrochen. Sie können einen Prozess im Runtime Explorer abbrechen, bevor er abgeschlossen wird. Weitere Informationen dazu finden Sie im Abschnitt **Prozess abbrechen**.

Prozessname

Wenn Sie einen neuen Prozess mit der 4D Funktion **New process** oder **Execute on server** starten, können Sie seinen Namen als Parameter in der Funktion festlegen. Dieser Name erscheint als Prozessname in der Liste der Prozesse.

Legen Sie keinen Prozessnamen mit einer Funktion fest, weist 4D dem Prozess automatisch einen Standardnamen zu. Dabei gilt folgendes:

- **Prozesse, die mit einem Menübefehl gestartet werden**, erhalten den Standardnamen "ML_Prozessnummer". Wird zum Beispiel der Prozess Nummer 7 über Menübefehl gestartet, lautet sein Name "ML_7".
- **Prozesse, die durch Ausführen einer Methode gestartet werden**, erhalten den Namen "P_Prozessnummer". Wird zum Beispiel der Prozess Nummer 5 vom Programm gestartet, erhält der Prozess den Namen "P_5".
- **Prozesse, die mit einer Funktion gestartet, aber nicht explizit benannt werden**: Starten Sie einen Prozess über eine Methode, geben den Namen aber nicht als Parameter in der 4D Funktion **New process** an, bleibt der Prozessname leer.

Hinweis für 4D Server: Beginnt der Name eines Prozesses mit einem Dollarzeichen (\$), ist es ein lokaler Prozess, der keinen Zugriff auf Tabellen oder auf 4D Server hat (siehe **Globale und lokale Prozesse**).

Prozessstatus

Der Status eines Prozesses ist der aktuelle Status der Ausführung, d.h. was der Prozess momentan macht.

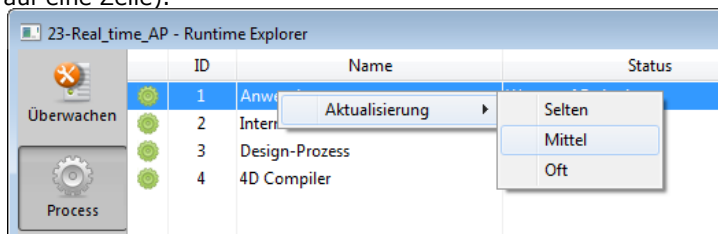
Es gibt folgende Statusmöglichkeiten im Fenster Runtime Explorer:

- **Ausführung:** Der Prozess wird momentan ausgeführt.
- **Schläft:** Der Prozess ruht für eine bestimmte Zeit. Während der Zeit, in der der Prozess ruht, beansprucht er keine Prozesszeit.
- **Wartet auf Ereignis:** Der Prozess wartet auf eine Aktion des Benutzers, wie z.B. das Klicken auf eine Schaltfläche oder das Auswählen eines Menüeintrags.
- **Wartet auf Eingabe/Ausgabe:** Der Prozess wartet auf eine Eingabe oder Ausgabe. Ein Prozess kann zum Beispiel warten, dass eine Gruppe von Datensätzen auf der Festplatte aktualisiert wird.
- **Wartet auf Semaphore:** Der Prozess wartet darauf, dass interne Prozesse die Ausführung von 4D Datenbankaufgaben abschließen.
- **Gestoppt:** Der Prozess wird angehalten, bis Sie ihn anweisen, mit der Ausführung fortzufahren. Während dieser Zeit beansprucht er keine Prozesszeit. Weitere Informationen finden Sie im Abschnitt **Prozess anhalten und fortsetzen**.
- **Abgebrochen:** Der Prozess wurde abgebrochen. In diesem Fall gibt 4D gesperrte Datensätze frei, löscht Transaktionen, die durch den Prozess geöffnet wurden, aber noch nicht übernommen oder gelöscht wurden, und gibt die aktuelle Auswahl und den aktuellen Datensatz frei. Prozesse werden nach der Ausführung automatisch abgebrochen. Sie können einen Prozess auch über den Runtime Explorer vor dem Abschließen abbrechen. Weitere Informationen dazu finden Sie im Abschnitt **Prozess abbrechen**.
- **Ausgeblendetes modales Fenster:** Ein Prozess, der ein modales Dialogfenster angezeigt hat, wurde ausgeblendet, so dass der Benutzer den Dialog nicht mehr sehen kann. Der Prozess wartet in diesem Status, bis der Dialog erneut angezeigt wird.

Prozesszeit

Der Runtime Explorer zeigt die Ausführungszeit für jeden Prozess. Beim Verwalten der Prozesse verteilt 4D die Arbeitszeit zwischen den bestehenden Prozessen, so dass keiner der Prozesse kontinuierlich ausgeführt wird. Somit ist die Prozesszeit die gesamte Ausführungszeit, die ein Prozess seit Beginn in Sekunden beansprucht hat, und nicht die Gesamtzeit, die seit Starten der Prozessausführung vergangen ist, da die Ausführung zwischen allen offenen Prozessen verteilt wird.

Die Zeitspanne, in welcher die Daten aktualisiert werden sollen, lässt sich über ein Kontextmenü einstellen (rechter Mausklick auf eine Zeile).



Je größer die Aktualisierungszeit ist, desto mehr CPU Zeit verbraucht der Prozess des Runtime Explorer.

Aktionen mit Prozessen

Mit dem Runtime Explorer können Sie die Ausführung der Prozesse durch Anhalten, Fortsetzen oder Abbrechen steuern. Sie können auch die Ausführung des Prozesses im Debug-Modus festlegen und die einzelnen Fenster jederzeit ein- oder ausblenden. Diese Operationen werden ausführlich in den folgenden Abschnitten behandelt.


Über die Schaltflächen unter der Liste der Prozesse können Sie einen Prozess ausblenden, einblenden oder nach vorne bringen. Sie können auch mehrere Prozesse gleichzeitig auswählen: (**Umschalttaste+Klick** für eine zusammenhängende Auswahl oder **Befehl/Strg-Taste+Klick** für eine nicht-zusammenhängende Auswahl).

Hinweis: Sie können einen Prozess auch für eine bestimmte Zeit verzögern. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter dem 4D Befehl **DELAY PROCESS**.


Prozess anhalten und fortsetzen

Sie können die Ausführung eines Prozesses vorübergehend aussetzen, indem Sie ihn anhalten. Sie können einen Prozess anhalten, um einem anderen Prozess mehr Ausführungszeit einzuräumen oder ein Ereignis zuzulassen, mit dem der Prozess auftritt.

Zum Beispiel können Sie einen Prozess starten, der eine Auswahl von Datensätzen druckt. Sie stellen aber dann fest, dass Sie die Daten in einem der Datensätze ändern wollen. Also halten Sie den Prozess an, nehmen Ihre Änderungen vor und setzen anschließend Ihren Prozess, das Drucken der Datensätze, fort.

- Um einen Prozess anzuhalten, markieren Sie ihn und klicken am unteren Rand auf die Schaltfläche **Pause** . Der Status des Prozesses in der Prozessliste ändert sich automatisch zu "Angehalten". Der Prozess bleibt unterbrochen, bis

Sie ihn anweisen, mit der Ausführung fortzufahren.


- Um einen Prozess fortzusetzen, markieren Sie ihn und klicken am unteren Rand auf die Schaltfläche **Ausführen**  .

Der Status des Prozesses kehrt zu dem Status zurück, den er vor dem Anhalten hatte. Wurde er zum Beispiel gerade ausgeführt, beginnt der Prozess wieder mit der Ausführung. Hatte er auf ein Ereignis gewartet, beginnt er wieder mit dem Warten auf ein Ereignis.

Prozess abbrechen

Ein Prozess wird nach seiner Ausführung automatisch abgebrochen. Manchmal müssen Sie jedoch einen Prozess schon vorher abbrechen, um Fehler zu beheben. Ein Prozess sollte nur zu diesem Zweck abgebrochen werden.

Wenn ein Prozess abgebrochen wird, gibt 4D gesperrte Datensätze frei, löscht Transaktionen, die durch den Prozess geöffnet, aber noch nicht übernommen oder gelöscht wurden, und gibt auch die aktuelle Auswahl und den aktuellen Datensatz frei.

Um einen Prozess abzubrechen, markieren Sie ihn und klicken am unteren Rand auf die Schaltfläche **Abbrechen**  . Sein Status im Runtime Explorer ändert sich automatisch zu "Abgebrochen".

Prozess verfolgen

Sie können die Fehler in einem Prozess beheben, wenn Sie dessen Ausführung im Debugger überwachen.

Um einen Prozess zu debuggen, markieren Sie ihn und klicken am unteren Rand auf die Schaltfläche **Schritt**  .

Der Debugger erscheint, wenn der Prozess ausgeführt wird. Hier können Sie Fehler im Prozess beheben, indem Sie seine Ausführung Schritt für Schritt nachvollziehen und Ausdrücke wie Datenfeldwerte und Variablen überprüfen, die in der Methode verwendet werden.

Hat der Prozess den Status „schläft“ oder „Pause“, speichert 4D die Anfrage und zeigt den Debugger, wenn der Prozess wieder aktiviert wird. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Debugging**.

Prozesse, die 4D intern erstellt und verwaltet, können Sie nicht debuggen.


Prozess ausblenden

Sie können einen Prozess in der Anwendungsumgebung ausblenden. Dann sind Fenster oder Menüs, die durch den Prozess erzeugt werden, für den Benutzer unsichtbar, während der Prozess ausgeführt wird.

Das Ausblenden von Prozessen ist sinnvoll bei Operationen, in denen Sie ein Fenster öffnen, das Sie später schließen wollen. Statt den Prozess abzubrechen, um das Fenster zu schließen, können Sie das Fenster für den Benutzer unsichtbar machen, indem Sie den Prozess ausblenden, der es geöffnet hat.

Um einen Prozess auszublenden, markieren Sie ihn und klicken am unteren Rand auf die Schaltfläche **Ausblenden:**  .


Der Prozess ist jetzt in der Anwendungsumgebung ausgeblendet, wird aber weiter ausgeführt. Sie können einen ausgeblendeten Prozess jederzeit wieder einblenden.

Um einen Prozess wieder anzuzeigen, klicken Sie am unteren Rand auf die Schaltfläche **Einblenden**  . Der Prozess erscheint wieder in der Anwendungsumgebung.

Prozess nach vorne bringen

Sie können ein Prozessfenster zum vordersten Fenster machen, wenn Sie dessen Prozess nach vorne bringen. Ist z.B. der Anwendungsprozess nach vorne gebracht, ist die Anwendungsumgebung auf dem Bildschirm vorne.

Sie können jeden Benutzerprozess nach vorne bringen. Haben Sie ein Fenster für einen Prozess erstellt, wird dieses Fenster zum vordersten Fenster auf dem Bildschirm. Ist eine Menüleiste mit dem Fenster verknüpft, bringt 4D diese nach vorne und die dazugehörigen Menüs sind jetzt die aktuellen Menüs.

Um einen Prozess nach vorne zu bringen, markieren Sie ihn und klicken am unteren Rand auf die Schaltfläche **Nach vorne bringen:**  .

Alle Fenster, die mit dem Prozess verknüpft sind, werden auf dem Bildschirm nach vorne gebracht. Außerdem erscheint die Menüleiste des vordersten Prozessfensters.

Seite Break







Auf der Seite **Break** können Sie die Unterbrechungspunkte ansehen und verwalten, die Sie für Ihre Datenbank gesetzt haben.

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Unterbrechungsliste**.

Seite **Catch**

Die Seite **Catch** zeigt die Unterbrechungspunkte, die in der Datenbank für Befehle oder Ausdrücke definiert wurden. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Befehle unterbrechen**.

Suchen und Ersetzen in der Struktur

-  Überblick
-  Suche ausführen
-  Suchergebnis
-  Ersetze in Inhalt
-  Umbenennen
-  Nach nicht verwendeten Elementen suchen

4D bietet verschiedene Funktionen zum Suchen und Ersetzen für Objekte in der gesamten Designumgebung.

- Sie können in Teilen oder in der gesamten Datenbankstruktur nach einem String suchen, z.B. mit den Suchkriterien „beginnt mit“, „enthält“. Sie können z.B. nach allen Variablen mit dem String „MeineVar“ nur in Methoden suchen, deren Namen mit "HR_" beginnt.
- Die Ergebnisse erscheinen in einem Fenster, wo Sie die Ersetzungen im Inhalt ausführen können. Sie können diese Ergebnisse auch in eine Textdatei exportieren, die sich in eine Tabellenkalkulation importieren lässt. Weitere Informationen dazu finden Sie im Abschnitt **Menü Optionen**.
- Sie können Variablen und Methoden finden, die in Ihrem Code nicht verwendet werden und sie entfernen, um Speicherplatz freizumachen.
- Sie können eine Projektmethode oder Variable in einer Operation in der gesamten Designumgebung umbenennen.

Hinweis: Es gibt auch Funktionen, um in Methoden in Ihrer Datenbank zu suchen, die im Kontextmenü der **Seite Methoden** im Explorer verfügbar sind: **Aufrufe suchen** und **Abhängigkeiten suchen**. Beide Funktionen zeigen die gefundenen Einträge in einem **Suchergebnis**.

Stellen für Suchläufe

Bei Suchläufen in der Designumgebung werden folgende Objekte standardmäßig gesucht:

- Menüs (Namen und Einträge) und Menüeinträgen zugewiesene Befehle
- Auswahllisten (Namen und Einträge)
- Bilder der Bibliothek (Namen)
- Hilfetipps (Namen und Inhalt)
- Namen von Formaten / Filtern (Namen und Inhalt)
- Namen von Tabellen und Feldern
- Namen von Formularen
- Namen von Projektmethoden
- Kommentare im Explorer
- Inhalt aller Methoden (Trigger, Datenbankmethoden, Projektmethoden, Objektmethoden)
- Inhalt von Formularen:
 1. Objektnamen
 2. Namen von Hilfetipps, die von Objekten verwendet werden
 3. Namen von Bildern, die von Objekten verwendet werden
 4. Namen von Variablen, die von Objekten verwendet werden
 5. Namen der Stilvorlagen, die von Objekten verwendet werden
 6. Formatierte Strings
 7. Titel von Objekten
 8. Referenzen von Feldern oder Tabellen

Suche ausführen

Suche starten

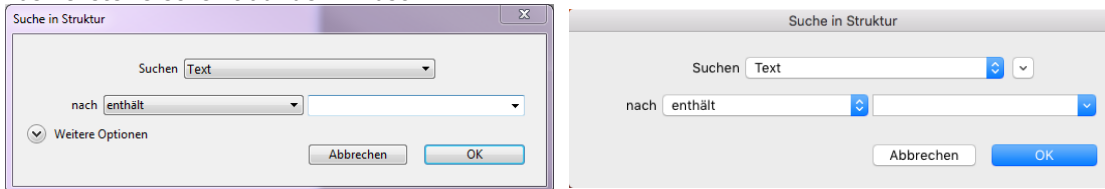
Die Suchkriterien definieren Sie im Suchdialogfenster. Sie können es auf zwei Arten aufrufen:

1. Klicken Sie in der Werkzeugleiste auf die Lupe 

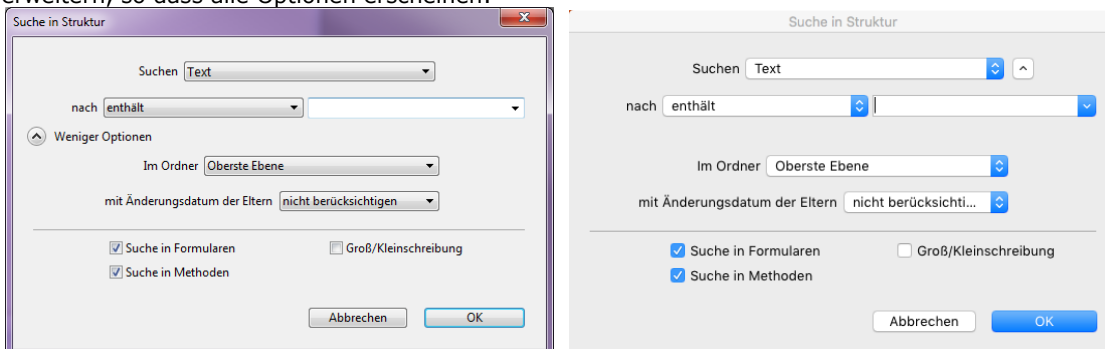
ODER

Wählen Sie im Menü **Bearbeiten** den Befehl **Suche in Struktur**.

Das Fenster erscheint auf dem Bildschirm:

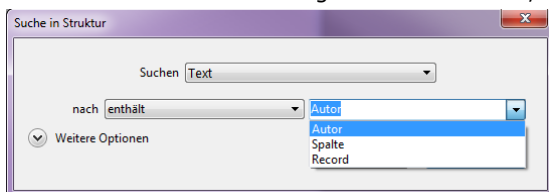


Die Bereiche im Fenster variieren je nach den gewählten Optionen in den DropDown-Menüs. Sie können das Fenster erweitern, so dass alle Optionen erscheinen:



2. Richten Sie Ihre Suche mit den verschiedenen Optionen im Bereich "Definition" ein und geben Sie bei Bedarf eine zu suchende Zeichenkette ein.
Die Beschreibung dazu folgt in den nächsten Abschnitten.
3. Markieren Sie bei Bedarf eine Suchoption, um die Suche einzugrenzen.
Die Beschreibung dazu folgt im Abschnitt "Suchoptionen".
4. Klicken Sie auf die Schaltfläche **OK** oder drücken Sie die **Eingabetaste**.
Die Suche läuft. Ist sie zu Ende, erscheint ein Fenster mit dem Suchergebnis. Es listet die gefundenen Objekte auf (siehe Abschnitt **Suchergebnis**).
Hinweis: Sie können eine laufende Suche jederzeit durch Drücken der **Esc-Taste** unterbrechen.

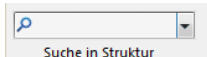
Haben Sie eine Suche ausgeführt, wird der im Suchbereich eingegebene Wert gespeichert. Er lässt sich dann, wie alle anderen während der Sitzung definierten Werte, in der Combobox auswählen:



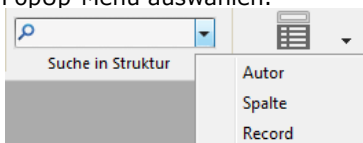
Werkzeugleiste verwenden

Sie können Suchläufe auch über die Werkzeugleiste der Designumgebung ausführen.

- Wurde während der Sitzung noch keine Suche ausgeführt, können Sie die gewünschte Zeichenkette im Eingabebereich eingeben und dann die **Zeilenschaltung** oder **Eingabetaste** drücken. Es wird automatisch eine Suche vom Typ "Text enthält" mit den Standardoptionen ausgeführt.



- Wollen Sie eine Suche wiederholen, die während der Sitzung schon einmal ausgeführt wurde, können Sie diese im PopUp-Menü auswählen.



Wurde die Suche über das Suchfenster ausgeführt, wird sie mit den dort gesetzten Optionen durchgeführt.

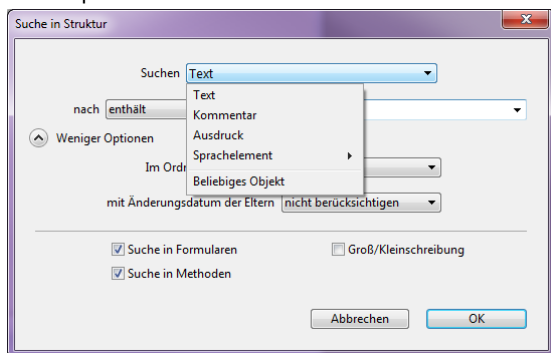
Die Suche startet unmittelbar. Ist sie zu Ende, erscheint ein Ergebnisfenster, das alle Objekte mit der angegebenen Zeichenkette enthält.

Suche definieren

Suchläufe in der Designumgebung können auf einem oder mehreren Suchkriterien basieren.

Suchen

Im DropDown-Menü Suche können Sie den Elementtyp angeben. Es gibt folgende Möglichkeiten:



- **Text:** Damit sucht 4D nach einer bestimmten Zeichenkette in der Designumgebung.. Die Suche wird im Volltextmodus ausgeführt ohne Berücksichtigung des Kontexts. Sie können zum Beispiel nach dem Text "WARNUNG("Fehlernummer:"+" oder "Schaltfläche27" suchen. In diesem Modus können Sie nicht das Joker-Zeichen verwenden, denn "@" gilt als Standardzeichen.
- **Kommentar:** Dieser Suchtyp ist ähnlich wie Text, er beschränkt sich jedoch auf den Inhalt von Kommentaren, d.h. Zeilen die mit // beginnen. So können Sie z.B. nach allen Kommentaren suchen, die den String "testen" enthalten.

Hinweis: Das Suchergebnis dieser Suchtypen hängt davon ab, welches Suchkriterium gewählt wurde (siehe Abschnitt **Suchkriterium**).

- **Ausdruck:** Damit suchen Sie nach allen gültigen 4D Ausdrücken; die Suche wird mit dem Kriterium "enthält" durchgeführt. Dabei ist die Gültigkeit ein wichtiges Kriterium, da 4D in der Lage sein muss, den Ausdruck zu prüfen, um danach zu suchen. So wird beispielsweise die Suche nach "[clients" nicht ausgeführt (ungültiger Ausdruck), jedoch nach [clients].
 - Diese Option eignet sich insbesondere für Suchen nach zugewiesenen Werten und Vergleichen. Zum Beispiel:
Suche nach "meinevar:=" (Zuweisung)
Suche nach "meinevar=" (Vergleiche)
- **Sprachelement:** Damit können Sie nach einem bestimmten Element der Programmiersprache über seinen Namen suchen. 4D kann zwischen folgenden Elementen unterscheiden:
 - **Projektmethode:** Name der Projektmethode, z.B. "M_Hinzu". Diese Suche entspricht zusammen mit dem Kriterium **ist exakt** dem Kontextmenü **Suche Referenzen** des Methodeneditors (siehe Abschnitt).
 - **Formular:** Formularname, z.B. "Eingabe". Der Befehl sucht in Projekt- und Tabellenformularen
 - **Feld oder Tabelle:** Name einer Tabelle bzw. eines Feldes, z.B. "Kunden".
 - **Variable:** Variablenname, z.B. "\$meinevar".
 - **4D Konstante:** Konstantenname, z.B. "Is Picture"
 - **Text in Anführungszeichen:** Fester Text, d.h. jeder Wert, der im Code-Editor in Anführungszeichen steht oder in Textbereiche des Formulareditors eingefügt wird (statischer Text oder Gruppenboxen). So ist beispielsweise die Suche nach "Martin" erfolgreich bei einem Code mit folgender Zeile:
QUERY ([Kunden];[Kunden]Name="Martin")
 - **4D Befehl:** 4D Befehlsname, z.B. "Alert".
 - **Plug-In Befehl:** Befehl eines Plug-In, das in der Anwendung installiert ist, z.B. "WR Find".
- **Beliebiges Objekt:** Damit können Sie eine Suche nach allen Objekten der Designumgebung ausführen. In diesem Fall ist nur das DropDown-Menü Änderungsdatum verfügbar. Ein typisches Beispiel ist die Suche nach "allen Objekten, die heute geändert wurden".

Suchkriterium

Im 2. DropDown-Menü (es lautet je nach Suchtyp **nach**, **das ist** oder **dessen Name**) definieren Sie, auf welche Art nach dem eingegebenen Wert gesucht wird. Es variiert je nach dem Element, das im 1. DropDown-Menü eingetragen wurde.

- **Suchoptionen für Text oder Kommentar:**
 - **enthält:** Sucht in allen Texten der Designumgebung nach dem angegebenen String. Die Suche nach "var" findet "meinevar", "Variable1" oder "eineVariable".
 - **enthält ganzes Wort:** Sucht in allen Texten der Designumgebung nach dem String als ganzes Wort. Die Suche nach "var" findet nur, was exakt zutrifft. Sie findet z.B. nicht "meinevar", jedoch "var:=10" oder "ID+var", da die Symbole : oder + Worttrenner sind.
 - **beginnt mit / endet mit:** Sucht nach dem String am Anfang oder Ende des Wortes (Suche nach Text) bzw. am Anfang oder Ende der Kommentarzeile (Suche nach Kommentar). Die Suche "Text endet mit" findet für "var" auch den String "meinevar".
- **Suchoptionen für Programmiersprache Element:** Dieses Menü bietet Standardoptionen (**ist exakt**, **enthält**, **startet mit**, **endet mit**). Beachten Sie, dass Sie den Joker (@) zum Suchen mit **ist exakt** verwenden können (gibt alle Objekte des angegebenen Typs zurück).

Ordner

Das DropDown-Menü "Im Ordner" beschränkt die Suche auf einen bestimmten Ordner. Standardmäßig ist die Option "Oberste Ebene" definiert, d.h. die Suche erfolgt in allen Ordnern.

Hinweis: Ordner für Objekte werden im Explorer auf der **Seite Home** angelegt.

Änderungsdatum der Eltern

Dieses Menü grenzt die Suche über das Erstellungs-/Änderungsdatum der Eltern ein (z.B. die Methode, welche die gesuchte Zeichenkette enthält). Es enthält die standardmäßigen Kriterien für Datum (**ist**, **ist vor**, **ist nach**, **ist nicht**) sowie verschiedene Optionen, über die Sie rasch einen bestimmten Suchzeitraum definieren können:

- **ist heute**: Die Suche beginnt ab Mitternacht (00:00 h) des aktuellen Tages.
- **ist seit gestern**: Die Suche enthält den aktuellen und den vorigen Tag.
- **ist diese Woche**: Die Suche beginnt am Montag der aktuellen Woche.
- **ist dieser Monat**: Die Suche beginnt am 1. Tag des aktuellen Monats.

Suchoptionen

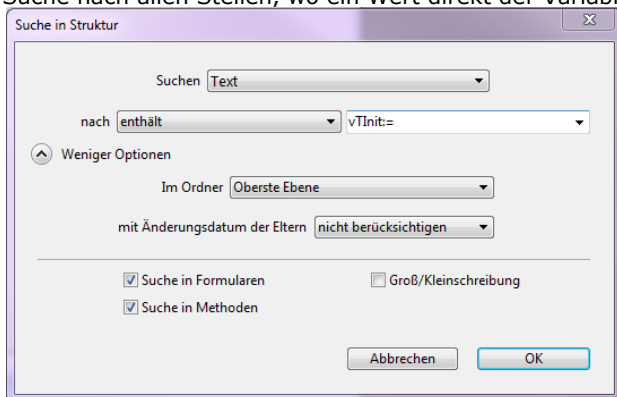
Über die folgenden Optionen können Sie Ihre Suchläufe beschleunigen:

- **Suche in Formularen**: Ist diese Option nicht markiert, läuft die Suche in der Datenbank mit Ausnahme von Formularen und Formularnamen.
- **Suche in Methoden**: Ist diese Option nicht markiert, läuft die Suche in der Datenbank mit Ausnahme von Methoden und Methodennamen.
- **Groß/Kleinschreibung**: Ist diese Option markiert, berücksichtigt die Suche Klein- und Großschreibung in den angegebenen Zeichenketten. Geben Sie z.B. "MyVar" an, findet 4D nicht "myVar".

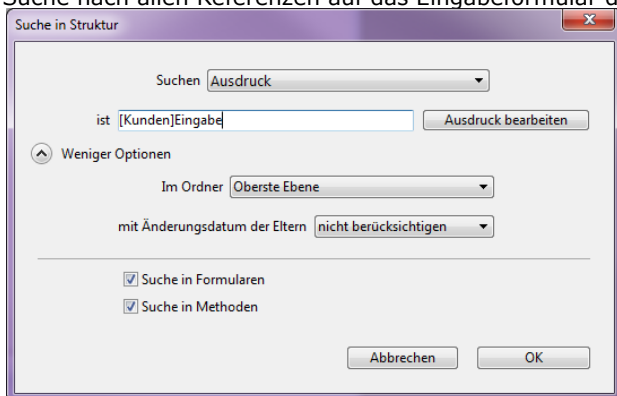
Suchbeispiele

Eine effiziente Suche ergibt sich aus einer klugen Kombination der verschiedenen Suchkriterien. Um zu zeigen, wie Suchläufe in 4D arbeiten, hier ein paar typische Beispiele mit der dazugehörigen Konfiguration.

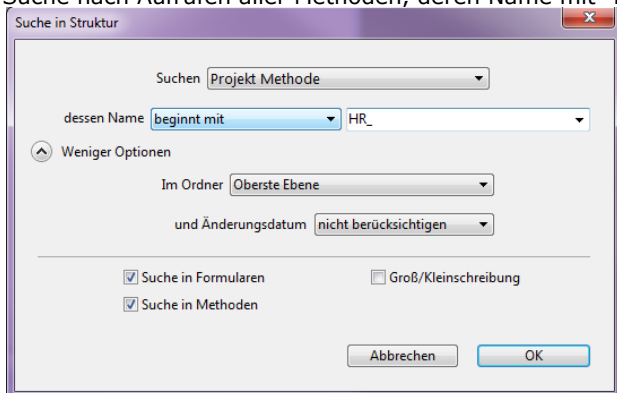
- Suche nach allen Stellen, wo ein Wert direkt der Variablen vTInit zugewiesen wurde:



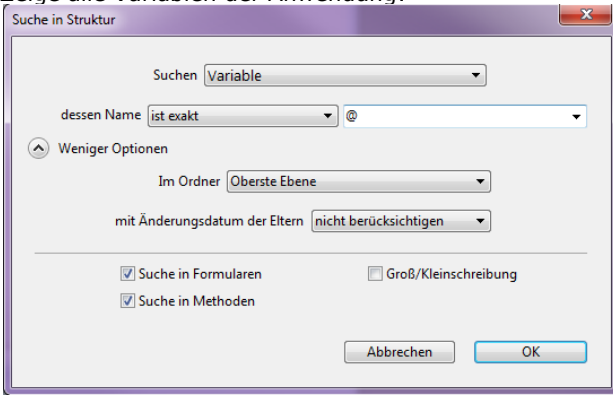
- Suche nach allen Referenzen auf das Eingabeformular der Tabelle [Kunden]:



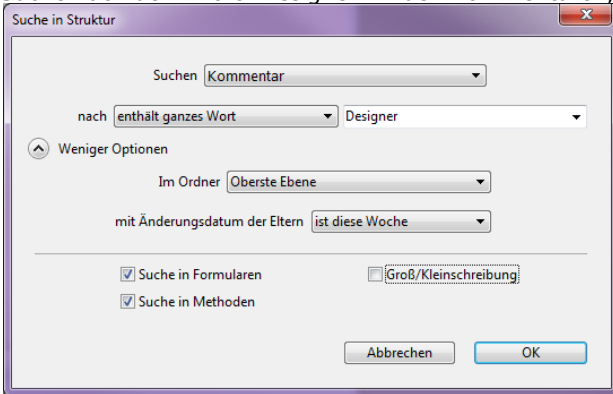
- Suche nach Aufrufen aller Methoden, deren Name mit "HR_" beginnt:



- Zeige alle Variablen der Anwendung:



- Suche nach dem Wort "Designer" in den Kommentaren, die diese Woche geschrieben wurden:

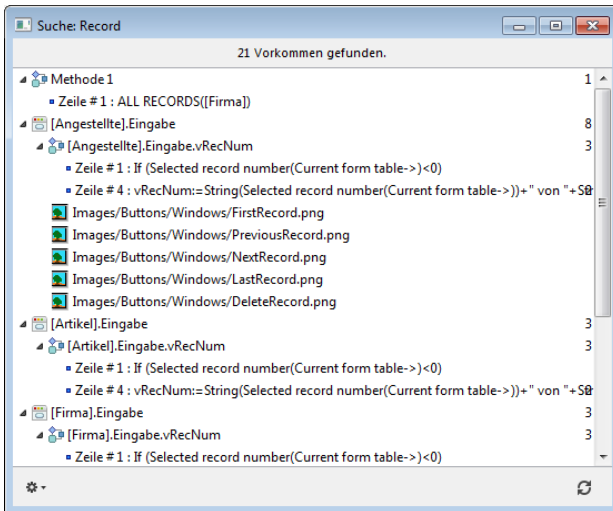


Suchergebnis

Das Fenster mit dem Suchergebnis zeigt alle gefundenen Objekte, die zu den verschiedenen Suchkriterien passen:

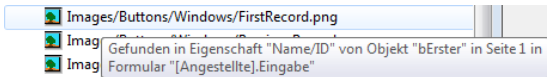
- Standardsuche (siehe Abschnitt **Suche ausführen**)
- Nach nicht-verwendeten Elementen suchen (siehe Abschnitt **Nach nicht verwendeten Elementen suchen**)
- Nach Aufrufern suchen (siehe Abschnitt **Nach Aufrufen suchen**)
- Nach Abhängigkeiten suchen (siehe Abschnitt **Abhängigkeiten suchen**) (ab 4D v13)
- Projektmethoden und Variablen umbenennen (siehe Abschnitt **Umbenennen**)

Es zeigt die Ergebnisse als hierarchische Liste, die nach Objektart geordnet ist. Sie können alle hierarchischen Einträge über entsprechende Befehle des Kontextmenüs der Seite bzw. des Menüs **Optionen** am unteren Rand auf- und zuklappen.

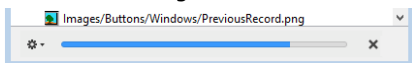


Durch Doppelklick auf eine Zeile können Sie das Objekt in seinem Editor ansehen. Sie können auch mehrere Suchen durchführen. Jede Suche öffnet ein eigenes Fenster, wobei die vorigen Fenster mit Suchergebnissen offen bleiben.

Wurden mehrere Ergebnisse für ein Objekt in einer Methode gefunden, zeigt dies die Nummer neben dem Objektnamen an. Jede Zeile kann einen Tipp mit zusätzlichen Informationen anzeigen, z.B. die Eigenschaft des Elements, das zum Suchergebnis passt oder die Nummer der Formularseite, wo das Element vorkommt:



Bei einer umfangreichen Suche erscheint ein Ablaufbalken mit einer Schaltfläche zum Abbrechen der Operation:



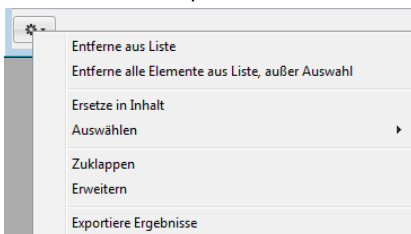
Hinweis: Die Schaltfläche **X** stoppt die laufende Suche, entfernt aber nicht die bereits gefundenen Ergebnisse.

Ist eine Suche abgeschlossen, können Sie diese über die Schaltfläche **↺** am unteren Rand mit denselben Kriterien und Optionen wiederholen.

Sie können auf die Funktion **Ersetze in Inhalt** auch im Menü **Optionen** des Fensters mit dem Suchergebnis zugreifen.

Menü Optionen

Über das Menü Optionen können Sie verschiedene Aktionen durchführen:



Hinweis: Sie können die meisten Aktionen auch über das Kontextmenü des Fensters auswählen.

- **Entferne aus Liste:** Entfernt die gewählten Einträge aus dem Ergebnisfenster. Auf diese Weise können Sie z.B. nur Einträge beibehalten, die ersetzt oder per Drag-and-Drop zwischen Programmen bewegt werden sollen.
- **Entferne alle Elemente aus Liste, außer Auswahl:** Entfernt alles aus dem Ergebnisfenster bis auf die gewählten Einträge.
- **Ersetze in Inhalt:** Ersetzt eine Zeichenkette innerhalb der gewählten Einträge. Weitere Informationen dazu finden Sie im Abschnitt **Ersetze in Inhalt**.

- **Auswählen >**: Wählt unter allen gefundenen Einträgen in **Suchergebnis** einen Typ aus. Das hierarchische Untermenü bietet auch Befehle, um alle Einträge auf einmal aus- (**Alle**) bzw. abzuwählen (**Nichts**).
- **Zuklappen/Erweitern**: Klappt alle hierarchischen Einträge der Ergebnisliste auf oder zu.
- **Exportiere Ergebnisse**: Exportiert Informationen über Objekte, die im Ergebnisfenster erscheinen. Für jeden Eintrag werden folgende Angaben, durch Tabs getrennt, in eine Textdatei exportiert:
 - Typ (Methode, Formularobjekt, Tabellenformular, Trigger)
 - Pfad des Objekts
 - Eigenschaft (falls zutreffend): Zeigt die Eigenschaft des Objekts, das zum Suchkriterium passt. Zum Beispiel werden im gleichen Formular ein String in einem Variablennamen gefunden (Eigenschaft Variable) und ein Objektname (Eigenschaft Name). Dieses Feld bleibt leer, wenn das passende Element das Objekt selbst ist.
 - Inhalt (falls zutreffend): Zeigt den Inhalt, der zum aktuellen Suchkriterium passt; z.B. die Code-Zeile mit dem angeforderten String.
 - Zeilennummer (für Methoden) oder Seitennummer (für Formularobjekte)

Hinweis: Diese Textdatei lässt sich dann in eine Tabellenkalkulation, z.B. Excel importieren.

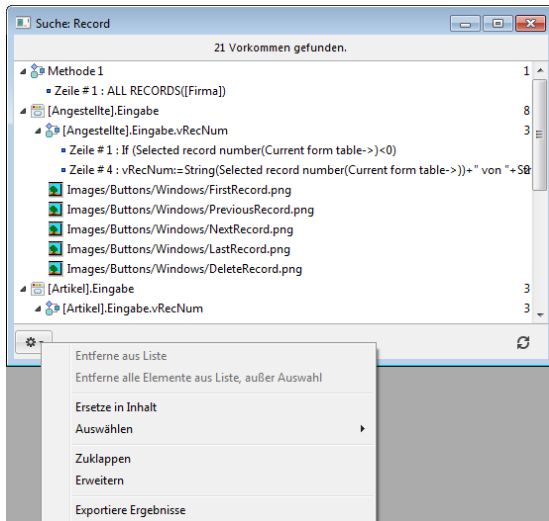
Drag and Drop verwenden

Sie können Einträge per Drag-and-Drop aus dem Fenster **Suchergebnis** entnehmen, um Objekte in der Designumgebung zwischen zwei Anwendungen zu bewegen. Weitere Informationen dazu, insbesondere "indissociable Objekte", finden Sie im Kapitel **Objekte bewegen**.

In Methoden zugewiesene Tabellen werden jetzt standardmäßig mit den Methoden zusammen bewegt. Um Methoden ohne Tabellen zu bewegen, führen Sie Drag-and-Drop mit gedrückter **Shift**-Taste aus.

Ersetze in Inhalt

Mit der Operation "Ersetze in Inhalt" können Sie in den gelisteten Objekten (siehe Abschnitt [Suchergebnis](#)) eine Zeichenkette durch eine andere ersetzen. Wählen Sie dazu im Menü **Optionen** am unteren Rand den Befehl **Ersetze in Inhalt**:

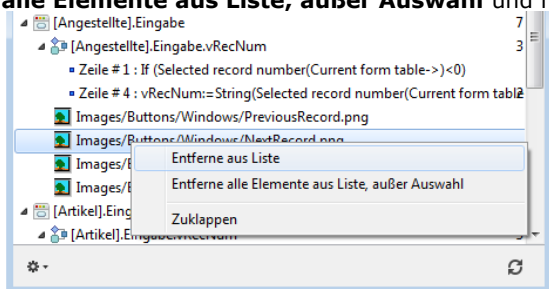


Wählen Sie diesen Befehl, erscheint ein Dialogfenster, wo Sie die Zeichenkette eingeben, die alle Vorkommen der ersten Suche ersetzen:



Operationen zum Ersetzen arbeiten wie folgt:

- Ersetzen wird immer in allen in der Liste gefundenen Einträgen ausgeführt und nicht nur für eine Auswahl. Diese Operation lässt sich auch eingrenzen. Wählen Sie dazu im Kontextmenü den Befehl **Entferne aus Liste** oder **Entferne alle Elemente aus Liste, außer Auswahl** und reduzieren Sie den Inhalt:



Hinweis: Diese Befehle sind auch im **Menü Optionen** verfügbar.

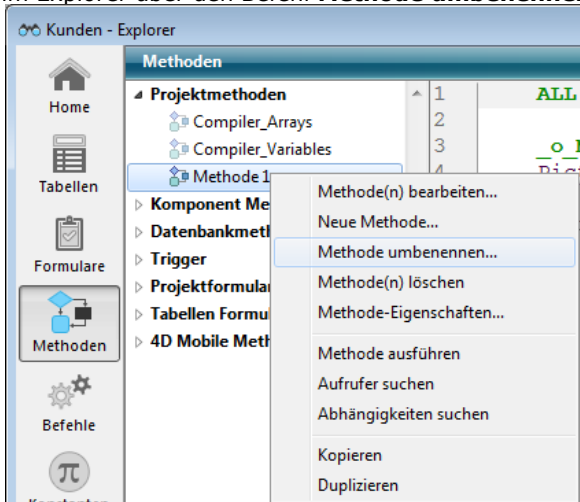
- Nur die in der Liste angezeigten Vorkommen werden ersetzt und nur, nachdem das erste Suchkriterium auf die Fälle überprüft wurde, bei denen Objekte zwischen der auslösenden Suche und den Operationen zum Ersetzen geändert wurden.
- Ersetzen wird ausgeführt in:
 - Inhalt von Methoden
 - Eigenschaften von Formularobjekten
 - Inhalt von Hilfemeldungen
 - Inhalt von Eingabefiltern
 - Inhalt von Menüeinträgen (Text von Einträgen und Aufrufe von Methoden)
 - Inhalt von Auswahllisten
 - Inhalt von Kommentaren zu Methoden, Formularen, Tabellen und Feldern im Explorer
- 4D prüft für jedes geänderte Objekt, ob es bereits von einem anderen Rechner oder in einem anderen Fenster geladen ist. Bei einem Konflikt erscheint ein Standarddialog, der angibt, dass das Objekt gesperrt ist. Sie können das Objekt schließen und dann das Ersetzen erneut versuchen oder abbrechen. Die Operation zum Ersetzen fährt dann mit den weiteren Objekten in der Liste fort.

- Wurde eine von der Operation "Ersetze in Inhalt" betroffene Methode bzw. ein Formular bereits vom gleichen 4D Programm bearbeitet, werden sie direkt im offenen Editor geändert. Es erscheint keine Warnung. Auf diese Weise geänderte Formulare und Methoden werden nicht automatisch gesichert: Sie müssen zum Bestätigen explizit die Befehle **Sichern** oder **Alle Sichern** einsetzen.
- Wurde ein Eintrag in der Liste ersetzt, erscheint er in Kursivschrift. Die Anzahl der Ersetzungen in Echtzeit erscheint am unteren Rand des Fensters.
- Objekte werden nicht durch die Operation "Ersetze in Inhalt" umbenannt, mit Ausnahme von Formularobjekten. So ist es möglich, dass u.U. bestimmte Einträge in der Liste von der Operation Ersetzen nicht betroffen sind. Das kann passieren, wenn nur der Name des Eintrags dem auslösenden Suchkriterium entspricht. Folglich werden dann nicht alle Einträge in der Liste in Kursivschrift angezeigt und die Anzahl der Ersetzungen kann geringer sein als die Anzahl der Vorkommen, die von der auslösenden Suche gefunden wurden.

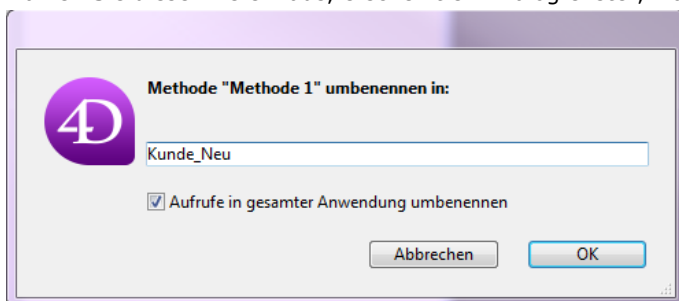
Umbenennen

4D bietet Umbenennen mit Auswirkung auf die gesamte Datenbank für Projektmethoden und Variablen. Das lässt sich folgendermaßen ausführen:

- im Methodeneditor über den Befehl **Umbenennen** des Kontextmenüs für Projektmethoden und Variablen
- im Explorer über den Befehl **Methode umbenennen** im Kontextmenü für Projektmethoden



Wählen Sie diesen Befehl aus, erscheint ein Dialogfenster, wo Sie den neuen Namen eingeben können:




Der neue Name muss die Namensregeln berücksichtigen; andernfalls erhalten Sie beim Bestätigen des Dialogfensters eine Warnung. Sie können z.B. eine Methode nicht in einen Befehlsnamen umbenennen, wie z.B. "Alert".

Der untere Teil des Dialogfensters variiert je nach Art des umzubenennenden Objekts (Projektmethode oder Variable):

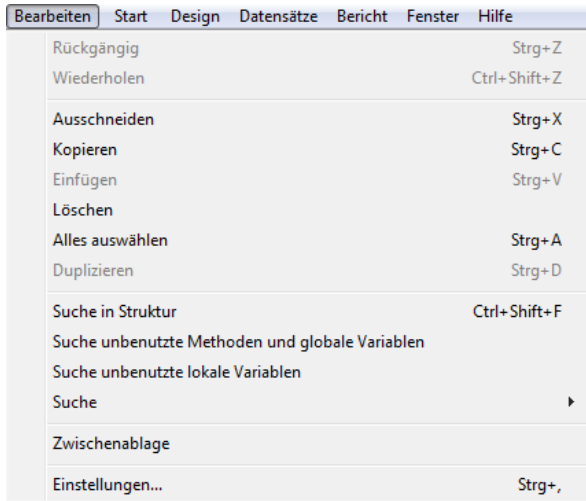
- **Projektmethode:** Die Option **Aufrufe in gesamter Anwendung umbenennen** dehnt die Umbenennung der Methode auf alle Objekte der Anwendung, die darauf verweisen, aus. Sie können diese Option deaktivieren, wenn Sie die Methode z.B. nur im Explorer selbst umbenennen wollen.
- **Prozess- und Interprozessvariable:** Die Option **Variable in gesamter Anwendung umbenennen** dehnt die Umbenennung der Variable auf alle Objekte der Anwendung, die darauf verweisen, aus. Deaktivieren Sie diese Option, wird die Variable nur in der aktuellen Methode umbenannt.
- **Lokale Variable:** Hier gibt es keine weitere Option; die Variable wird nur in der aktuellen Methode umbenannt.

Suche wiederholen

Über die Schaltfläche  am unteren Rand können Sie die Suche mit denselben Kriterien und Optionen wiederholen. Das ist z.B. hilfreich, um sicherzustellen, dass alle gewünschten Ersetzungen ausgeführt wurden.

Nach nicht verwendeten Elementen suchen

Es gibt zwei Suchbefehle, um Variablen und Methoden zu finden, die in Ihrem Code nicht verwendet werden. Sie können diese dann entfernen, um Speicherplatz wieder freizugeben. Diese Befehle liegen im Designmodus im Menü **Bearbeiten**:



Unbenutzte Methoden und globale Variablen suchen

Der Befehl **Suche unbenutzte Methoden und globale Variablen** sucht nach Projektmethoden sowie "globalen" Variablen (Prozess- und Interprozessvariablen), die deklariert sind und nicht verwendet werden. Die Suchergebnisse erscheinen in einem Standard Ergebnisfenster **Suchergebnis**.

Eine Projektmethode gilt als unbenutzt, wenn:

- sie nicht im Papierkorb liegt
- nirgendwo im 4D Code aufgerufen wird
- nicht durch einen Menübefehl aufgerufen wird
- nicht als String Konstante im 4D Code aufgerufen wird (4D erkennt einen Methodennamen in einem String auch, wenn danach Parameter in Klammern folgen.)

Eine Prozess- oder Interprozessvariable gilt als unbenutzt, wenn:

- sie im 4D Code über einen Befehl vom Typ C_XXX oder ARRAY XXX deklariert ist,
- nirgendwo sonst im 4D Code aufgerufen wird
- in keinem Formularobjekt verwendet wird

Beachten Sie, dass die Funktion bestimmte Verwendungsfälle nicht herausfinden kann - ein Element, das als unbenutzt eingestuft ist, kann trotzdem in Verwendung sein. Sehen Sie hierzu nachfolgendes Beispiel:

```
v:="method"  
EXECUTE FORMULA("my"+v+String(42))
```

Dieser Code erstellt einen Methodennamen. Die Projektmethode **mymethod42** gilt als unbenutzt, wird jedoch aufgerufen. Aus diesem Grund empfiehlt es sich, vor dem Entfernen von Elementen, die als unbenutzt deklariert sind, zu prüfen, ob sie ohne Bezug sind.




Unbenutzte lokale Variablen suchen

Der Befehl **Suche unbenutzte lokale Variablen** sucht nach lokalen Variablen, die deklariert sind und nicht verwendet werden. Die Suchergebnisse erscheinen in einem Standard Ergebnisfenster **Suchergebnis**.

Eine lokale Variable gilt als unbenutzt, wenn:

- sie im 4D Code über einen Befehl vom Typ C_XXX oder ARRAY XXX deklariert ist
- nirgendwo sonst im 4D Code verwendet wird

Objekte bewegen

-  Überblick
-  Dialogfenster Bewegen
-  Eigenschaften zum Bewegen

In 4D können Sie in der Designumgebung Objekte zwischen zwei Anwendungen in der Designumgebung bewegen. Sie können also die in einer Datenbank angelegten Tabellen, Formulare, Methoden, etc. in eine andere Datenbank kopieren. Das vereinfacht und beschleunigt die Datenbank-Entwicklung ganz entscheidend.

Das Bewegen ist nicht auf einzelne Objekte begrenzt, es gilt auch für Objekte, die an ein anderes Objekt gekoppelt sind, also untergeordnete Objekte. Auf diese Weise können Sie ganze Funktionalitäten bewegen. Haben Sie z.B. einen eigenen Suchdialog erstellt, können Sie sowohl das verwendete Formular, als auch alle Methoden, Bilder und andere dazugehörigen Objekte mitverschieben. Sie können das Formular in eine andere Datenbank kopieren oder in eine Objektbibliothek, welche die in Ihren Datenbanken verwendeten Funktionalitäten gruppiert.

Bestimmte Objekte sind untrennbar. Sie lassen sich nur zusammen mit dem Hauptobjekt verschieben. Die Liste untrennbarer Objekte wird unten aufgeführt.

Bewegbare Objekte

Objekte lassen sich aus der Toolbox, dem **Explorer** und dem Formulareditor bewegen. Sie können darüberhinaus auch Ergebnisse einer Suche in der Designumgebung verschieben (siehe Abschnitt **Suchergebnis**).

Um die Struktur zu wahren, werden beim Kopieren bestimmter Objekte auch die damit verbundenen Objekte mitkopiert. Kopieren Sie z.B. ein Formular, werden alle dazugehörigen Formular- und/oder Objektmethoden mitkopiert. Diese untrennbaren Objekte lassen sich nicht eigenständig kopieren.

Folgende Objekte haben untrennbare Objekte und lassen sich nur zusammen bewegen:

Bewegbare Objekte

---- Toolbox ----

Listen
Stilvorlagen
Formate/Filter
Bilder aus der Bibliothek
Hilfetipps

Untrennbare Objekte

-
-
-
-
-

---- Explorer ----

Projektformulare
Tabellenformulare
Projektmethoden
Ordner/Unterordner
Tabellen

Formularmethoden
Formularmethoden
-
-
Felder, Trigger

---- Formulareditor ----

Alle Formularobjekte, wie Schaltflächen, Variablen, o.ä., beim Bewegen eines Formulars werden alle darin enthaltenen Objekte mitverschoben.

Objektmethoden

Funktionsweise

Sie können Objekte entweder per Drag&Drop oder Kopieren/Einfügen bewegen.

Wollen Sie Objekte zwischen zwei Datenbanken per Drag&Drop bewegen, müssen Sie Ihre 4D Version duplizieren.

Beim Bewegen zwischen Datenbanken können Sie die Objekte wieder in dieselbe Umgebung wie in der Ausgangsdatenbank legen, z.B. Toolbox, Explorer, oder in andere Bereiche der Anwendung. 4D führt je nach Kontext die geeignete Aktion aus. Ziehen Sie z.B. ein Formular in den Methodeneditor, wird der Name des Formulars in die Methode eingefügt.

Gibt es in der Zieldatenbank bereits ein Objekt mit demselben Namen und vom selben Typ, wird es standardmäßig durch das bewegte Objekt ersetzt. In diesem Fall erscheint die Meldung, dass die Objekte ersetzt werden. Sie können dann die Aktion ausführen oder abbrechen.

Es gelten folgende Regeln:

- **Ansichten und Ebenen:** Bewegte Formularobjekte behalten dieselben Platzierungsmerkmale wie im Editor, insbesondere ihre Position in den Ansichten und verschiedenen Ebenen des Formulars.
- **Vererbte Formulare:** Diese Formulare werden nicht mit den Quellformularen bewegt, die Referenz darauf wird jedoch beibehalten. Da sie als abhängige Objekte betrachtet werden, können Sie bei der Ausführung der Operation ein anderes vorhandenes Formular als vererbtes Formular verwenden. Weitere Informationen dazu finden Sie im nächsten Abschnitt.
- **Zugriffsrechte:** Bewegte Formulare und Projektmethoden behalten nicht ihre ursprünglichen Zugriffsrechte. Sie erhalten automatisch den Standardwert "Alle Gruppen".
- **Ordner:** Bewegen Sie einen Ordner aus der Hauptseite des Explorers, wird die Operation für den gesamten Inhalt ausgeführt, das sind Tabellen, Formulare und Projektmethoden. Das kann eine beträchtliche Menge an Daten sein. Auf dem Bildschirm erscheint eine Meldung, dass alle diese Elemente übertragen werden. Sie können dann die Aktion ausführen oder abbrechen.

Hinweis: Es ist nicht möglich, Objekte aus dem Papierkorb des Explorers zu verschieben.

Abhängige Objekte

- Ein Tabellen- oder Projektformular kann auf mehrere andere Objekte verweisen, z.B. Listen, Bilder. Diese Objekte werden abhängige Objekte genannt. Sie können selbst bestimmen, ob Sie alle, ein paar oder keine der abhängigen Objekte mitübertragen wollen. Das können Sie in den Voreinstellungen zum Bewegen und im **Dialogfenster Bewegen** steuern. Sie können für jede Objektart verschiedene Standardoptionen festlegen. Weitere Informationen dazu finden Sie im Abschnitt **WEB SET HOME PAGE**.
- Ab 4D v13 lassen sich in Methoden zugewiesene Tabellen standardmäßig zusammen mit den Methoden bewegen. Um die Methode ohne Tabellen zu bewegen, ziehen Sie die Methode bei gedrückter **Shift**-Taste aus dem Fenster Suchergebnis bzw. Explorer.

Dialogfenster Bewegen

Beim Bewegen einer Objektauswahl (per Drag&Drop oder Kopieren/Einsetzen) zwischen zwei 4D Datenbanken oder einer Datenbank und einer Objektbibliothek kann ein Dialogfenster erscheinen, das alle bewegten Objekte und die damit verbundenen Aktionen in der Zieldatenbank auflistet. Es hat den Namen "Kopieren-Dialog" und erscheint, wenn einer der folgenden Fälle zutrifft:

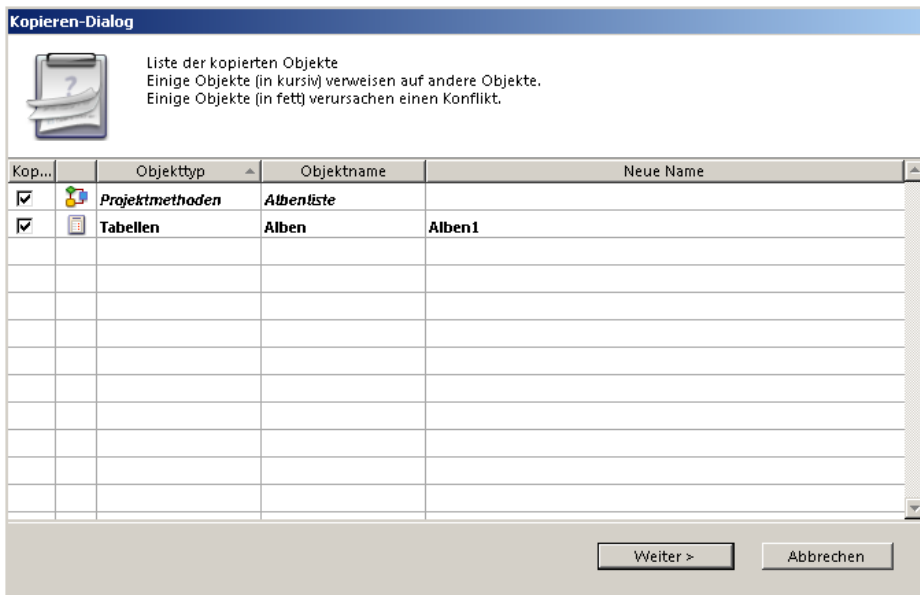
- In den Datenbank-Eigenschaften ist auf der Seite Verschieben die Option "Immer" markiert (siehe [WEB SET HOME PAGE](#)).
- Mindestens eins der bewegten Objekte verursacht einen Namenskonflikt mit einem Objekt in der Zieldatenbank.
- Mindestens einer Art von bewegtem abhängigen Objekt ist die Standardoption **Anderes Objekt verwenden** zugewiesen.

Trifft keiner dieser Fälle zu und verursacht das Bewegen von Objekten keine Konflikte, erscheint dieses Dialogfenster nicht und die Objekte werden direkt kopiert.

Dieses Dialogfenster ermöglicht je nach Kontext, die Einstellungen zum Bewegen anzusehen bzw. zu verändern. Es besteht aus zwei Seiten: **Hauptseite** und **Detailseite**. Über die Schaltflächen **Weiter>** und **<Zurück** können sie zwischen beiden hin- und herwechseln.

Hauptseite

Die Hauptseite zeigt die Liste der bewegten Objekte:



Objekte mit Namenskonflikt erscheinen in **Fettschrift**, abhängige Objekte in *Kursivschrift*. Die einzelnen Spalten zeigen für jedes Objekt Name und Typ sowie den neuen Namen in der Zieldatenbank. Diesen Namen können Sie bei Bedarf auf der Seite Details verändern. Mehr dazu im folgenden Absatz.

Das Kästchen in der Spalte "Kopieren" gibt an, ob das Objekt in die Zieldatenbank kopiert wird. Sie können es deaktivieren, um rasch Kopierprobleme zu lösen, die bei einem bestimmten Objekt auftreten. Beachten Sie, dass bei Problemen mit einem abhängigen Objekt die Referenz in der Zieldatenbank erhalten bleibt. Mit der Tastenkombination **Strg+Klick** unter Windows, **Wahltaste+Klick** auf Mac OS können Sie alle Kästchen auf einmal aktivieren bzw. deaktivieren. Sie können auch das Kontextmenü des Dialogfensters verwenden.

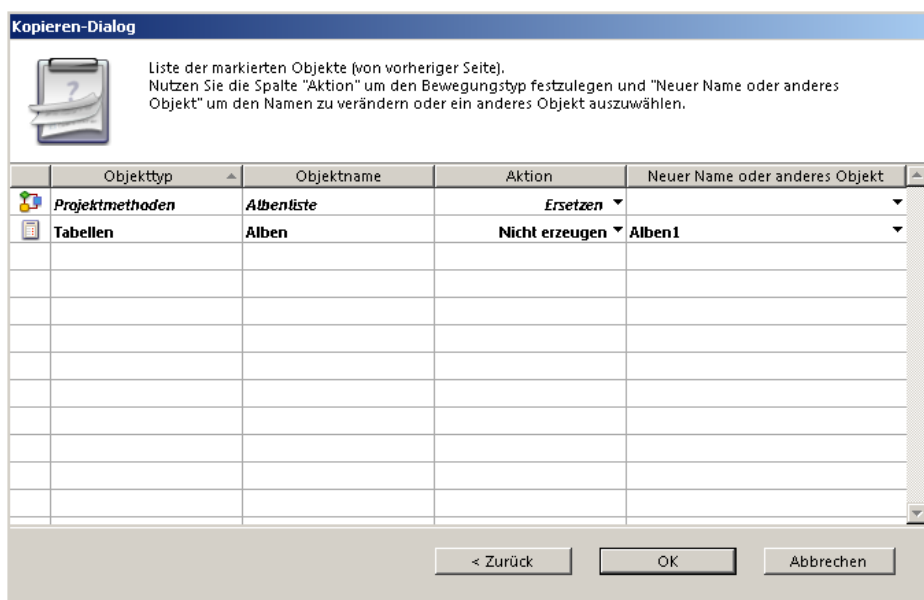
Passen die standardmäßigen Einstellungen, können Sie direkt auf die Schaltfläche **OK** klicken, um das Bewegen der Objekte auszuführen.

Hinweis: Wurde einem Objekt die Aktion **Anderes Objekt verwenden** zugewiesen, wird der Kopiervorgang erst ausgeführt, wenn Sie ein Zielobjekt in der Zieldatenbank bestimmt haben. Sie können auch das betreffende Objekt deaktivieren.

Wollen Sie bestimmte Aktionen verändern, klicken Sie auf die Schaltfläche **Weiter>**, um die Detailseite anzuzeigen. Wollen Sie den Kopiervorgang abbrechen, klicken Sie auf die Schaltfläche **Abbrechen**.

Detailseite

Auf der Detailseite erscheinen die Objekte, welche auf der Hauptseite ausgewählt sind. Hier können Sie die Einstellungen zum Bewegen verändern:



Über das PopUp-Menü "Aktion" können Sie Aktionen für Objekte verändern. Die Aktionen sind je nach gewähltem Objekttyp unterschiedlich. Sie werden im folgenden beschrieben.

Sie können mehreren Zeilen zugewiesene Aktionen auf einmal ändern. Wählen Sie dazu die entsprechenden Zeilen aus und dann in der Spalte Aktion für eine der markierten Zeilen eine neue Aktion. Die Änderung wird dann für alle Zeilen in der Auswahl ausgeführt. Passt die Aktion nicht zu einer der Zeilen, wird diese Zeile nicht geändert. Auf dem Bildschirm erscheint dann eine Meldung, die Sie darüber informiert.

Die Spalte "Neuer Name oder anderes Objekt" zeigt den Namen, den das Objekt in der Zieldatenbank erhält. Sie können diesen Namen verändern. Achten Sie jedoch darauf, dass Sie keinen Namen verwenden, der in der Zieldatenbank schon vorhanden ist, denn das führt zu einem Namenskonflikt.

Bei abhängigen Objekten können Sie in dieser Spalte auch ein anderes Objekt der Zieldatenbank bestimmen. Dazu muss die Aktion **Anderes Objekt verwenden** ausgewählt sein. Bewegen Sie z.B. ein Tabellenformular, können Sie in der Zieldatenbank eine vorhandene Tabelle dafür festlegen anstatt die Tabelle anzulegen.

Mögliche Aktionen

Es gibt folgende Aktionen:

- **Nicht erzeugen:** Das Objekt wird nicht kopiert. Bei abhängigen Objekten wird deren Referenz (Name) beibehalten, wenn in der Datenbank bereits ein Objekt mit demselben Namen existiert. Es wird dann vom Hauptobjekt verwendet. Gibt es kein gleichlautendes Objekt, wird die Referenz gelöscht.
- **Ersetzen:** Diese Option ist aktiv, wenn es in der Datenbank bereits ein Objekt gleichen Namens und Typs gibt. Das Objekt in der Zieldatenbank wird dann durch das aus der Ausgangsdatenbank ersetzt.
- **Erzeugen:** Diese Option ist aktiv, wenn es keinen Namenskonflikt gibt. Das abhängige Objekt wird dann mit seinen Eigenschaften in die Zieldatenbank kopiert.
- **Erstellen und umbenennen:** Diese Option ist aktiv, wenn es in der Datenbank bereits ein Objekt gleichen Namens und Typs gibt. Standardmäßig wird dem Objektnamen eine Nummer angehängt. Sie können dann das bewegte Objekt in der Spalte "Neuer Name oder anderes Objekt" umbenennen. Die Referenzen zum Objekt werden dann natürlich in der Zieldatenbank aktualisiert.
- **Anderes Objekt verwenden:** Diese Option ist nur für abhängige Objekte verfügbar. Damit können Sie als Referenz ein anderes Objekt verwenden, das in der Zieldatenbank bereits vorhanden ist. Die Spalte "Neuer Name oder anderes Objekt" enthält die Liste der verwendbaren Objekte.
- **Tabelle mit gleichem Namen verwenden:** Diese Option ist aktiv, wenn in der Zieldatenbank bereits eine Tabelle mit dem gleichen Namen existiert. Die Spalte "Neuer Name oder anderes Objekt" enthält die Liste der Tabellen, die Sie anstelle der abhängigen Tabelle verwenden können.

Verweisen abhängige Objekte selbst auf andere Objekte, wird die Liste gemäß Ihren Einstellungen aktualisiert.

Passen die Einstellungen, klicken Sie auf die Schaltfläche **OK**, um die Objekte zu bewegen. Klicken Sie auf die Schaltfläche **<Zurück**, um auf die Hauptseite zurückzukehren. Klicken Sie auf die Schaltfläche **Abbrechen**, um die Aktion abzubrechen.

Auf der Seite Verschieben der **PICTURE LIBRARY LIST** können Sie die Optionen zum Bewegen von Objekten der Datenbank im Designmodus definieren. 4D wendet diese Einstellungen auf Objekte an, die per Drag-and-Drop oder durch Kopieren/Einsetzen in diese Datenbank gesetzt werden, wenn diese als Zieldatenbank verwendet wird.

Standardaktion fürs Kopieren von abhängigen Objekten

Diese Optionen bestimmen das Bewegen von abhängigen Objekten. Das sind die Objekte, die mit den zu bewegendenden Formularen verknüpft sind (siehe **Überblick**). Sie können für jeden abhängigen Objekttyp eine Aktion setzen.

Diese Standardaktionen werden automatisch angewendet, wenn beim Bewegen von Objekten keine Konflikte auftreten und die Option **Nur bei Namenskonflikten** ausgewählt ist (siehe nächster Absatz). Sonst werden sie standardmäßig im *Kopieren-Dialog* aufgerufen.


















Die Optionen **Ignorieren**, **Erzeugen (bei Bedarf umbenennen)**, **Erzeugen (bei Bedarf ersetzen)** sowie die Aktionen für **Anderes Objekt verwenden** sind für jeden Objekttyp verfügbar. Andere kontextbezogene Aktionen sind verfügbar, wenn das **Dialogfenster Bewegen** angezeigt wird. Es folgt die Beschreibung dieser Optionen:

- **Ignorieren**: Ist dieser Typ zugewiesen, wird das abhängige Objekt nie in die Zieldatenbank kopiert. Im *Kopieren-Dialog* erscheint standardmäßig die Aktion **Nicht erzeugen**.
- **Erzeugen (Bei Bedarf umbenennen)**: Ist dieser Typ zugewiesen, wird das abhängige Objekt immer in die Zieldatenbank kopiert. Im *Kopieren-Dialog* erscheint standardmäßig die Aktion **Erzeugen**, wenn das Objekt in der Zieldatenbank noch nicht existiert.
Bei einem Namenskonflikt mit einem Objekt in der Zieldatenbank erhält der Name des kopierten Objekts eine Nummer, die bei Bedarf hochgezählt wird. In diesem Fall erscheint im *Kopieren-Dialog* die Aktion **Umbenennen**.
- **Erzeugen (Bei Bedarf ersetzen)**: Ist dieser Typ zugewiesen, wird das abhängige Objekt immer in die Zieldatenbank kopiert. Im *Kopieren-Dialog* erscheint standardmäßig die Aktion **Erzeugen**, wenn das Objekt noch nicht in der Zieldatenbank existiert..
Bei einem Namenskonflikt mit einem Objekt in der Zieldatenbank ersetzt das kopierte Objekt das vorhandene Objekt. In diesem Fall erscheint im *Kopieren-Dialog* standardmäßig die Aktion **Umbenennen**.
- **Anderes Objekt verwenden**: Ist diese Option markiert, erscheint der *Kopieren-Dialog* systematisch, und auch, wenn die Option "Nur bei Namenskonflikt anzeigen" gewählt ist. Beim Bewegen müssen Sie ein Objekt in der Zieldatenbank bestimmen, das anstelle des zu kopierenden abhängigen Objekts benutzt werden soll.

Hinweis: Diese Optionen gelten nur für abhängige Objekte. Für die anderen Objekte gilt beim Bewegen die Standardaktion **Erzeugen (Bei Bedarf umbenennen)**.

- **Kopieren-Dialog anzeigen**: Mit dieser Option legen Sie fest, ob der *Kopieren-Dialog* angezeigt wird.
Ist die Option **Immer** gewählt, erscheint das Fenster immer, wenn Objekte bewegt werden und ermöglicht so die Operation genauer zu steuern. Ist die Option **Nur bei Namenskonflikten** gewählt, erscheint das Dialogfenster nur, wenn beim Bewegen von Objekten Namenskonflikte mit Objekten in der Zieldatenbank auftreten.

Datenbankstruktur erstellen

-  Grundlagen der Datenbank
-  Leistungsumfang der 4D Datenbank
-  Struktureditor
-  Inspektorfenster
-  Tabellen erstellen und ändern
-  Tabelleneigenschaften
-  Primärschlüssel-Assistent
-  Datenfelder erstellen und ändern
-  4D Datenfeldtypen
-  Feldeigenschaften
-  Daten extern speichern
-  Namensregeln für Tabellen und Datenfelder
-  Indizes erstellen und ändern
-  Verknüpfungen erstellen und ändern
-  Verknüpfungsarten
-  Verknüpfungseigenschaften
-  Strukturdefinitionen exportieren und importieren

Grundlagen der Datenbank

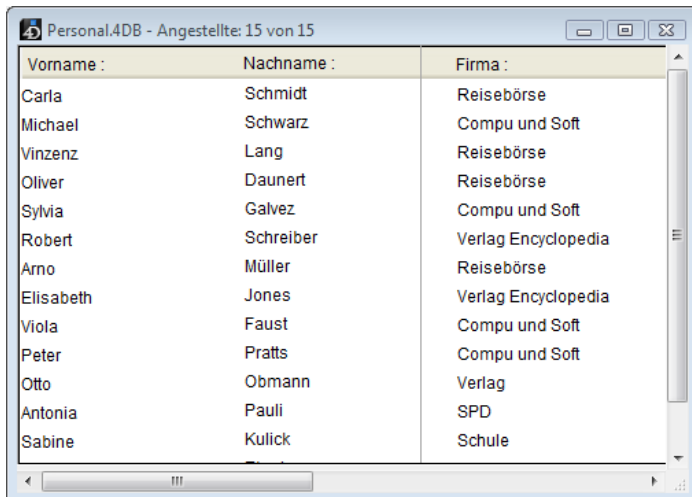
Eine Datenbank ist eine Sammlung von Informationen, die für eine effiziente Verwendung strukturiert ist. Ein Telefonverzeichnis ist ein gutes Beispiel für eine Datenbank. Gleiches gilt für ein Wörterbuch, einen Kalender oder ein Kochbuch.

Die Informationen in einer Datenbank sind in Form von Datensätzen geordnet. Jeder Datensatz enthält alle Informationen über eine Person oder Sache in der Datenbank. Jeder Datensatz in einem Telefonverzeichnis enthält z.B. den Namen einer Person, die Adresse und die Telefonnummer.

Jeder Datensatz enthält Datenfelder. In einem Datenfeld werden einzelne Teile einer Information gespeichert. In der Datenbank eines Telefonverzeichnisses enthält z.B. ein Datenfeld den Namen der Person; ein zweites die Adresse dieser Person; und ein drittes die Telefonnummer dieser Person. Jeder Datensatz enthält diese drei Datenfelder und jeder Datensatz kann in diesen Datenfeldern Informationen enthalten.

Ein Feldname kennzeichnet normalerweise die in dem Datenfeld enthaltene Information. Ein Feldname ist gewöhnlich so etwas wie Name, Adresse, oder Telefonnummer. Jedes Datenfeld hat einen Feldtypen, der die Art von Information kennzeichnet, die in das Datenfeld eingegeben werden kann: Zahlen, Daten, alphanumerische Zeichen und andere. Da jedes Datenfeld einen besonderen Typ von Daten enthält, kann man Berechnungen und andere Operationen mit den Informationen in den Datenfeldern durchführen. Man kann z.B. die Zahlen aus zwei Datenfeldern addieren. Ein Datum in einem Datenfeld kann mit dem Datum aus einem anderen Datenfeld verglichen werden. Der Vorname einer Person, gespeichert in einem Datenfeld, kann vor dem Nachnamen, gespeichert in einem anderen Datenfeld, dargestellt werden und so die erste Zeile auf einem Adressen-Etikett sein.

Alle Datensätze zusammen bilden eine **Tabelle**. Jede Datenbank kann viele Tabellen enthalten. Die folgende Abbildung zeigt, wie diese Elemente miteinander verknüpft sind



Vorname :	Nachname :	Firma :
Carla	Schmidt	Reisebörse
Michael	Schwarz	Compu und Soft
Vinzenz	Lang	Reisebörse
Oliver	Daunert	Reisebörse
Sylvia	Galvez	Compu und Soft
Robert	Schreiber	Verlag Encyclopedia
Arno	Müller	Reisebörse
Elisabeth	Jones	Verlag Encyclopedia
Viola	Faust	Compu und Soft
Peter	Pratts	Compu und Soft
Otto	Obmann	Verlag
Antonia	Pauli	SPD
Sabine	Kulick	Schule

4D kann Datensätze umordnen und Berechnungen mit den Informationen durchführen, damit Sie diese Information sinnvoll einsetzen können. 4D kann z.B. den Gesamtwert eines Datenfeldes berechnen und die Summe in einem Datensatz darstellen. Es kann für jeden Verkäufer eine Summe berechnen und eine Grafik anzeigen, in der die Verkaufszahlen verglichen werden.

Tabellen und Felder

4D kann pro Anwendung von 1 bis zu 32 767 Tabellen anlegen. So können Sie eine Struktur erstellen, die genau zu Ihren Anforderungen passt.

Strukturen mit einer Tabelle

Einige Datenbanken setzen nur eine Tabelle ein. Ein einzelne Tabelle benutzen Sie für eine Kategorie von Informationen, wie z.B. Angestellte, Firmen oder Inventar. Eine Tabelle kann so viele Datenfelder enthalten, wie Sie benötigen, max. 32 767.

Angestellte

Nachname	A
Vorname	A
Firma	A
Straße	A
PLZ	0.5
Ort	A
Gehalt	0.5
Kinder	2 ³²

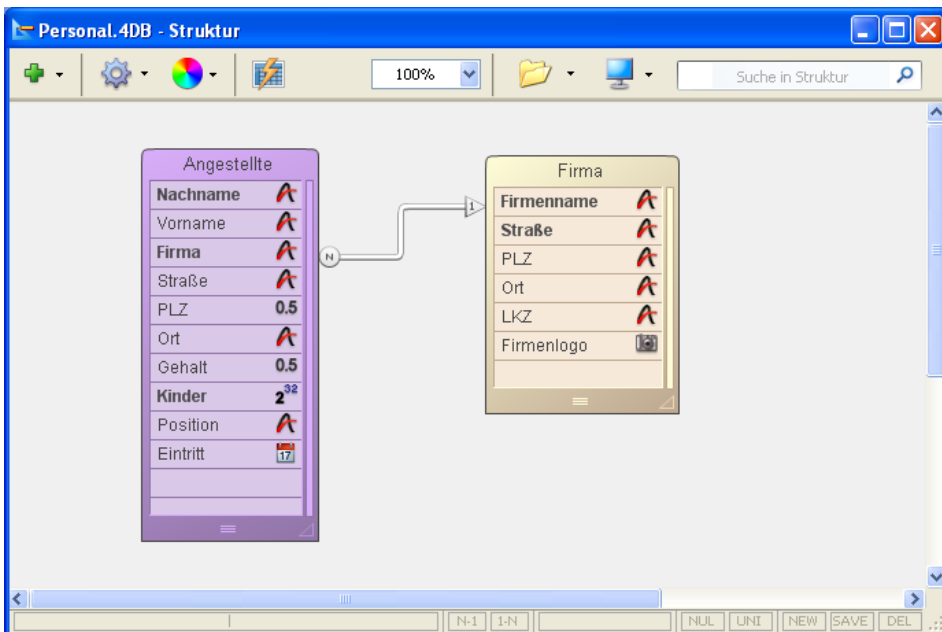
Personal.4DB - Angestellte: 15 von 15

Vorname :	Nachname :	Firma :
Carla	Schmidt	Reisebörse
Michael	Schwarz	Compu und Soft
Vinzenz	Lang	Reisebörse
Oliver	Daunert	Reisebörse
Sylvia	Galvez	Compu und Soft
Robert	Schreiber	Verlag Encyclopedia
Arno	Müller	Reisebörse
Elisabeth	Jones	Verlag Encyclopedia
Viola	Faust	Compu und Soft
Peter	Pratts	Compu und Soft
Otto	Obmann	Verlag
Antonia	Pauli	SPD
Sabine	Kulick	Schule

In dieser Abbildung benötigt der Datensatz jedes Angestellten dieselbe Art von Daten. Die Datenbank wächst mit der Anzahl der gespeicherten Angestellten.

Strukturen mit mehreren Tabellen

Oftmals kann eine Datenbank die Daten besser speichern und darauf zugreifen, wenn mehr als eine Tabelle benutzt wird. Eine gute Regel ist, dass unterschiedliche Arten von Informationen in unterschiedlichen Tabellen gespeichert werden sollten. Ein gängiges Beispiel ist eine Datenbank, in der Angestellte und Firmen erfasst sind. Die Datensätze für Angestellte und Firmen sind in verschiedenen Tabellen gespeichert. Wenn sich die Adresse einer Firma ändert, müssen Sie nur diesen einen Datensatz ändern. Sie müssen die neue Adresse nicht für jeden Angestellten eingeben, der in dieser Firma arbeitet. Bei nur einer Tabelle müssten Sie die Adresse bei jedem einzelnen Datensatz eingeben; bei zwei Tabellen geben Sie diese Information nur einmal ein. Geben Sie in den Datensatz eines Angestellten einen Firmennamen ein, kann 4D nach dem Datensatz der Firma suchen und automatisch die richtige Adresse anzeigen. Die folgende Abbildung zeigt die Struktur einer Datenbank, bei der zwei Tabellen miteinander verknüpft sind. Der Pfeil zwischen dem Firmenfeld und dem Datenfeld mit dem Firmennamen zeigt die Verknüpfung.



Die Daten für jeden Angestellten sind in der Tabelle [Angestellte] gespeichert. Die Daten jeder Firma sind in der Tabelle [Firma] abgelegt.

Hinweis: In der 4D Dokumentation erscheinen Tabellennamen in eckigen Klammern. So werden sie auch im Methodeneditor angezeigt.

4D wird als relationale Datenbank bezeichnet, d.h. es werden mehrere Tabellen benutzt, die unterschiedlich miteinander verknüpft sind. Sie können z.B. einen Datensatz für die Tabelle [Angestellte] erstellen, der in der Tabelle [Firma] sucht und automatisch die Informationen über die Firma jedes Angestellten anzeigt und druckt. Durch die Verknüpfungen zwischen den Tabellen stehen die Informationen aus jeder Tabelle für den Datensatz zur Verfügung.

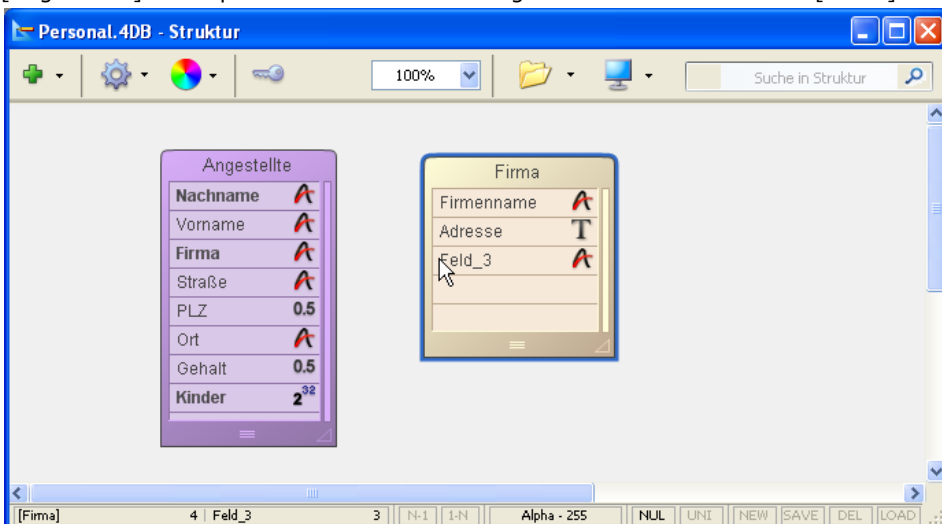
Sie können Daten direkt in verknüpfte Tabellen eingeben. In einer Datenbank für Rechnungen können Sie z.B. in einem Formular für eine Rechnung auch die Informationen der einzelnen Positionen verändern, obwohl diese in einer anderen Tabelle gespeichert sind. Sie können auch mit der Programmiersprache von 4D Daten in verknüpfte Tabellen schreiben.

Manchmal benötigen Sie eine Struktur mit mehreren Tabellen, bei der die Tabellen nicht direkt miteinander verknüpft sind. Es kann vorteilhaft sein, eine Datenbank zu haben, in der unterschiedliche Arten von Informationen gespeichert sind, wie z.B. eine Adressenliste und eine Kostentabelle.

4D erlaubt in jeder Datenbank bis zu 32 767 Tabellen. Eine Tabelle kann bis zu 32 761 Datenfelder haben. Bei Mehrfachtabellen ist virtuell jede Art von Datenbankstruktur möglich.

Tabellen verknüpfen

In der Regel erstellen Sie Strukturen, in denen Informationen auf mehrere Tabellen verteilt sind. Nehmen wir z.B. an, Sie erstellen eine Datenbank, in der Sie Angestellte und ihre Firmen erfassen. Die unten dargestellte Struktur enthält eine Tabelle [Angestellte] zum Speichern der Daten der Angestellten und eine Tabelle [Firma] zum Speichern der Firmendaten:



Obwohl sinnvoll, erfüllen die in jeder einzelnen Tabelle gespeicherten Daten nicht Ihre Anforderungen zum Erfassen der Informationen. Wenn Sie sich einen Datensatz aus der Tabelle [Angestellte] ansehen, wollen Sie auch Informationen über die Firma haben, in welcher dieser Angestellte arbeitet; und wenn Sie sich einen Datensatz aus der Tabelle [Firma] ansehen, wollen Sie Informationen über alle Angestellten haben, die in dieser Firma arbeiten.

Sie können zwei Tabellen, die solche Informationen enthalten, miteinander verknüpfen - mit anderen Worten, zwischen den Daten jeder Tabelle kann eine Verbindung hergestellt werden.

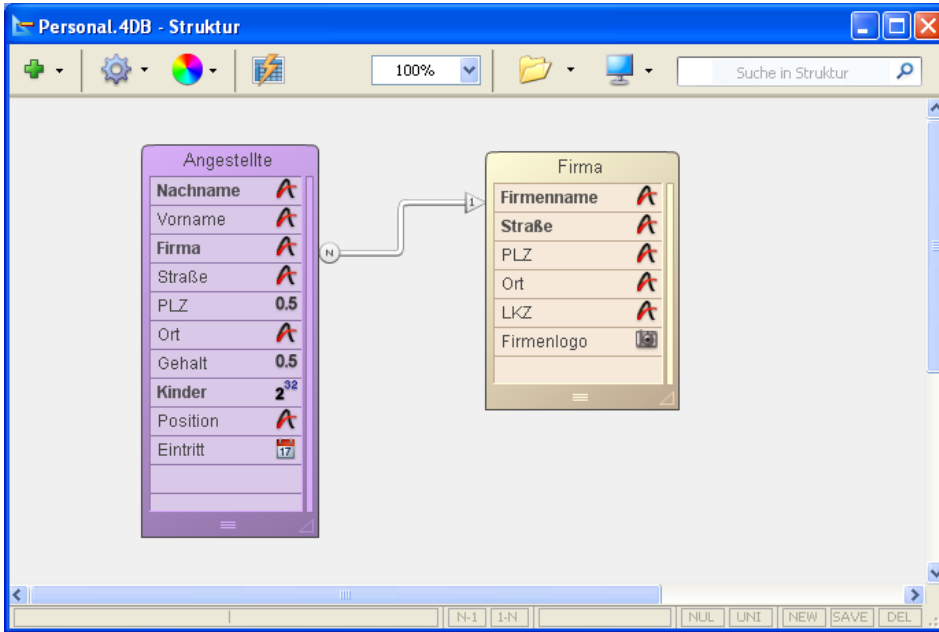
In 4D haben Sie durch Tabellenverknüpfungen Zugriff auf Daten aus einer anderen Tabelle. Tabellen, die Informationen über eine Verknüpfung verfügbar machen, werden als verknüpfte Tabellen bezeichnet.

Verknüpfte Tabellen bieten vielfältige Möglichkeiten. Sie können:

- Daten effektiv speichern

- Daten an einer Stelle aktualisieren, die Änderungen werden überall übernommen, wo diese Daten vorkommen
- Verknüpfte Informationen ansehen
- Suchen und Sortierungen in einer Tabelle ausführen, die auf Daten in einer anderen Tabelle beruhen
- In verknüpften Tabellen Datensätze erstellen, ändern oder löschen

Die folgende Abbildung zeigt eine Verknüpfung zwischen der Tabelle [Angestellte] und der Tabelle [Firma] im Struktureditor:



Die Tabelle [Angestellte] enthält einen Datensatz pro Angestellten. Die Tabelle [Firma] enthält einen Datensatz pro Firma. Die Verknüpfung zwischen den beiden Tabellen ermöglicht Ihnen, auf Informationen aus beiden Tabellen zuzugreifen, Informationen einzugeben, zu ändern oder zu löschen. Zum Beispiel:

- Ist der Datensatz eines Angestellten geöffnet, können Sie sich die zugehörige Firmeninformation ansehen oder ändern, z.B. die Firmenadresse aktualisieren.
- Wenn Sie einen neuen Angestellten hinzufügen, können Sie den Datensatz dieses Angestellten mit dem zugehörigen Datensatz der Firma verbinden, wenn die Firma bereits eingegeben ist, oder den neuen Datensatz der Firma erstellen, während Sie den Datensatz für diesen Angestellten erstellen. Weitere Informationen finden Sie im Abschnitt **Verknüpfungseigenschaften**.
- Sie können sich bei jeder Firma die Informationen für alle Angestellten dieser Firma ansehen oder ändern - Name, Titel, Telefonnummer usw. Sie können auch den Datensatz eines Angestellten innerhalb des Datensatzes der Firma hinzufügen.

Verknüpfte Datenfelder

Informationen aus verknüpften Tabellen zeigen Sie mit den verknüpften Datenfeldern an - das sind die Datenfelder, die zwei Tabellen über eine Verknüpfung miteinander verbinden. Das Verknüpfen von Tabellen soll 4D anweisen, welcher bzw. welche Datensätze in einer Tabelle zu aktualisieren sind, basierend auf dem aktualisierten Datensatz in der anderen Tabelle. Verknüpfte Tabellen verwenden Daten in zwei verknüpften Datenfeldern, um dazugehörige Datensätze zu erkennen. Im vorigen Beispiel ist der Firmenname in beiden Tabellen, der Tabelle [Angestellte] und der Tabelle [Firma] gespeichert. Das Firmenfeld in der Tabelle [Angestellte] und das Firmennamensfeld in der Tabelle [Firma] verknüpfen die beiden Tabellen. Das Firmennamensfeld in der Tabelle [Firma] ist das Primärschlüsselfeld für [Firma]. Es kennzeichnet jeden Firmendatensatz eindeutig. Ein Primärschlüsselfeld sollte die Eigenschaften Indiziert und Einmalig haben. Ist das Primärschlüsselfeld nicht indiziert, weist 4D dieses Attribut automatisch zu. Das Firmenfeld in der Tabelle [Angestellte] ist ein verknüpftes Datenfeld. Jeder Wert in einem verknüpften Datenfeld entspricht einem Wert in dem Primärschlüsselfeld einer anderen Tabelle. In diesem Beispiel passt ein Wert des verknüpften Datenfeldes in [Angestellte] genau zu einem Wert des Primärschlüsselfeldes in [Firma]. Das verknüpfte Datenfeld ist ebenfalls indiziert, aber seine Werte sind nicht eindeutig, da ja in der gleichen Firma mehrere Angestellte tätig sein können.

Ab 4D Version 14 müssen Felder mit Primärschlüsseln explizit in jeder Tabelle der Datenbank definiert werden. Die Werte des Primärschlüsselfeldes werden automatisch zugewiesen – entweder durch Zuweisen einer laufenden Nummer, die von 4D vergeben wird, oder durch eine vom Benutzer geschriebene Methode. Diese Vorgehensweise gewährleistet die Einmaligkeit des Schlüsselfeldes. Ist z.B. das Primärschlüsselfeld in der Tabelle [Firma] ein Datenfeld mit laufender Nummer anstelle des Firmennamens, wäre es für Benutzer möglich, mehrere Firmen mit dem gleichen Namen, aber unterschiedlichen Adressen einzugeben. Ändert sich ein Firmenname, kann der Benutzer ebenfalls die Datenbank ändern, ohne die Verknüpfung zwischen den beiden Tabellen zu beeinträchtigen.

Wenn der Benutzer berechtigt ist, den Wert des Primärschlüsselfeldes einzugeben, sollten Sie beide Eigenschaften **Einmalig** und **Nicht änderbar** vergeben (siehe **Feldeigenschaften**), um die Eindeutigkeit der Ersteingabe zu überprüfen und zu verhindern, dass Benutzer den einmal eingegebenen Eintrag ändern können. Wollen Sie die Eigenschaft **Nicht änderbar** nicht verwenden, müssen Sie anderweitig dafür sorgen, dass Benutzer keine "verwaisten" Datensätze in verknüpften Tabellen schaffen, da sie die Werte eines Primärschlüsselfeldes ändern können.

Sind die Verknüpfungen eingerichtet, können Sie Werte in eine Tabelle schreiben und auslesen, während Sie in der verknüpften Tabelle arbeiten. Wenn Sie z.B. einen Firmennamen in den Datensatz eines Angestellten eingeben, sucht 4D in der Tabelle [Firma] nach dieser Firma und zeigt die Firmenanschrift und Telefonnummer im Datensatz des Angestellten. Sehen Sie sich den Datensatz einer Firma an, sucht 4D in der Tabelle [Angestellte] nach allen Angestellten, die in dieser Firma arbeiten und zeigt deren Datensätze in dem Firmendatensatz.

Sie können diese Verknüpfungen automatisch aufrufen, d.h. Sie müssen nichts programmieren, oder Sie können Verknüpfungen manuell durchführen. Im zweiten Fall benutzen Sie Methoden, um verknüpfte Datensätze zu laden und wieder zu entfernen und kontrollieren das Erstellen, Ändern oder Löschen verknüpfter Datensätze. In komplexen Strukturen, in

denen mehr als zwei Tabellen miteinander verknüpft sind und Sie das Laden bzw. Nicht-Laden von verknüpften Datensätzen kontrollieren wollen, sind manuelle Verknüpfungen manchmal von Vorteil.

Bei automatischen Verknüpfungen ordnen Sie die entsprechenden Eigenschaften zu, wenn Sie die Verknüpfung zwischen den Tabellen festlegen. Weitere Informationen finden Sie im Abschnitt **Verknüpfungseigenschaften**.

Leistungsumfang der 4D Datenbank

Diese Übersicht zeigt die maximal möglichen Werte der 4D Datenbank Engine:

Leistungsumfang für	Maximum in der 4D Engine
Anzahl Tabellen	32,767
Anzahl Felder pro Tabelle	32,767
Anzahl Datensätze pro Tabelle	1 Milliarde
Anzahl Indexschlüssel pro Tabelle	128 Milliarden
Größe eines alphanumerischen Feldes	255 Zeichen
Größe eines Textfeldes	2 GB
Größe eines BLOB Feldes	2 GB
Größe eines Objektfeldes	2 GB
Anzahl Attribute pro Objektfeld	bis zu 128 Milliarden (abhängig von der Anzahl der Indexschlüssel)
Anzahl Transaktionsebenen	Unbegrenzt

Struktureditor

Um den Struktureditor zu öffnen, wählen Sie im Menü **Design>Datenbankstruktur** oder klicken Sie in der Werkzeugleiste auf das Icon .

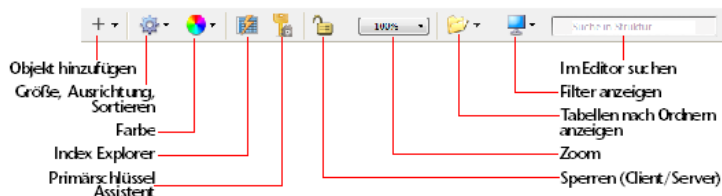
Mit dem Struktureditor verwalten Sie die Struktur der Datenbank, d.h. die Tabellen und ihre Verknüpfungen. Sie kontrollieren z.B. die Tabellen, Tabelleneigenschaften, Datenfelder, Feldeigenschaften und Tabellenverknüpfungen.

Der Struktureditor zeigt eine grafische Ansicht der Datenbankstruktur und stellt eine Werkzeugleiste sowie Kontextmenüs für Layoutoperationen zur Verfügung.

Im Struktureditor wird jede Tabelle grafisch dargestellt. Sie zeigt die Datenfelder und die zugewiesenen Typen in Form von Icons. Wenn Sie mit der Maus über ein Feld ziehen, erscheint am unteren Rand eine Informationsleiste mit seinen Merkmalen. Im Inspektorfenster können Sie die Eigenschaften der Strukturobjekte und des Struktureditors selbst ansehen und verändern. Weitere Informationen dazu finden Sie im Abschnitt **Inspektorfenster**.

Werkzeug und Informationsleiste

Der Struktureditor enthält eine Werkzeugleiste, die das Erstellen von Objekten und Navigieren erleichtern. Das sind Funktionen zum Hinzufügen von Objekten, Navigieren sowie Anzeigeeoptionen:



Am unteren Rand des Editors erscheint eine Leiste mit Informationen zum Bereich, über den die Maus gerade bewegt wird. Es gibt die Bereiche Tabelle, Feld oder Verknüpfung.

Angestellte	1	N-1	1-N	NUL	UNI	NEW	SAVE	DEL	LOAD			
Tabellenname und -nummer		Tabellen-Trigger: NEW = Neuer Datensatz sichern SAVE = Datensatz ändern DEL = Datensatz löschen LOAD = Datensatz laden										
Angestellte	1	Nachname	:	N-1	1-N	Alpha-2C	NUL	UNI	NEW	SAVE	DEL	LOAD
Tabellenname und -nummer		Datenfeldname und -nummer	Datenfeldtyp	Feldattribute: NUL = Setzt Nullwerte auf leere Werte UNI = Einmalig								
Songs	Albumname	Albumname	:	N-1	1-N	NUL	UNI	NEW	SAVE	DEL	LOAD	
Ausgangsfeld (Viele-Feld)		Zielfeld (Eine-Feld)	Verknüpfungsattribute: N-1 = Auto Eine-Verknüpfung 1-N = Auto Eine-zu-Viele									

Objekt auswählen

Um mit einem Objekt des Struktureditors zu arbeiten, müssen Sie es auswählen. Dann können Sie seine Eigenschaften festlegen, es bewegen, in der Größe verändern, löschen, etc. Sie können auch mehrere Objekte vom gleichen Typ simultan auswählen, um z.B. gemeinsame Eigenschaften in der Inspektorpalette anzuzeigen oder zu ändern. Sie können auch Datenfelder aus verschiedenen Tabellen auswählen.

Um eine Tabelle auszuwählen:

1. Klicken Sie auf die Tabellendarstellung
ODER
Wählen Sie die Taste **[Tab]** oder **Shift+[Tab]** um die Tabellen im Struktureditor nacheinander auszuwählen.
Die ausgewählte Tabelle erscheint mit blauem Rahmen. Nachfolgende Aktionen gelten dann für die gewählte Tabelle.

Um ein Datenfeld oder eine Verknüpfung auszuwählen:

1. Klicken Sie auf das Datenfeld oder die Verknüpfung
ODER
Drücken Sie bei ausgewählter Tabelle die Tasten **[Pfeil nach oben]** oder **[Pfeil nach unten]**, um die Datenfelder der Reihe nach auszuwählen.
ODER
Drücken Sie bei ausgewählter Tabelle die Tasten **[Pos1]** oder **[Ende]**, um das erste bzw. letzte Datenfeld der Tabelle auszuwählen.

Um mehrere Objekte vom selben Typ gleichzeitig auswählen (Tabellen, Felder, Verknüpfungen):

1. Verwenden Sie die Tastenkombination **[Shift+Klick]** um mehrere zusammenhängende Objekte auszuwählen
ODER
[Strg+Klick] unter Windows bzw. **[Befehl+Klick]** auf Mac OS, um mehrere nicht-zusammenhängende Objekte auszuwählen
ODER
Klicken Sie in einen leeren Bereich und ziehen ein Auswahlrechteck um das gewünschte Objekt (nur für Tabellenauswahl)
ODER
Wählen Sie im Menü **Bearbeiten** oder im Kontextmenü des Editors den Befehl **Alles auswählen** (nur für Tabellenauswahl).

Mit der Tabellenanzeige arbeiten

Sie können die Tabellenanzeige im Struktureditor beliebig in der Größe verändern oder bewegen.

In der Datenfeldliste blättern

Wenn Sie in einer Tabelle Datenfelder hinzufügen oder die Tabellengröße verkleinern, kann es mehr Datenfelder geben als in der Tabellendarstellung sichtbar sind. In diesem Fall fügt 4D am Rand automatisch eine Scrollbox hinzu, über die Sie durch die Datenfelder scrollen können:



Tabellengröße ändern

Sie können die Tabellendarstellung vergrößern, um weitere Datenfelder der Tabelle anzuzeigen oder Ihren Bildschirm anders zu organisieren. Es gibt mehrere Möglichkeiten:

- **Manuell anpassen:** Klicken Sie in den unteren Rand oder in die linke Ecke und ziehen die Tabelle größer oder kleiner



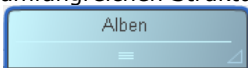
- **Optimale Größe:** Damit wird die ausgewählte Tabelle automatisch an die Anzahl der Felder angepasst. Leerzeilen werden nicht angezeigt. Es gibt folgende Möglichkeiten.

Um den Befehl **Optimale Größe** auf eine oder mehrere Tabellen anzuwenden, führen Sie folgendes aus:

- Wählen Sie den Befehl **Optimale Größe** in der Werkzeugpalette des Editors. Dieser Befehl erscheint nur, wenn eine Tabelle ausgewählt ist.



- Klicken Sie auf die gewünschte Tabelle und wählen im Kontextmenü den Befehl **Optimale Größe**.
- Doppelklicken Sie bei gedrückter **Umschalttaste** auf den Titel der entsprechenden Tabelle. Wiederholen Sie diese Kombination, läuft folgende Sequenz ab: Originalgröße -> optimale Größe -> Zuklappen.
- **Zuklappen:** Damit klappt das Bild der gewählten Tabelle zu, so dass nur der Titel sichtbar bleibt. Das ist besonders bei umfangreichen Strukturen hilfreich.



Es gibt folgende Möglichkeiten:

- Klicken Sie in den Titel der gewünschten Tabelle und wählen im Kontextmenü den Befehl **Zuklappen**,
- Doppelklicken Sie bei gedrückter **Umschalttaste** auf den Titel der entsprechenden Tabelle. Wiederholen Sie diese Kombination, läuft folgende Sequenz ab: Originalgröße -> optimale Größe -> Zuklappen.

Hinweise:

- Ein Doppelklick auf den Tabellentitel öffnet den Inspektor.
- **Alt-Taste** auf Windows, **Wahltaste** auf Mac OS + Doppelklick öffnet die Trigger-Methode der Tabelle im Methodeneditor
- **Strg**-Taste auf Windows, **Befehlstaste** auf Mac OS + Doppelklick öffnet den Explorer auf der Seite Formulare.

4D Server: Wenn Sie eine Tabellendarstellung unter 4D Server in der Größe ändern, wird die Tabelle für alle Benutzer im Designmodus geändert.

Tabellenanzeige bewegen

Sie können die Tabellendarstellungen im Strukturfenster bewegen, um sie nach ihrer Funktion zu gruppieren oder das Fenster neu zu organisieren. Sie können Tabellen auch bündig ausrichten, um die Lesbarkeit zu verbessern. Sind Verknüpfungen

vorhanden, werden sie dazu passend verschoben.

Um eine Tabelle manuell zu bewegen:

1. Ziehen Sie bei gedrückter Maustaste die Titelleiste der Tabelle.
Ziehen Sie bei gedrückter Maustaste nur die Titelleiste. Das Ziehen anderer Teile der Darstellung kann andere Effekte hervorrufen, z.B. eine neue Tabellen-Verknüpfung erstellen oder die Größe einer Darstellung verändern.

4D Server: Wenn Sie eine Tabellendarstellung unter 4D Server bewegen, erscheint diese Tabelle für alle Benutzer des Designmodus an der neuen Stelle.

Um Tabellen auszurichten:

1. Wählen Sie die gewünschten Tabellen und dann in der Werkzeugpalette eine Option zum Ausrichten:



Diese Optionen sind nur aktiv, wenn mindestens zwei Tabellen ausgewählt sind.

Scrollen

Spezifische Funktionen zum Scrollen erleichtern das Navigieren bei komplexen Strukturen:

- Mit dem Mausecndchen können Sie in vertikaler Richtung durch den Inhalt des Editorfensters scrollen. Setzen Sie den Cursor auf eine Tabelle, können Sie durch die Datenfelder dieser Tabelle scrollen.
- Mit gedrückter **Umschalttaste** können Sie mit dem Mausecndchen in horizontaler Richtung durch den Inhalt des Fensters scrollen.
- Drücken Sie die **Umschalttaste**, wird die "Hand" aktiv, mit der Sie den gesamten Inhalt des Fensters ziehen können, wenn Sie in einen leeren Bereich klicken.

Zoom

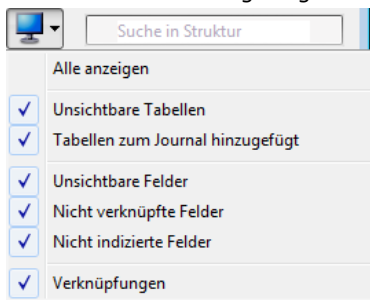
Über das DropDown-Menü **Zoom** in der Werkzeugleiste können Sie die Anzeige der Struktur verändern. Beim Öffnen der Datenbank ist standardmäßig 100% eingestellt. Gibt es eine Auswahl, wird der Zoom darauf angewendet.

Die aktuelle Zoom-Einstellung ist für jeden Benutzer spezifisch und wird beim Schließen des Fensters gespeichert.

Hinweis: In den 4D Einstellungen können Sie die grafische Qualität des Struktureditors definieren. Weitere Informationen dazu finden Sie auf der [Seite Struktur](#).

Objekttypen anzeigen


Die Schaltfläche **Anzeige** in der Werkzeugleiste ist mit einem PopUp-Menü gekoppelt. Hier können Sie die Objekte wählen, die in der Struktur angezeigt werden sollen:




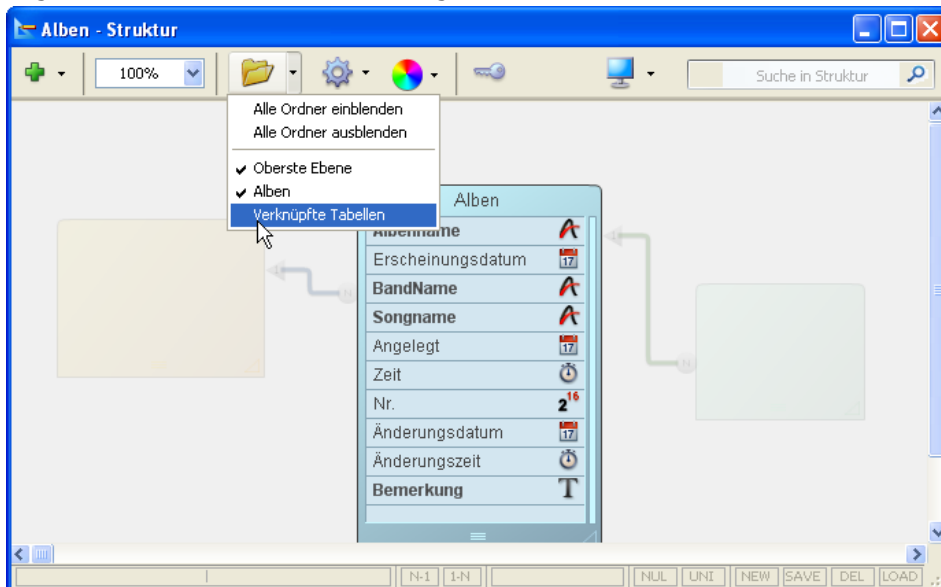
Standardmäßig werden alle Objekte angezeigt. Diese Funktionalität ermöglicht verschiedene Darstellungen und Ebenen — von der einfachsten bis zur komplexesten — und bietet gleichzeitig ein Werkzeug zum Analysieren der Struktur je nach gewünschter Informationsebene. Eine Auswahl gilt für alle Tabellen und Felder. Die jeweilige Einstellung wird pro Benutzer gesichert und beim Schließen des Fensters abgespeichert. Das Ausblenden bestimmter Objekte verändert nicht die Position der Tabellen.

- Sind mehrere Datenfeldtypen markiert, bestimmt der logische Vergleichsoperator ODER die Anzeige der Objekte. Sind z.B. die Optionen **Unsichtbare Felder** und **Nicht indizierte Felder** markiert, werden alle Datenfelder vom Typ **Nicht indiziert** (sichtbar oder unsichtbar) und **Unsichtbar** (indiziert oder nicht) angezeigt.
- Tabellen haben Vorrang vor Datenfeldern: Wird eine Tabelle nicht angezeigt, werden die dazugehörigen Datenfelder ebensowenig angezeigt.
- Fügen Sie ein Objekt hinzu, dessen Typ nicht angezeigt wird (Tabelle, Datenfeld oder Verknüpfung), erscheint es im Editor. Sie müssen die entsprechende Option erneut im PopUp-Menü der Schaltfläche **Anzeige** auswählen, um es auszublenden.

Tabellen nach Ordner ein-/Ausblenden

Sie können im Struktureditor Tabellengruppen ausblenden, die in Ordnern liegen, die im Explorer auf der **Seite Home** definiert wurden. Über die Schaltfläche  in der Werkzeugleiste können Sie die Tabellenanzeige steuern. Das zugeordnete DropDown-Menü enthält die Befehle zum Verwalten der Anzeige sowie die Liste mit den in der Datenbank angelegten Ordnern. Jeder angezeigte Ordner ist mit einer Marke gekennzeichnet. Sie können die aktuelle Anzeige durch Auswählen oder Abwählen der Ordner in diesem Menü verändern.

Mit den Befehlen **Alle Ordner einblenden/Alle Ordner ausblenden** können Sie alle Tabellen in Ordnern der Datenbank anzeigen bzw. ausblenden. Jeder Klick auf die Schaltfläche  kehrt die Anzeige um, d.h. eingeblendete Tabellen werden ausgeblendet und umgekehrt. Bei ausgeblendeten Tabellen erscheinen nur die Umrisse im Struktureditor. Nur die eingeblendeten Tabellen sind vollständig sichtbar:



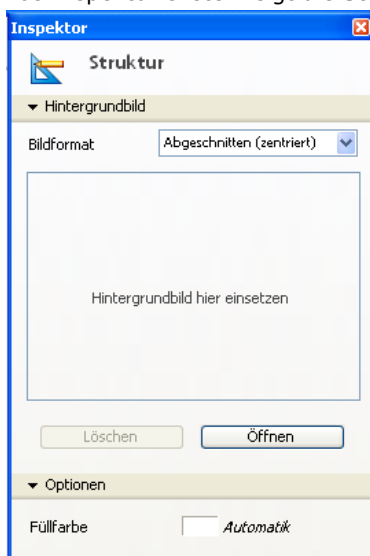
Hinweis: Sie können im Struktureditor die Anzeige von den per Ordner ausgeblendeten Tabellen auf Umriss oder Unsichtbar setzen. Diese Option wird in den 4D Einstellungen auf der **Seite Struktur** festgelegt. Eine Änderung wird erst wirksam, wenn Sie den Struktureditor schließen und erneut öffnen.

Editorfenster anpassen

Der Struktureditor hat spezifische Eigenschaften, die sich über den Inspektor einrichten lassen. Sie können die Eigenschaften folgendermaßen aufrufen:

- Doppelklick (oder Klick, wenn das Fenster schon angezeigt wird) in einen leeren Bereich des Fensters ODER
- Rechter Mausklick in einen leeren Bereich des Fensters und im Kontextmenü den Eintrag Struktureigenschaften wählen.

Das Inspektorfenster zeigt die Struktureigenschaften:



Sie können folgende Eigenschaften festlegen:

- **Hintergrundbild:** Sie können sowohl das Hintergrundbild als auch dessen Anzeigeformat ändern. Um das Bild zu ändern, klicken Sie auf die Schaltfläche **Öffnen** oder mit der rechten Maustaste in die Vorschau und wählen dann im Kontextmenü den Eintrag **Öffnen**. Wählen Sie die Datei mit dem gewünschten Bild. Sie können jedes Bildformat verwenden. Das gewählte Bild erscheint sofort in der Vorschau und im Struktureditor. Um das Anzeigeformat des Bildes zu ändern, wählen Sie einen Wert im PopUp-Menü **Bildformat**. Zur Verfügung stehen die standardmäßigen Bildformate von 4D. Um ein eigenes Bild zu löschen, klicken Sie auf die Schaltfläche **Löschen** oder mit der rechten Maustaste in die Vorschau und wählen dann im Kontextmenü den Eintrag **Löschen**.

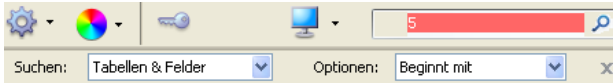
- **Füllfarbe:** Sie können die Farbe für den Hintergrund des Struktureditors verändern. Dazu klicken Sie in den Bereich zum Auswählen einer Farbe und wählen die gewünschte Farbe aus.
- **Standardschrift:** Um die Standardschrift für Tabellen- und Feldnamen zu ändern, wählen Sie einen Wert aus diesem Menü. Über das Menü **Größe** können Sie die standardmäßige Schriftgröße verändern.

Suche in der Struktur

In 4D können Sie Suchläufe im Struktureditor ausführen. Sie können nach folgenden Elementen suchen:

- Tabellennamen und/oder Datenfeldnamen
- Tabellennummern

Um eine Suche auszuführen, geben Sie im Suchbereich des Struktureditors eine Zeichenkette oder eine Tabellennummer ein. Sobald Sie einen Wert eingeben, erscheinen darunter zwei weitere Eingabebereiche, wo Sie den gewünschten Suchbereich und Suchtyp angeben können:

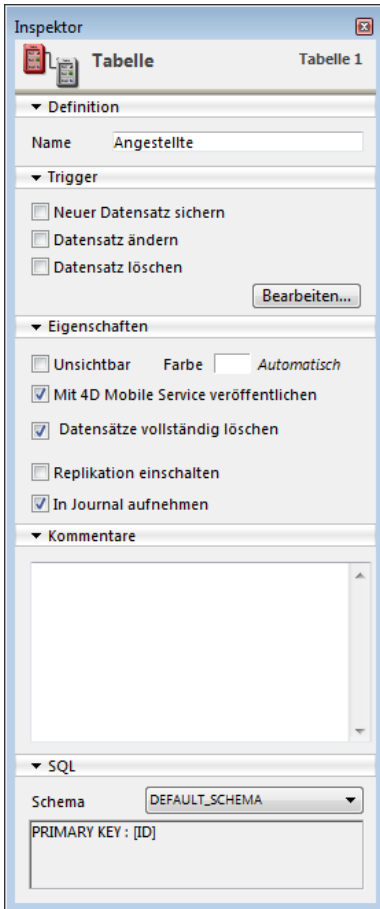


- Im PopUp-Menü **Suchen** legen Sie den Bereich fest: Sie können nach **Tabellen und Feldern** oder nur nach **Tabellen** suchen.
- Im PopUp-Menü **Optionen** definieren Sie die Art der Suche:
 - **Beinhaltet** (Standardeinstellung): Suchen nach "le" findet "Tabelle", "Elemente", "Lektion" usw.
 - **Beginnt mit:** Suchen nach „pa“ findet „Papa“, „Papier“, jedoch nicht Reparatur.
 - **Nummer:** Suchen nach "2" findet die Tabellennummern 2, 12, 20, 21, etc.

Suchläufe werden in Echtzeit ausgeführt, so wie Sie die Werte eingeben. Die gefundenen Tabellen und Datenfelder werden hervorgehoben. Wird nichts gefunden, erscheint der Suchbereich rot. Wollen Sie den Suchmodus verlassen, klicken Sie auf das Icon neben dem Eingabebereich für Optionen oder entfernen Sie alle Zeichen im darüberliegenden Eingabebereich.

Inspektorfenster

Über das Inspektorfenster können Sie die Eigenschaften der Strukturobjekte (Tabellen, Felder, Verknüpfungen) ansehen und verändern:



Dieses Fenster erscheint, wenn Sie auf eine Tabelle doppelklicken. Der Inhalt richtet sich nach dem gewählten Objekt. Klicken Sie in einen leeren Bereich des Fensters, zeigt das Fenster die allgemeinen Eigenschaften des Struktureditors an. Weitere Informationen dazu finden Sie im Abschnitt **Struktureditor**.

Sie können die einzelnen Bereiche im Fenster auf- bzw. zuklappen. Klicken Sie dazu auf den Pfeil vor dem jeweiligen Titel, um die dazugehörige Information ein- oder auszublenden.



Es gibt folgende Tastenkürzel:

- **Umschalttaste + Klick** auf die Titelleiste eines geschlossenen Bereichs öffnet diesen und schließt alle anderen.
- **Alt** (Windows) oder **Wahltaste** (Mac OS) + **Klick** auf die Titelleiste eines geschlossenen Bereichs öffnet alle Bereiche.
- **Alt** (Windows) oder **Wahltaste** (Mac OS) + **Klick** auf die Titelleiste eines geöffneten Bereichs schließt alle Bereiche.

Die Fensterposition und der Status auf-/zugeklappt der Bereiche werden beim Schließen gesichert.

Tabellen erstellen und ändern

Sie können Tabellen jederzeit erstellen. 4D nennt die erste Tabelle [Tabelle_1] und weitere Tabellen mit fortlaufender Nummer bis zu [Tabelle_32767]. Sie können den Tabellen für Ihre Zwecke aussagekräftige Namen geben.

Sie können überholte oder versehentlich angelegte Tabellen löschen. Das Löschen kann dauerhaft sein oder nicht. Beachten Sie bei endgültigem Löschen, das alle mit der Tabelle gespeicherten Daten auch gelöscht werden. Sie können eine Tabelle auch unsichtbar machen, so dass sie nicht mehr in den Editoren des Anwendungsmodus erscheint.

Sie können jede Tabelle nach Bedarf mit Triggern und Attributen ausrüsten. Weitere Informationen dazu finden Sie im Abschnitt **Tabelleneigenschaften**.

Tabelle erstellen

Sie können eine neue Tabelle entweder direkt im Strukturfenster oder über ein Dialogfenster anlegen. Direktes Erstellen ist schneller, über das Dialogfenster lassen sich jedoch bestimmte Parameter für die Tabelle angeben und die Operation bei Bedarf abbrechen. Sie können Tabellen auch per Kopieren/Einsetzen erstellen.

Um eine Tabelle direkt anzulegen:

1. Klicken Sie bei gedrückter rechter Maustaste in einen leeren Bereich des Strukturfensters und wählen im Kontextmenü den Eintrag **Neue Tabelle**

ODER

Wählen Sie in der Werkzeugleiste des Struktureditors im Drop-Down-Menü **Hinzufügen** den Eintrag **Tabelle**:



4D erstellt die Tabelle direkt im Struktureditor. Sie können dann den Tabellentitel verändern, Felder hinzufügen, etc.

Hinweis: Soll die Tabelle in einem bestimmten Ordner liegen, müssen Sie die **Seite Home** des Explorers verwenden.

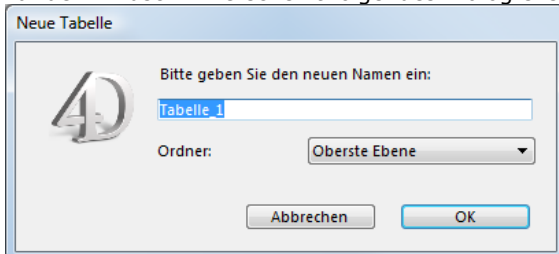
Um eine Tabelle über das Dialogfenster "Neue Tabelle" anzulegen:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Neu>Tabelle** oder klicken Sie in der Werkzeugleiste auf das Icon **Neu**.

ODER

Klicken Sie auf der **Seite Tabellen** des Explorers auf die Schaltfläche **+**.

Auf dem Bildschirm erscheint folgendes Dialogfenster:



2. (optional) Ändern Sie bei Bedarf den Tabellennamen.
Der Tabellename kann bis zu 31 Zeichen lang sein und muss mit einem Buchstaben beginnen. Er kann Buchstaben, Ziffern, Leerzeichen und Unterstriche enthalten. 4D schneidet längere Namen ab und entfernt Leerzeichen zu Beginn und am Ende des Namens.
3. (optional) Wählen Sie den Ordner, in welchen die neue Tabelle gelegt werden soll.
Die Tabelle liegt standardmäßig auf der obersten Ebene der Ordnerliste. Weitere Informationen dazu finden Sie im Abschnitt **Seite Home**.
4. Klicken Sie auf die Schaltfläche **OK**.
Wollen Sie die Operation nicht ausführen, klicken Sie auf die Schaltfläche **Abbrechen**.
4D erstellt eine neue Tabellendarstellung. Diese wird die ausgewählte Darstellung im Fenster des Struktureditors.

Hinweis: Ab 4D v14 hat jede neu erstellte Tabelle in der Anwendung standardmäßig einen Primärschlüssel. Weitere Informationen dazu finden Sie im Abschnitt **Standardfeld mit Primärschlüssel**.

Um eine Tabelle über Kopieren/Einsetzen zu erstellen:

1. Wählen Sie die gewünschte Tabelle und dann den Standardbefehl zum Kopieren: im 4D Menü **Bearbeiten** bzw. im Kontextmenü den Befehl **Kopieren** oder die Tastenkombination **Strg+C** unter Windows, **Befehl+C** auf Mac OS.
2. Wählen Sie dann den Befehl **Einsetzen**, entweder im Menü **Bearbeiten**, im Kontextmenü oder über die Tastenkombination **Strg+V** unter Windows, **Befehl+V** auf Mac OS. Die Tabelle wird mit allen Datenfeldern übertragen und standardmäßig umbenannt in *Kopie(X)_von_Tabellenname*. Dabei ist *Tabellenname* der ursprüngliche Tabellename, X die Nummer der Kopie der Tabelle.

Tabelle umbenennen

Sie können Tabellen jederzeit umbenennen. Wenn Sie in einer Methode den alten Namen verwendet haben, ändert 4D ihn automatisch um in den neuen Tabellennamen, vorausgesetzt die Methode ist geschlossen. Bei geöffneter Methode müssen Sie die Änderung selbst vornehmen. Innerhalb einer Anwendung muss jeder Tabellename einmalig sein.

Sie können eine Tabelle direkt im Struktureditor, im Explorer oder im Inspektorfenster umbenennen.

Um eine Tabelle direkt umzubenennen (Struktureditor oder Seite Tabellen im Explorer):

1. Doppelklicken Sie auf den Tabellennamen oder klicken Sie einmal auf den Namen, wenn die Tabelle bereits ausgewählt ist.

Der Titel wechselt in den Bearbeitungsmodus:



2. Geben Sie den neuen Namen ein und klicken Sie außerhalb der Tabelle.

Um eine Tabelle über das **Inspektorfenster** des Struktureditors umzubenennen:

1. Doppelklicken Sie auf den Tabellennamen

ODER

Klicken Sie mit der rechten Maustaste auf die Tabelle und wählen dann im Kontextmenü den Eintrag

Tabelleneigenschaften

ODER

Doppelklicken Sie im Explorer auf der Seite **Tabellen** auf den Tabellennamen.

Im Inspektor erscheinen Name und Eigenschaften der gewählten Tabelle sowie die Tabellenummer. Ist der Inspektor bereits geöffnet und zeigt die Eigenschaften einer anderen Tabelle oder eines anderen Objekts, wählen Sie die gewünschte Tabelle aus. Der Inspektor zeigt dann die Eigenschaften dieser Tabelle.

2. Geben Sie den neuen Namen im Bereich Name ein.

Er erscheint sofort in der Tabelle.

Hinweis: Weitere Informationen zu den Namensregeln finden Sie im Abschnitt **GET REGISTERED CLIENTS**.

Tabelle löschen

Sie können Tabellen aus Ihrer Datenbank löschen.

Diese Operation lässt sich über die SQL-Engine von 4D oder über den Struktureditor ausführen. Weitere Informationen dazu finden Sie im Handbuch *4D SQL Reference* unter **SET MENU ITEM**.

Sie können eine Tabelle auf zwei Arten löschen: Dauerhaftes Löschen, d.h. die Tabelle und ihre Daten werden aus der Datenbank entfernt; und vorübergehendes Löschen, d.h. die Tabelle wird in den Papierkorb gelegt (siehe **Seite Papierkorb**) und kann später wieder daraus entnommen werden.

Dauerhaftes Löschen

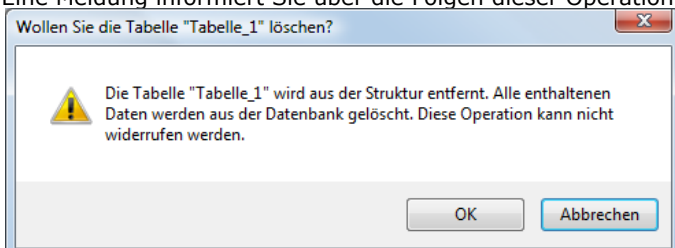
Um eine oder mehrere Tabellen im Struktureditor zu löschen:

1. Markieren Sie die entsprechende Tabelle und wählen dann im Menü **Bearbeiten** den Befehl **Löschen**

ODER

Wählen Sie im Kontextmenü der Tabelle den Eintrag **Löschen**

Eine Meldung informiert Sie über die Folgen dieser Operation:



Klicken Sie auf die Schaltfläche **OK**, führt 4D folgendes aus:

- Die Tabelle wird unwiderruflich aus der Struktur entfernt. Alle damit verknüpften Daten werden unwiderruflich aus der Datendatei gelöscht.
- Alle mit der Tabelle verknüpften Trigger-Methoden werden gelöscht.
- Die der Tabelle zugeordneten Formulare werden in Projektformulare umgewandelt und in den Papierkorb des Explorers gelegt. Weitere Informationen dazu finden Sie im Abschnitt **Seite Papierkorb**.

Vorübergehendes Löschen

Bei vorübergehendem Löschen wird die Tabelle in den Papierkorb der Datenbank gelegt. Das funktioniert wie beim Löschen von Dateien unter Windows und auf Mac OS. Der Papierkorb ist über den Explorer zugänglich.

Die Tabelle erscheint dann nicht mehr in den 4D Editoren, ihr Inhalt ist nicht mehr zugänglich. Die Tabelle lässt sich aber wiederherstellen, solange der Papierkorb nicht geleert wird.

Um eine oder mehrere Tabellen aus dem Struktureditor in den Papierkorb zu legen:

1. Wählen Sie die entsprechende(n) Tabelle(n) aus und drücken die **Löschen-** bzw. **Rückschritt-Taste**
ODER
Wählen Sie im Kontextmenü der Tabelle den Eintrag **In Papierkorb legen**.
In beiden Fällen erscheint ein Dialogfenster zum Bestätigen.
2. Klicken Sie auf die Schaltfläche **OK**, wird die Tabelle in den Papierkorb gelegt. Sie können sie jederzeit wieder daraus hervorholen. Weitere Informationen dazu finden Sie im Abschnitt [Seite Papierkorb](#).

Nummern gelöschter Tabellen

Wird eine Tabelle gelöscht, werden die anderen Tabellen und Felder der Datenbank nicht umnummeriert, damit die Stabilität der Datenbank gewahrt bleibt. So kann eine Datenbank z.B. drei Tabellen mit den Nummern 2, 4 und 5 haben.

Dagegen werden die Nummern dauerhaft gelöschter Tabellen beim Anlegen neuer Objekte wieder verwendet.

Tabelleneigenschaften

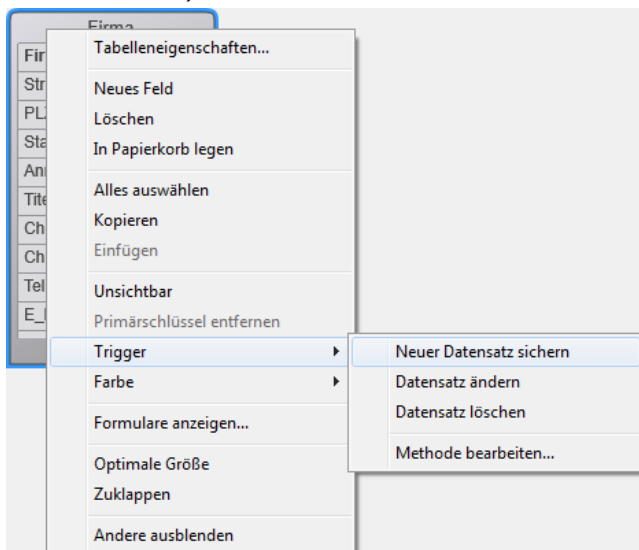
Sie können die Tabelleneigenschaften über das **Inspektorfenster** definieren. Bestimmte Eigenschaften lassen sich auch das Kontextmenü für Tabellen festlegen.

Trigger

Ein **Trigger** ist eine automatisch ablaufende Methode, sobald bestimmte Ereignisse in der Tabelle auftreten. Solche Ereignisse sind:

- On saving new record
- On saving an existing record
- On deleting record

Trigger-Ereignisse können Sie im **Inspektorfenster** oder über das Kontextmenü der Tabelle einrichten (rechter Mausklick in den Tabellentitel):



Markieren Sie die Ereignisse, für die Sie einen Trigger aktivieren wollen.

Sie können direkt im Struktureditor auf Trigger zu einer Tabelle zugreifen, wenn Sie im Inspektor die Schaltfläche **Bearbeiten**, im Kontextmenü den Eintrag **Methode bearbeiten** wählen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Methodenarten**.

Hinweise:

- Sie können Trigger auch im Explorer auf der **Seite Methoden** des Explorer einrichten und anzeigen
- Die Informationsleiste im **Struktureditor** zeigt an, welche Trigger für eine Tabelle aktiviert sind

Trigger, die Sie im Methodeneditor eingegeben haben, laufen nur, wenn die von Ihnen bestimmten Ereignisse auftreten.

Attribute

Sie können verschiedene Attribute zuweisen, um die Darstellung und das allgemeine Verhalten einer Tabelle festzulegen.

Unsichtbar

Die Option „Unsichtbar“ blendet eine Tabelle im Anwendungsmodus und für Plug-Ins aus. Aktivieren Sie diese Option für Hilfstabellen oder Tabellen, die zeitweise nicht benutzt werden. Auf diese Weise können Sie auch steuern, welche Operationen ein Benutzer in einer Tabelle durchführen kann. Sie blenden die Tabelle und ihre Datenfelder in allen Editoren und einigen Dialogfenstern des Anwendungsmodus oder für Plug-Ins aus. Dazu gehören:

- Alle Sucheditoren (siehe **Datensätze suchen**)
- **Sortiereditor**
- **Etiketteneditor**
- **Schnellberichteditor**
- Dialogfenster für **Daten importieren und exportieren**
- **Formeleditor**

In keinem dieser Editoren kann der Benutzer die Tabelle oder eines ihrer Datenfelder sehen oder anwählen. Der Benutzer kann z.B. keine Datenfelder einer ausgeblendeten Tabelle in einen Schnellbericht oder ein Etikett setzen.

Hinweis: Wenn Benutzer diese Editoren einsetzen, können sie ihre Spezifikationen in Dateien auf der Platte speichern, z.B. Kriterien einer Suche oder Sortierung. Dafür werden auch unsichtbare Tabellen oder Datenfelder verwendet. Außerdem können Benutzer die Namen von ausgeblendeten Tabellen und Datenfeldern in das Dialogfenster eingeben.

Im Designmodus erscheinen ausgeblendete Tabellen und Datenfelder im Fenster des Struktureditors kursiv.


Farbe der Tabellendarstellung

Sie können für jede Tabelle eine Farbe festlegen. Mit Farben machen Sie die Struktur umfangreicher Datenbanken übersichtlicher. Sie können z.B. eine Farbe für alle Tabellen zu Kunden definieren und eine andere für Tabellen zu Artikeln.



Hinweis: Sie können auch jedem Feld oder jeder Verknüpfung eine Farbe zuweisen (siehe Abschnitt **Definition**).

Wählen Sie eine oder mehrere gewünschte Tabellen aus und weisen eine Farbe zu. Diese können Sie folgendermaßen aufrufen:

- In der Werkzeugpalette des Editors mit der Schaltfläche **Farbe** 
- Im Kontextmenü der Tabellen mit dem Eintrag **Farbe**,
- Im **Inspektorfenster** mit der Option **Farbe**.

Die Option **Automatik** stellt wieder die ursprüngliche Farbe der Tabelle her.

Datensätze definitiv löschen

Mit dieser Option können Sie das Löschen einer Datensatzauswahl optimieren, die über den 4D Befehl **DELETE SELECTION** erstellt wurde.

Wenn 4D eine Auswahl löscht, werden auch die Datensatz-Marker gelöscht. Ein Marker ist ein dem Datensatz zugewiesener Kopfteil mit Informationen zu diesem Datensatz. Das Löschen von Markern und Datensätzen ist langsamer als nur Datensätze löschen. In bestimmten Fällen ist es wünschenswert, nicht automatisch die Datensatz-Marker zu löschen.

Diese Option ermöglicht, die gewünschte Löschart zu wählen. Um das Löschen einer umfangreichen Auswahl über den 4D Befehl **DELETE SELECTION** zu beschleunigen, deaktivieren Sie die Option **Datensätze definitiv löschen**. Dann werden die Datensatz-Marker nicht gelöscht. Diese Option lässt sich nicht per Programmierung setzen.

Replikation einschalten

Ist diese Option markiert, erzeugt 4D die erforderliche Information zur Replizierung der Datensätze der Tabelle (Sie basiert insbesondere auf dem Primärschlüssel der Tabelle). Damit lassen sich Daten zwischen zwei oder mehr 4D Anwendungen synchronisieren.

Ist die Option aktiv, muss der Mechanismus zur Replikation über spezifische Befehle der SQL Programmiersprache in 4D implementiert werden. Weitere Informationen dazu finden Sie im Handbuch *4D SQL Reference* im Abschnitt **Replication via SQL**. Weitere Informationen zur Replikation mit HTTP finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **URL 4DSYNC/**.

Diese Option ist standardmäßig nicht markiert. Damit sie verfügbar wird, müssen Sie einen Primärschlüssel für die zu replizierende Tabelle definieren. Andernfalls ist die Option in Grauschrift. Den Primärschlüssel setzen Sie direkt im Struktureditor (siehe unten).

Mit 4D Mobile Service veröffentlichen

Diese Option steuert, ob die Tabelle für 4D Mobile Anfragen, die an die 4D Datenbank gesendet werden, zugänglich ist. Standardmäßig sind alle Tabellen in 4D Mobile veröffentlicht.

Aus Sicherheitsgründen wollen Sie aber nur bestimmte Tabellen Ihrer Datenbank für 4D Mobile Aufrufe veröffentlichen. Sie wollen z.B. die Tabelle [Benutzer], die Namen und Kennwörter speichert, nicht veröffentlichen.

Wollen Sie eine Tabelle bzw. alle darin enthaltenen Felder nicht veröffentlichen, deaktivieren Sie die Option **Mit 4D Mobile Service veröffentlichen** für die Tabelle.

Hinweis: Sie können diese Option auch für ein einzelnes Feld der Tabelle setzen, siehe **Feldeigenschaften**.

Diese Option lässt sich im Rahmen der "4D Mobile" Funktionalität verwenden. Weitere Informationen dazu finden Sie im Handbuch **4D Mobile**.

In Journal aufnehmen

Diese Option ist standardmäßig für jede neu angelegte Tabelle und für alle Tabellen in konvertierten Datenbanken markiert. Diese Option sollte für die Tabellen in der Regel markiert sein. Für temporäre Tabellen oder alles was keine Überwachung benötigt, können Sie die Option deaktivieren und so die Größe des Logbuchs verringern.

Hinweis: Die Option erscheint in Grauschrift, wenn die Tabelle keinen Primärschlüssel hat.

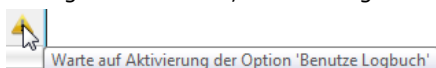
Bitte beachten Sie, dass diese Option nur angibt, dass die Daten der Tabelle protokolliert werden sollen; sie aktiviert nicht das Führen des Logbuchs selbst auf Strukturebene (siehe **Logbuch verwalten**).

Warnung

4D zeigt rechts neben der Option **In Journal aufnehmen** ein Warn-Icon, wenn die erforderlichen Bedingungen nicht erfüllt sind:



Solange es erscheint, ist kein Logbuch aktiviert. Ziehen Sie die Maus über das Icon, erscheint eine Erläuterung zur Warnung



Es gibt folgende Meldungen:

Meldung	Ursache	Erforderliche Korrektur
Warte auf Festlegung eines Primärschlüssels	IOperationen lassen sich nicht protokollieren, wenn die Tabelle keinen Primärschlüssel hat	In der Tabelle einen Primärschlüssel per SQL erstellen oder das Kontextmenü der Tabelle nutzen
Warte auf Korrigieren der Werte des Primärschlüssels	Werte des Primärschlüssels wurden geprüft und Anomalien festgestellt	Duplikate oder Nullwerte in den Datensätzen für das Feld entfernen (oder einen anderen Primärschlüssel nutzen)
Warte auf Aktivierung des Logbuch auf Datenbankebene	Die globale Option zum Aktivieren des Logbuchs ist nicht markiert	Die Option 'Benutze Logbuch' in den Datenbank-Eigenschaften auf der Seite Backup>Konfiguration prüfen

Kommentar

Im Bereich Kommentar des **Inspektorfenster** können Sie zusätzliche Information zur Tabelle ablegen. Diese Kommentare sind für alle Entwickler zugänglich. Beachten Sie, dass jedes Feld und jede Verknüpfung einen eigenen Kommentarbereich hat.

SQL

Der Bereich SQL des **Inspektorfenster** enthält das PopUp-Menü Schemas und einen Informationsbereich.

- Das Menü Schemas listet alle SQL Schemas auf, die in der Datenbank spezifiziert sind. Über dieses Menü können Sie das Schema setzen, zu dem die Tabelle gehören soll (Sie können diese Eigenschaft auch über SQL Befehle ändern). Jede Datenbank hat mindestens ein Schema mit Namen DEFAULT_SCHEMA. Standardmäßig gehören alle Tabellen zu diesem Schema. Weitere Informationen dazu finden Sie im Handbuch *4D SQL Reference* unter **Schemas**.
- Der Informationsbereich gibt an, ob der Name der Tabelle für SQL nicht-erlaubte Zeichen enthält. SQL erlaubt z.B. im Gegensatz zu 4D keine Namen mit Leerzeichen.

Primärschlüssel

Ein Primärschlüssel gibt die Felder zur eindeutigen Identifizierung der Datensätze in einer Tabelle an. Das ist insbesondere für die Funktionalität *Datensatz Replikation* in einer 4D Tabelle (siehe **Replication via SQL**) und zum Protokollieren der Daten (siehe **Logbuch verwalten**) notwendig.

Sie können Primärschlüssel direkt über das Kontextmenü des Struktureditors erstellen, bearbeiten und entfernen.

Hinweis: Primärschlüssel lassen sich auch in der SQL Sprache über die Klausel PRIMARY KEY, gefolgt von der Liste der Spalten, setzen (siehe **Primary key** im Handbuch *4D SQL Reference*).

Regeln für Felder mit Primärschlüssel

Felder mit Primärschlüssel sollten mit Bedacht verwaltet werden, um jederzeit die Datenintegrität zu gewährleisten. Sie müssen insbesondere folgende Regeln einhalten:

- sie dürfen nicht leer sein
- sie müssen einmalig sein
- einmal angelegt, sollten sie prinzipiell nie verändert werden, besonders wenn:
 - 4D Features zur Replikation oder Synchronisation aktiviert sind
 - 4D Mobile (oder ähnlich) aktiviert ist
 - die Logbuch-Funktion der Anwendung aktiviert ist (Backup oder logisches Spiegeln über das Logbuch)
 - Primärschlüssel zur Kommunikation oder Synchronisation mit externen Systemen verwendet werden.

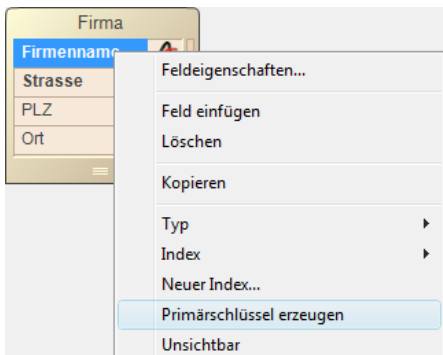
Wir raten generell davon ab, Primärschlüssel zu verändern, selbst wenn 4D das für spezifische Fälle erlaubt. Wenn Sie dennoch Ihr Feld mit Primärschlüssel verändern müssen, ist es besser, stattdessen ein neues Feld mit Primärschlüssel mit automatischer Zuweisung anzulegen.

Felder mit Primärschlüssel können vom Typ Lange Ganzzahl oder im UUID Format (alphanumerisch) sein. Für die meisten Fälle ist das UUID Format mit aktivierter Option Auto UUID besser geeignet. Denn UUIDs haben viele Vorteile, auch wenn der Typ Lange Ganzzahl weniger Speicherplatz benötigt. Sie vereinfachen z.B. das Zusammenlegen voneinander unabhängigen Anwendungen, den Datenimport und -export, die Implementation von Spiegel / Replikation Strategien oder die Synchronisation mit externen Systemen.

Primärschlüssel setzen

Um einen Primärschlüssel im Struktureditor zu erstellen:

1. Wählen Sie das bzw. die Felder für den Primärschlüssel der Tabelle aus.
2. Klicken Sie mit der rechten Maustaste und wählen im Kontextmenü den Eintrag **Primärschlüssel erzeugen**. Dieser Menübefehl ist nur für Primärschlüssel passende Feldtypen aktiv:



Pro Tabelle ist immer nur ein Primärschlüssel möglich. Ist bereits ein Primärschlüssel angegeben, erscheint eine Fehlermeldung, dass zuerst der vorhandene Primärschlüssel deaktiviert wird.

Der Primärschlüssel wird sofort angelegt. Die im Primärschlüssel enthaltenen Felder erscheinen im Editor unterstrichen, die SQL Beschreibung zeigt das PRIMARY KEY Schlüsselwort.

Erstellen Sie einen Primärschlüssel in einer Tabelle, die bereits Datensätze enthält (in einem neuen oder vorhandenen Feld), prüft 4D die Konformität aller Werte oder führt automatisch Operationen zur Aktualisierung durch:

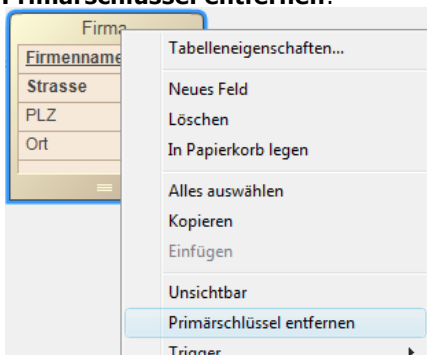
- Das bzw. die Felder, die zum Primärschlüssel gehören, dürfen keine doppelten Werte enthalten. Bei doppelten Werten in den Datensätzen der Tabelle erscheint eine Fehlermeldung, dass sich aus diesem Grund kein Primärschlüssel anlegen lässt.
- Sind die duplizierten Werte leer oder Nullwerte (bei hinzugefügtem Schlüsselfeld), können Sie die entsprechende Option zum automatischen Generieren prüfen: für Felder vom Typ Alpha in oder für Felder vom Typ Lange Ganzzahl. In diesem Fall prüft 4D alle Datensätze der Tabelle und weist jedem Primärschlüssel automatisch einen berechneten Wert zu. Beachten Sie, dass dieser Prozess sequentiell ist und bei einer größeren Anzahl Datensätze eine Zeit lang dauert.

Hinweis: Die Spalten im Primärschlüssel akzeptieren keine NULL Werte.

Primärschlüssel entfernen

Um einen Primärschlüssel aus einer Tabelle zu entfernen:

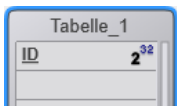
1. Klicken Sie mit der rechten Maustaste auf die Tabelle mit dem Primärschlüssel und wählen im Kontextmenü den Eintrag **Primärschlüssel entfernen**:



Auf dem Bildschirm erscheint ein Dialogfenster zum Bestätigen. Klicken Sie auf die Schaltfläche **OK**, um den Primärschlüssel zu entfernen.

Standardfeld mit Primärschlüssel

In 4D v14 erhält jede neu erstellte Tabelle in der Struktur standardmäßig einen Primärschlüssel:



Dieses Feld heißt standardmäßig "ID", ist vom Typ Lange Ganzzahl und hat folgende Attribute:

- Einmalig
- Eingabe von NULL Werten verweigern
- Autoincrement,
- Automatischer Index

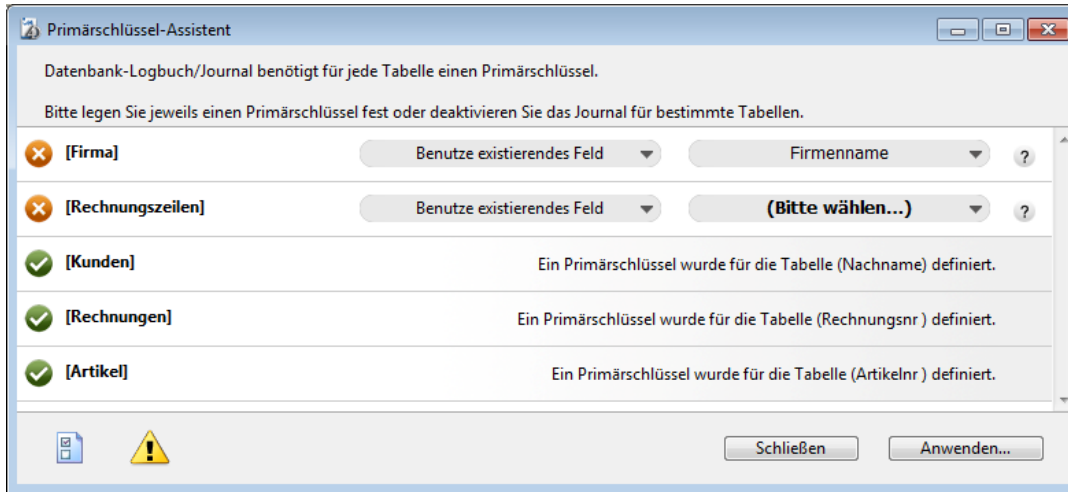
Hinweis: Auf der **Seite Struktur** der 4D Einstellungen können Sie Name und Typ der standardmäßig angelegten Primärschlüssel verändern.

Sie können dieses Feld so verwenden oder bei Bedarf Name bzw. Eigenschaften verändern, wenn Sie z.B. ein UUID Feld verwenden wollen. Sie können es auch löschen, um ein anderes Feld bzw. Felder als Primärschlüssel einzusetzen. Sie müssen jedoch in jeder 4D Tabelle mindestens einen Primärschlüssel beibehalten.

Hinweis: Tabellen, die in die Datenbank importiert werden oder mit dem Befehl **CREATE TABLE** erstellt wurden, erhalten keine standardmäßigen Primärschlüssel.

Primärschlüssel-Assistent

Der "Primärschlüssel-Assistent" dient dazu, Probleme, die bei Primärschlüsseln für Anwendungen mit Logbuch auftreten, rasch zu beheben:




Sie können folgendes ausführen:

- Für jede Tabelle prüfen, ob sie mit der Strategie zum Protokollieren in 4D kompatibel ist,
- Bei nicht-kompatiblen Tabellen einen Primärschlüssel festlegen.

Hinweis: Dieser Assistent ist auch als 4D v13 Komponente verfügbar, so dass Sie die Migration von 4D Anwendungen auf v14 vorbereiten können.

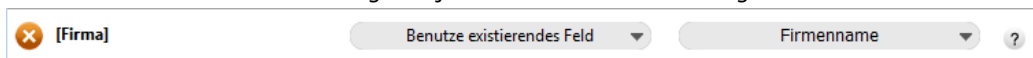
Assistent anzeigen

Es gibt zwei Wege, den Primärschlüssel-Assistenten anzuzeigen:

- Über den **Fehlerdialog zu Primärschlüssel**: Klicken Sie hier am unteren Rand auf die Schaltfläche "Assistent starten"
Hinweis: Der Assistent erscheint im Designmodus. Startet die Anwendung im Anwendungsmodus, wird der Assistent nicht direkt angezeigt. Dazu müssen Sie erst in den Designmodus wechseln.
- Über das **Struktureditor**: Klicken Sie in der Toolbar am oberen Rand auf die Schaltfläche .

Den Assistenten einsetzen

Der Primärschlüssel-Assistent zeigt für jede Tabelle der Anwendung eine Zeile an:

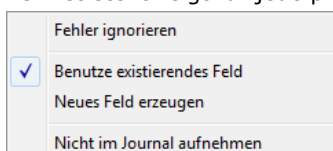



Hinweis: Der Assistent berücksichtigt keine Tabellen, die im Papierkorb liegen.

Das Icon zu Beginn jeder Zeile gibt an, ob die Tabelle passt oder verändert werden muss:

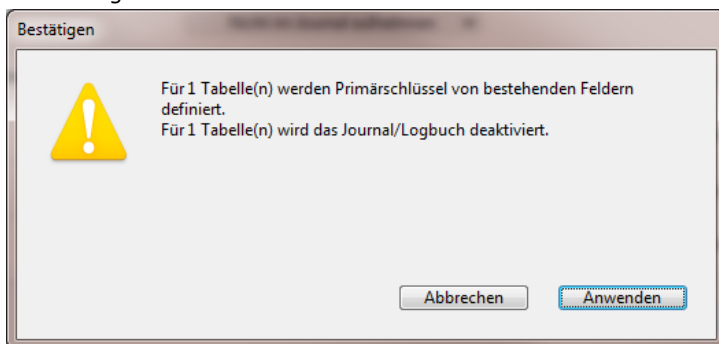
- **OK** Die Tabelle hat einen gültigen Primärschlüssel.
Kein
- **Primärschlüssel, auswählbare Feld(er)** Die Tabelle hat keinen Primärschlüssel, enthält aber mindestens ein Feld, das als Primärschlüssel geeignet ist.
Kein
- **Primärschlüssel, kein Feld auswählbar** Die Tabelle hat keinen Primärschlüssel und hat keine Felder, die sich dafür eignen. Sie müssen selbst ein Feld mit Primärschlüssel anlegen.
- **Warnung** Die Tabelle hat keinen Primärschlüssel, wird aber auch nicht protokolliert, d.h. die Option "Benutze Logbuch" ist für die Tabelle deaktiviert. Um Tabellen mit diesem Status auszublenden, können auf Sie auf das Icon für Warnungen klicken.

Der Assistent zeigt für jede protokollierte Tabelle ohne Primärschlüssel ein DropDown-Menü mit folgenden Optionen:

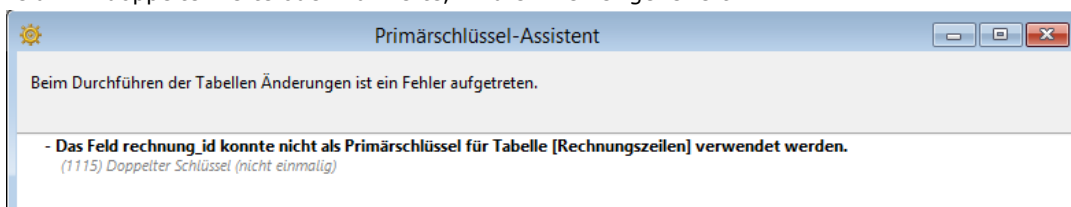


- **Fehler ignorieren** ändert die Tabelle nicht. Fehler werden nicht korrigiert und der Status der Tabelle ändert sich nicht. Wählen Sie diese Option, wenn Sie später eingreifen wollen oder den Primärschlüssel unabhängig vom Assistenten erstellen wollen. Diese Option ist z.B. erforderlich, wenn Sie in mehreren Feldern der Tabelle einen Primärschlüssel erzeugen wollen.
- **Benutze existierendes Feld** (erscheint nur für Tabellen, in denen mindestens ein Feld als Primärschlüssel wählbar ist) bestimmt ein Feld in der Tabelle als Primärschlüssel. Wählen Sie diese Option, schlägt der Assistent standardmäßig das am besten geeignete Feld vor. Wollen Sie ein anderes Feld verwenden oder bietet der Assistent kein passendes Feld an ("Bitte wählen" erscheint), klicken Sie auf das zweite DropDown Menü, um selbst ein geeignetes Feld zu definieren.
- **Neues Feld erzeugen** erstellt in der Tabelle ein neues Feld mit Primärschlüssel. Dieses Feld hat dieselben Attribute wie das Feld, das beim Anlegen einer neuen Tabelle mit Primärschlüssel angelegt wird. Weitere Informationen dazu finden Sie im Abschnitt **Standardfeld mit Primärschlüssel**). Standardmäßig wird ein Feld vom Typ Lange Ganzzahl mit Namen "ID" angelegt. Sie können Typ und Name dieses Standardfeldes in den 4D Einstellungen auf der **Seite Struktur** verändern. Klicken Sie dazu im Primärschlüssel-Assistenten am unteren Rand auf das Icon . Sie können die Standardvorgaben für das neue Feld auch direkt im Primärschlüssel-Assistenten verändern.
- **Nicht im Journal aufnehmen** deaktiviert die Option "In Journal aufnehmen" für die Tabelle. Wählen Sie diese Option für temporäre Tabellen (siehe Abschnitt **In Journal aufnehmen**). Bestätigen Sie dieses Dialogfenster, wird der Tabelle der Status "Warnung" zugewiesen.

Haben Sie Ihre Einstellungen festgelegt, können Sie am unteren Rand auf die Schaltfläche **Anwenden** klicken, um die Änderungen anzuwenden oder auf die Schaltfläche **Schließen**, um das Fenster ohne Änderungen zu schließen. Klicken Sie auf Anwenden, erscheint ein weiteres Dialogfenster, das die auszuführenden Operationen anzeigt. Sie können die Einstellungen anwenden oder abbrechen:



Haben Sie vorhandene Felder als Primärschlüssel bestimmt, prüft 4D jede Tabelle, um sicherzustellen, dass die vorhandenen Daten die notwendigen Attribute für diesen Datentyp (einmalig und Eingabe von Nullwerten verweigern) haben. Enthält ein Feld z.B. doppelte Werte oder Nullwerte, wird ein Fehler generiert:



Sie müssen erst diese Fehler finden und beheben, damit die Protokollierung der Daten im Journal möglich ist.

Hinweis: Um die Korrektur von Fehlern bei Primärschlüsseln zu vereinfachen, erscheinen die Werte der Primärschlüssel auf der **Seite Aktivitätsanalyse** des Maintenance- und Security Center.

Datenfelder erstellen und ändern

Sie müssen für jede Tabelle Datenfelder anlegen, welche die Daten aufnehmen, die Sie speichern und verwalten wollen. Wenn Sie ein Datenfeld erstellen, weisen Sie ihm einen Feldtyp zu, der die Art von Information beschreibt, die in diesem Datenfeld gespeichert werden. 4D verwendet den Feldtyp für verschiedene Operationen mit den Inhalten der Datenfelder. Enthält z.B. ein Datenfeld ein Datum, erstellen Sie es mit dem Feldtypen Datum. 4D kann somit Datumswerte berechnen, wie z.B. Dauer der Dienstzeit oder Qualifikation für Vergütungen. Zusätzlich kann 4D anhand der Daten in diesem Datenfeld Datensätze in chronologischer Reihenfolge sortieren. Weitere Informationen zu Feldtypen finden Sie im Abschnitt **4D Datenfeldtypen**.

Zusätzlich zum Feldtyp kann jedes Datenfeld einer Tabelle verschiedene Eigenschaften haben. Diese bestimmen die Konditionen für Eingabe, Anzeigen oder Ändern von Daten in den Datenfeldern. Weitere Informationen dazu finden Sie im Abschnitt **Feldeigenschaften**.

Sie können für ein Datenfeld nach dem Erstellen jederzeit den Feldtyp oder seine Eigenschaften ändern, oder es auch löschen.

Datenfeld anlegen

In einer Tabelle sind bis zu 32.767 Datenfelder möglich. Sie erstellen ein neues Datenfeld über die Standardbefehle oder per **Kopieren/Einsetzen**.

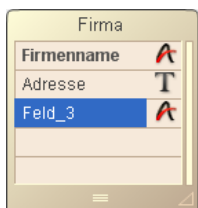
Um ein Datenfeld zu erstellen:

1. Klicken Sie bei gedrückter rechter Maustaste in den Tabellennamen und wählen im Kontextmenü den Eintrag **Neues Feld** bzw. **Feld anfügen** (erscheint bei Anklicken eines vorhandenen Feldes)
ODER
doppelklicken Sie in der Tabellendarstellung auf eine leere Zeile unterhalb des letzten Feldnamens
ODER
Wählen Sie die Tabelle und dann in der Werkzeugleiste des Struktureditors im DropDown-Menü **Hinzufügen** den Eintrag **Feld**:



2. (optional) Um rasch weitere Datenfelder hinzuzufügen, drücken Sie die **Zeilenschaltung** so oft wie erforderlich.

Das Feld wird angefügt und ausgewählt. Es erhält standardmäßig den Typ alphanumerisch und den Namen `Feld_X`, wobei X die aktuelle Feldnummer ist. Wurden Felder gelöscht, entspricht diese Nummer u.U. nicht der aktuellen Feldnummer in der Tabelle. Weitere Informationen dazu finden Sie im nachfolgenden Abschnitt "Nummern gelöschter Felder".



Sie können den Namen verändern und Feldeigenschaften zuweisen.

Um ein Datenfeld per Kopieren/Einsetzen anzulegen:

1. Wählen Sie das/die gewünschten Datenfelder aus und dann den Standardbefehl für Kopieren, entweder im Menü **Bearbeiten**, im Kontextmenü oder über die Tastenkombination **Strg+C** unter Windows, **Befehl+C** auf Mac OS.
2. Wählen Sie die Tabelle aus, wo Sie neue Felder hinzufügen möchten und dann den Standardbefehl für Einfügen, entweder im Menü **Bearbeiten**, im Kontextmenü oder über die Tastenkombination **Strg+V** unter Windows, **Befehl+V** auf Mac OS
TDas Feld wird mit dem zugewiesenen Typ und Eigenschaften dupliziert. Gibt es bereits ein gleichnamiges Feld in der Zieltabelle, erhält es den Namen `Kopie_von_Feldname`, wobei Feldname der Originalname des Feldes ist. Kopieren Sie dasselbe Feld mehrmals, wird zusätzlich eine Nummer angefügt.

Hinweis: Die Eigenschaft Indiziert wird nicht beibehalten, wenn Felder per Kopieren-Einsetzen angelegt werden.

Feld umbenennen

Sie können Feldnamen jederzeit ändern, 4D aktualisiert die Namen der Felder überall, wo sie vorkommen, d.h. in Formularen, Methoden und Dateien. Das gilt nicht in geöffneten Methoden. Sie müssen diese erst schließen und erneut öffnen, damit die Namen geändert werden.

Jeder Datenfeldname in einer Tabelle muss einmalig sein.

Sie können ein Feld direkt im Struktureditor und im Explorer auf der **Seite Tabellen** umbenennen oder die Inspektoralette öffnen.

Um ein Datenfeld direkt umzubenennen (**Struktureditor** oder **Seite Tabellen** des Explorer):

1. Doppelklicken Sie auf den Feldnamen bzw. klicken einmal auf ein bereits ausgewähltes Feld ODER (nur Struktureditor) wählen Sie das gewünschte Feld und drücken die **Zeilenschaltung**. Das Feld wechselt in den Bearbeitungsmodus.
2. Geben Sie den neuen Namen ein und klicken außerhalb des Feldes.

Um ein Datenfeld über die Inspektorpalette des Struktureditors umzubenennen:

1. Doppelklicken Sie auf den Feldnamen ODER
Klicken Sie mit der rechten Maustaste auf das Feld und wählen dann im Kontextmenü den Eintrag **Feldeigenschaften** ODER
Doppelklicken Sie im Explorer auf der Seite Tabellen auf den Feldnamen.
4D zeigt Name und Eigenschaften des gewählten Feldes im **Inspektorfenster**. Das Fenster gibt auch die Tabellen- und Feldnummer an.
Ist der Inspektor bereits geöffnet und zeigt die Eigenschaften eines anderen Datenfeldes, wählen Sie das gewünschte Feld aus. Der Inspektor zeigt dann die Eigenschaften dieses Feldes.
2. Geben Sie den neuen Namen in den Bereich „Name“ ein.
Er erscheint sofort im Datenfeld.

Weitere Informationen zu den Namensregeln finden Sie im Abschnitt **Namensregeln für Tabellen und Datenfelder**.

Reihenfolge der Datenfelder

In 4D können Sie die Reihenfolge der Datenfelder in der Tabelle verändern. Das ist in manchen Fällen hilfreich, um die Struktur übersichtlicher zu machen.

Beachten Sie, dass die Sortierung in den Tabellen im Struktureditor keine Auswirkung auf die Anzeige der Datenfelder in den anderen Editoren der Anwendung hat. 4D zeigt die Felder immer in der Reihenfolge, wie sie erstellt wurden, auch wenn der Befehl **Feld einfügen** benutzt wird. Davon ausgenommen ist die Anzeige im Explorer, hier erscheinen die Objekte in alphabetischer Reihenfolge.

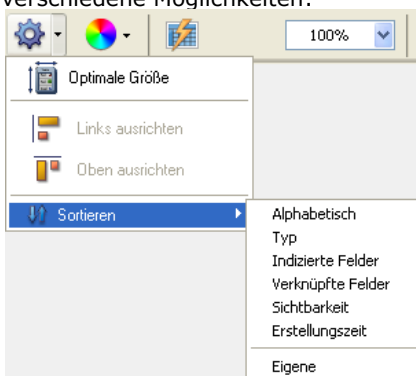
Es gibt zwei Möglichkeiten:

- Sie können die Felder innerhalb der Tabelle per Drag&Drop verschieben. Drücken Sie dazu unter Windows die **Alt-Taste**, auf Mac OS die **Wahl taste** — der Cursor wird zu einer Hand — und setzen Sie jedes Feld an die gewünschte Stelle:



Diese Sortierung wird gespeichert und lässt sich bei Bedarf wieder aufrufen. Dazu wählen Sie im Menü **Sortieren** den Befehl **Eigene** (siehe unten).

- Sortierkriterium anwenden: Das Untermenü **Sortieren** zur Schaltfläche in der Werkzeugleiste des Struktureditors bietet verschiedene Möglichkeiten:



Wählen Sie eine Option, wird sie auf alle ausgewählten Tabellen im Fenster angewandt.

Dieses Menü ist nur aktiv, wenn eine Tabelle ausgewählt wurde.

Ein Häkchen vor einem Menüeintrag gibt an, dass dieses Sortierkriterium aktuell angewendet ist. Sie können auch mehrere Kriterien gleichzeitig wählen.

Sie können zwischen folgenden Sortierkriterien wählen:

- **Alphabetisch**: Zeigt die Felder in alphabetischer Reihenfolge an.
- **Typ**: Zeigt die Felder nach Typ an, wobei die Namen alphabetisch sortiert sind.
- **Indizierte Felder**: Zeigt am Anfang die indizierten Felder an.
- **Verknüpfte Felder**: Zeigt zuerst die Primär-, dann die Fremdschlüsselfelder an.
- **Sichtbarkeit**: Zeigt am Anfang die sichtbaren Felder an.
- **Erstellungszeit**: Zeigt die Felder in der Reihenfolge ihrer Erstellung. Dies ist die Standardeinstellung.
- **Eigene**: Es wird keine spezifische Sortierung angewendet. Diese Option speichert die vorige Suche, die manuell per Drag&Drop ausgeführt wurde. Wurde zuvor kein Drag&Drop ausgeführt, hat diese Option keine Auswirkung.

Hinweis: Fügen Sie in einer Tabelle ein Datenfeld hinzu, wird es immer nach dem letzten Feld gesetzt, unabhängig vom aktuellen Sortierkriterium.

Datenfeld löschen

Sie können Datenfelder aus Ihren Tabellen löschen. Diese Aktion können Sie jedoch nicht mehr rückgängig machen. Die Operation lässt sich über die SQL-Engine von 4D oder über den Struktureditor ausführen. Weitere Informationen zu SQL finden Sie im Handbuch *4D SQL Reference*.

Um ein oder mehrere Datenfelder im Struktureditor zu löschen:

1. Markieren Sie das bzw. die entsprechenden Datenfelder
Sie können gleichzeitig Felder aus verschiedenen Tabellen auswählen.
2. Wählen Sie im Menü **Bearbeiten** den Befehl **Löschen**
ODER
Wählen Sie im Kontextmenü des Datenfeldes den Eintrag **Löschen**
ODER
Drücken Sie die **Löschen-** bzw. **Rückschritt-Taste**
Eine Meldung informiert Sie über die Folgen dieser Operation.
Klicken Sie auf die Schaltfläche **OK**, wird das bzw. die Felder aus der Struktur gelöscht. Alle damit verknüpften Daten werden unwiderruflich aus der Datendatei gelöscht.

Nummern gelöschter Felder

Wurde ein Datenfeld gelöscht, werden die anderen Felder der Datenbank nicht umnummeriert, damit die Stabilität der Datenbank gewahrt bleibt. So kann eine Tabelle z.B. vier Felder mit den Nummern 1, 4, 6 und 8 haben.

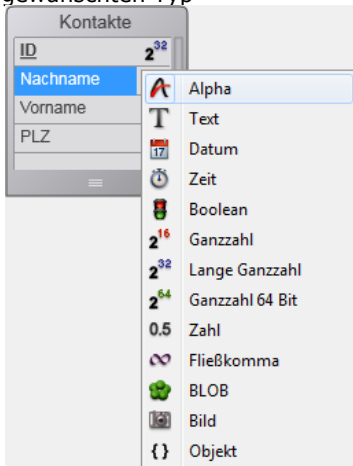
Beachten Sie, dass die Nummern dauerhaft gelöschter Felder beim Anlegen neuer Felder wieder verwendet werden.

4D Datenfeldtypen

Sie müssen für jedes Datenfeld einen Feldtyp festlegen. Der Feldtyp bestimmt, wie 4D die Daten in einem Feld bearbeitet und speichert und wie Sie Daten in Formularen eingeben oder anzeigen. Standardmäßig weist 4D jedem neuen Feld den Typ Alpha zu.

Um einen Feldtypen zu ändern:

1. Klicken Sie in der Tabellendarstellung auf das Icon des Feldtypen und wählen dann im Drop-Down-Menü den gewünschten Typ



ODER

Klicken Sie mit der rechten Maustaste in das entsprechende Feld, wählen im Kontextmenü den Eintrag **Typ** und dann den gewünschten Typ

ODER

Wählen Sie das entsprechende Feld und klicken dann im **Inspektorfenster** in den Bereich Typ, um die Liste der Feldtypen anzuzeigen.

Sie können den Typ des Datenfeldes jederzeit ändern, außer das Feld hat eine Verknüpfung oder gehört zu einem Hauptschlüssel. In diesem Fall ist das Menü zum Auswählen eines Feldtyps deaktiviert. Sie müssen die Verknüpfung oder den Hauptschlüssel explizit entfernen, bevor Sie den Typ ändern können.

Ändern Sie den Datentyp vor dem Eingeben von Daten in das Feld, ändert 4D einfach den Typ ab. Ändern Sie den Typ nach dem Eingeben von Daten in das Feld, konvertiert 4D die Daten - soweit möglich - beim ersten Laden der Daten nach Ändern in den neuen Typ. Daten aus einem Feld vom Typ Bild, das in einen anderen Typ konvertiert wurde, werden nicht angezeigt, ebensowenig wie Daten aus einem Feld, das in ein Feld vom Typ Bild konvertiert wurde.

Konvertieren Sie einen Feldtypen, behält 4D den ursprünglichen Wert des Feldes bei, bis Sie den Datensatz ändern. Enthält z.B. ein Feld vom Typ Text den Eintrag "über 10" und ändern Sie den Typ um in Zahl, zeigt das geänderte Feld "10" an.

Ändern Sie das Feld wieder zurück in ein Textfeld, ohne den Feldwert zu bearbeiten, zeigt 4D wieder "über 10" an.

4D unterstützt folgende Feldtypen:

- **Alpha**: Alphanumerischer Text zwischen 1 und 255 Zeichen
- **Text**: Text bis zu 2 GB
- **Datum**: Datumsangaben zwischen dem Jahr 100 und dem Jahr 32 767
- **Zeit**: Zeitangaben im Format Stunden:Minuten:Sekunden
- **Boolean**: Ein Datenfeld, das nur die Werte WAHR oder FALSCH annehmen kann
- **Ganzzahl**: Zahlen im Bereich von -32 768 bis 32 767
- **Lange Ganzzahl**: Zahlen im Bereich von plus oder minus 2.147.483.647
- **Ganzzahl 64 Bit (1)**: Zahlen mit 8 Bytes im Bereich von $\pm 2E64$
- **Zahl**: Zahlen mit Fließkomma im Bereich von $\pm 1,7E\pm 308$ (mit 13 signifikanten Stellen)
- **Fließkomma (1)**: Zahlen mit Fließkomma
- **BLOB** (Binary Large Object): in beliebiges binäres Objekt, wie z.B. eine Grafik, eine andere Anwendung oder ein Dokument
- **Bild**: Ein Bild im nativ von 4D unterstütztem Format (siehe **Native Unterstützung von Formaten** im Handbuch *4D Programmiersprache*)
- **Objekt**: Eigenschaft/Wert Paare in JSON Notation, bis zu 2 GB

(1) Achtung: Nur die SQL Engine von 4D verwendet diese Feldtypen. Bei Verwendung in der 4D Programmiersprache werden ihre Werte intern in den Typ Zahl umgewandelt.

Hinweis: Der generische Begriff "String" gibt den Typ Alpha oder Text an, "Zahl" gibt den Typ Ganzzahl, Lange Ganzzahl, Ganzzahl 64 Bit, Zahl oder Fließkomma an.

Alpha

Ein Alpha-Feld enthält alphanumerische Zeichen, d.h. Buchstaben und Zahlen, Interpunktionszeichen und besondere Zeichen wie z.B. Stern *, Prozentzeichen %, Bindestrich -, usw. Benutzen Sie ein Alpha-Feld für jede Information, die man als Text ansehen kann, mit einer max. Länge von 255 Zeichen.

Ein Alpha-Feld kann indiziert sein, ein Textfeld hingegen nicht. Weitere Informationen dazu finden Sie im Abschnitt **Indizes erstellen und ändern**.

Alpha ist der am meisten verwendete Feldtyp. Allgemein verwenden Sie dieses Datenfeld für Namen, Adressen, Telefonnummern, Postleitzahlen usw.. Bei der Eingabe akzeptiert ein Alpha-Feld alle Buchstaben, Zahlen, Interpunktionszeichen und Sonderzeichen.

Postleitzahlen werden aus zwei Gründen am besten in ein Alpha-Feld eingegeben: In numerischen Datenfeldern wird eine Null am Anfang nicht angezeigt und einige Postleitzahlen in Verbindung mit einem Auslandskennzeichen enthalten einen Bindestrich. Beim Abwägen zwischen einem numerischen oder einem alphanumerischen Datenfeld sollte in der Regel ein alphanumerisches Datenfeld gewählt werden, außer es wird für numerische Berechnungen verwendet oder für eine Suche oder Sortierung, die auf numerischen Werten basiert.

Sie können die Maximallänge eines Alpha-Feldes zwischen 1 und 255 Zeichen festlegen.

Sie können zwei oder mehr Alpha-Felder miteinander verketteten. Sie möchten z.B. den Vornamen und Nachnamen einer Person für ein Etikettenformular zusammensetzen. Hierzu verwenden Sie folgende Einzeilen-Methode:

```
VollerName:=[Angestellte]VorName+" "+[Angestellte)NachName
```

Die Variable *VollerName* kann angezeigt oder gedruckt werden. Sie können auch einen Teil der Information herausziehen, um ihn an anderer Stelle zu benutzen, z.B. Teil einer Zeichenkette. Die Teilmenge kann ebenfalls angezeigt oder gedruckt werden.

Text

Ein Textfeld ist ähnlich zum Alpha Feld, bis auf kleine Unterschiede.

Ein Textfeld kann bis zu 2 GB alphanumerische Zeichen enthalten. Im allgemeinen benutzen Sie ein Textfeld für umfangreiche Textblöcke, wie z.B. Kommentare oder Beschreibungen.

Aus Optimierungsgründen lässt sich der Inhalt eines Textfeldes außerhalb der Datensätze speichern (siehe Abschnitt "Im Datensatz speichern" unter **Feldeigenschaften**). In diesem Fall kann dem Textfeld kein Standardindex zugewiesen werden, ein Index mit Schlüsselwörtern ist jedoch möglich. Weitere Informationen dazu finden Sie im Abschnitt **Indizes erstellen und ändern**.

In einem Eingabeformular können Sie einem Textfeld Bildlaufleisten hinzufügen. In einem gedruckten Bericht kann ein Textfeld bei Bedarf erweitert werden, um alle Informationen zu drucken, auch wenn diese sich über mehrere Seiten erstrecken.

Bei einer Texteingabe sind die gängigen Funktionen der Textverarbeitung verfügbar: Sie können blättern, Zeilenumbruch im Anzeigebereich des Datenfeldes, doppelklicken, um ein Wort zu wählen, die Einfügemarke mit den Pfeiltasten verschieben und die Standardoperationen Ausschneiden, Kopieren und Einfügen ausführen. Hat ein Textfeld die Eigenschaft **mehrzeilig**, ist bei der Dateneingabe die Zeilenschaltung für neue Absätze erlaubt. Für ein Alpha-Feld ist das nicht möglich. Ist die Option **Mehrfachstil** zugewiesen, kann das Feld Text mit Formatierung enthalten, z.B. Wörter in Fettschrift, unterstrichen oder farbig.

Sie können Text in Textfelder einfügen, einschließlich Texte von Textverarbeitungsprogrammen.

Hinweis: Eine andere Möglichkeit zum Speichern und Bearbeiten von Text in einem Datensatz ist das Plug-In **4D Write**. Damit können Sie verschiedene Schrifteigenschaften zuweisen, Absätze formatieren und andere Möglichkeiten der Textverarbeitung einsetzen, die Ihnen in normalen Textfeldern nicht zur Verfügung stehen. Dazu muss, wie für 4D Plug-Ins üblich, der 4D Write Bereich in ein Feld vom Typ BLOB gesetzt werden — und nicht vom Typ Text. Weitere Informationen dazu finden Sie z.B. im Abschnitt **Get menu item** im Handbuch *4D Write Programmiersprache*.

Datum

Diesen Feldtyp verwenden Sie zum Speichern von Datumswerten, wie z.B. Startdatum, Verkaufsdatum, Geburtstage usw.. Ein Datumfeld kann jeden Datumswert, d.h. Tage, Monate, Jahre im Format DD/MM/JJJJ speichern, und zwar zwischen dem Jahr 100 und dem Jahr 32 767.

Hinweis: In den USA erfolgen Datumsangaben im Format Monat/Tag/Jahr, d.h. MM/DD/JJJJ. Andere Länder verwenden andere Formate, wie z.B. DD/MM/JJJJ im deutschen System und JJ/MM/DD im schwedischen System. Die Daten werden in dem Format angezeigt, das Sie im Betriebssystem Ihres Rechners festgelegt haben.

Zeit

Ein Zeitfeld benutzen Sie zum Verwalten von Zeiten, wie z.B. Aktuelle Zeit, Besprechungszeit, Berechnete Zeit usw.. Ein Zeitfeld speichert jeden Zeitwert, der im Format HH:MM:SS eingegeben wird.

Boolean

Boolean-Felder werden manchmal als logische Datenfelder bezeichnet und enthalten die Werte WAHR und FALSCH. Sie können ein Boolean-Feld entweder als ein Kontrollkästchen formatieren oder als ein Paar von Optionsfeldern. Ein markiertes Kontrollkästchen hat den Wert WAHR; deaktiviert ist dieser Wert FALSCH. Es wird entweder das erste Optionsfeld gewählt, dieses ist WAHR, oder das zweite, dieses ist FALSCH.

Sie sollten ein Boolean-Feld so bezeichnen, dass Sie die Frage stellen können: "Ist der Feldname wahr?" Diese Frage ist für eine Suche sinnvoll, weil 4D bei einer Suche nach WAHR- und FALSCH-Werten in einem Boolean-Feld sucht. Sie bezeichnen z.B. ein Feld mit "Männlich" anstatt mit "Geschlecht". Ihre Suchbedingung heißt dann "Männlich entspricht Wahr", anstatt "Geschlecht entspricht Wahr".

Ganzzahl

Diesen Feldtyp verwenden Sie für jedes Datenfeld, das ganze Zahlen speichert, d.h. Zahlen ohne Dezimalstellen, wie Datensatznummern, Rechnungsnummern usw.. Ganzzahlfelder können Zahlen von - 32 768 bis + 32 767 (2^{16}) enthalten.

Lange Ganzzahl

Diesen Feldtyp verwenden Sie für jedes Datenfeld, das ganze Zahlen speichert, die für ein Ganzzahlfeld zu groß sind. Diese Datenfelder enthalten ganze Zahlen, also ohne Dezimalstellen zwischen +/- 2.147.483.647 (2^{32}).

Ganzzahl 64 bit

Diesen Feldtyp verwenden Sie zum Speichern von Ganzzahlen mit 8 Bytes. Damit können Sie sehr große Zahlen im Bereich +/- 2^{64} speichern.

Hinweis: Beachten Sie, dass nur die SQL Engine von 4D die beiden Feldtypen Ganzzahl 64 bits und Fließkomma verwendet. Bei Verwendung in der 4D Programmiersprache werden sie intern in Werte vom Typ Zahl umgewandelt.

Zahl

Ein Feld vom Typ Zahl speichert Zahlen, d.h. Dezimalzahlen, wie Preise, Monatseinkommen, Kosten usw.. Felder vom Typ Zahl enthalten beliebige Zahlen im Bereich $\pm 1.7E\pm 308$.

Numerische Anzeigeformate berücksichtigen automatisch die landesspezifischen Parameter des Betriebssystems. 4D verwendet Komma und Punkt zur Anzeige von Tausend und Dezimal gemäß dem gewählten Betriebssystem.

Hinweis: In der 4D Datenbank Engine werden Vergleiche vom Typ Zahl immer mit einem *epsilon* Wert von 10^{-6} ausgeführt, um ausreichende Genauigkeit zu erhalten. Um die Konsistenz von Daten und Berechnungen sicherzustellen, lässt sich dieser *epsilon* Wert nicht verändern, d.h. die Datenbank Engine berücksichtigt nicht den Befehl **SET REAL COMPARISON LEVEL**. Er wird nur bei Bearbeitungen angewandt, die in der 4D Programmiersprache ausgeführt werden. Beachten Sie, dass Daten vom Typ Zahl nicht geeignet sind zum Speichern bzw. Berechnen von Werten mit einer großen Anzahl signifikanter Stellen, wie z.B. Seriennummern oder astronomische Messungen, da deren Genauigkeit von Natur aus begrenzt ist.

Fließkomma

Diesen Feldtyp verwenden Sie zum Speichern von Fließkommazahlen. Das sind Werte ohne Verlust der Genauigkeit.

Hinweis: Beachten Sie, dass nur die SQL Engine von 4D diesen Feldtyp verwendet. Bei Verwendung in der 4D Programmiersprache werden sie intern in Werte vom Typ Zahl umgewandelt.

BLOB (Binary Large Object)

Datenfelder für Blobs, d.h. Binary Large Object, oder Große Binäre Objekte speichern binäre Objekte jeder Art. Sie können z.B. Unterlagen speichern, die mit anderen Anwendungen erstellt wurden, gescannte Bilder oder andere Anwendungen. Ein Blob kann bis zu 2 Gigabyte Größe haben. Wenn Sie an einem Datensatz arbeiten, der ein Blob-Feld enthält, wird das gesamte Blob in den Speicher geladen. Sie können ein Blob-Feld auch benutzen, um ein ganzes Dokument in Ihrer Datenbank zu speichern. Sie können den Inhalt eines Blob-Feldes in ein Dokument schreiben. Sie können z.B. ein Blob-Feld in einem Verwaltungssystem für Dokumente benutzen, welches Dokumente in der Datenbank speichert und diese auf Anfrage Benutzern zur Verfügung stellt.

Blob Felder werden über die BLOB Befehle der 4D Programmiersprache verwaltet. Über die Befehle **DOCUMENT TO BLOB** und **BLOB TO DOCUMENT** können Sie Dokumente in Blob-Felder eingeben oder auslesen. Über die Befehle **COMPRESS BLOB**, **EXPAND BLOB** und **BLOB PROPERTIES** können Sie mit komprimierten Blobs arbeiten.

Zur Optimierung wird der Inhalt vom BLOB Feldern außerhalb der Datensätze gespeichert. BLOBS werden nur bei Bedarf geladen, z.B. wenn der Datensatz mit dem BLOB-Feld bei einer Suche gefunden wird. Da ein BLOB alle Arten von Daten enthalten kann, wird der Inhalt eines Blob-Feldes nicht auf dem Bildschirm angezeigt.

Bild

Bildfelder werden benutzt, um digitalisierte Fotos, Diagramme, Karten und mit grafischen Anwendungen erstellte Illustrationen zu speichern. Die Bilder bleiben in ihrem nativen Format erhalten. Einige grafische Anwendungen speichern zu Bildern zusätzliche Informationen mit Hinweisen für Ausgabegeräte wie z.B. einen PostScript™ Drucker, oder ab Version 12 Metadaten. Diese Information wird mitgeliefert, wenn das Bild kopiert oder in ein Bildfeld eingefügt wird. 4D verwendet diese Angaben zum Drucken des Bildes auf einem passenden Ausgabegerät oder bei Metadaten über die Befehle **GET PICTURE METADATA** und **SET PICTURE METADATA**.

Zur Optimierung wird der Inhalt vom Bildfeldern außerhalb der Datensätze gespeichert. Bilder werden nur bei Bedarf geladen, z.B. wenn der Datensatz mit dem Bildfeld bei einer Suche gefunden wird. Ab Version 13 können Sie Bilder auch außerhalb der Datendatei speichern. Weitere Informationen dazu finden Sie im Abschnitt **Daten extern speichern**.

Standardnamen von Bilddateien

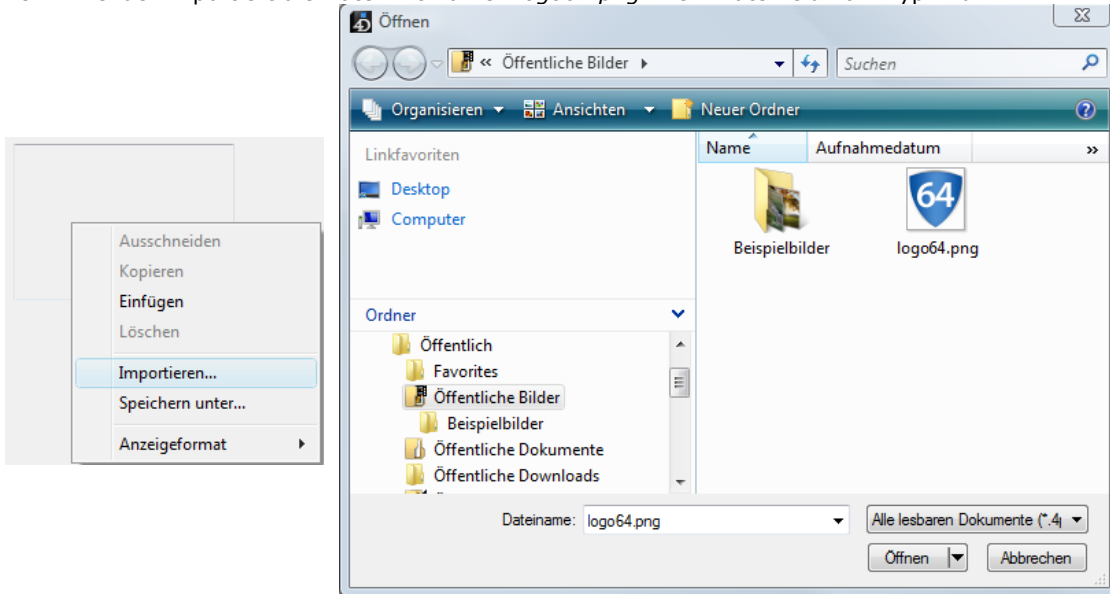
4D kann jetzt für jedes Bild, das einem Datenfeld zugeordnet ist, einen Standardnamen speichern. Folglich können Sie einen standardmäßigen Dateinamen setzen, wenn Sie den Inhalt eines Datenfeldes vom Typ Bild auf die Festplatte schreiben, und zwar durch Export eines Benutzers oder über den Befehl **WRITE PICTURE FILE**, wenn Sie im Parameter *DateiName* einen leeren String übergeben. Wird der Inhalt des Feldes in eine Variable oder in ein anderes Feld kopiert, wird der Standardname ebenfalls kopiert..

Es gibt zwei Möglichkeiten, einem Bild, das in einem Datenfeld vom Typ Bild gespeichert ist, einen Standardnamen zuzuweisen:

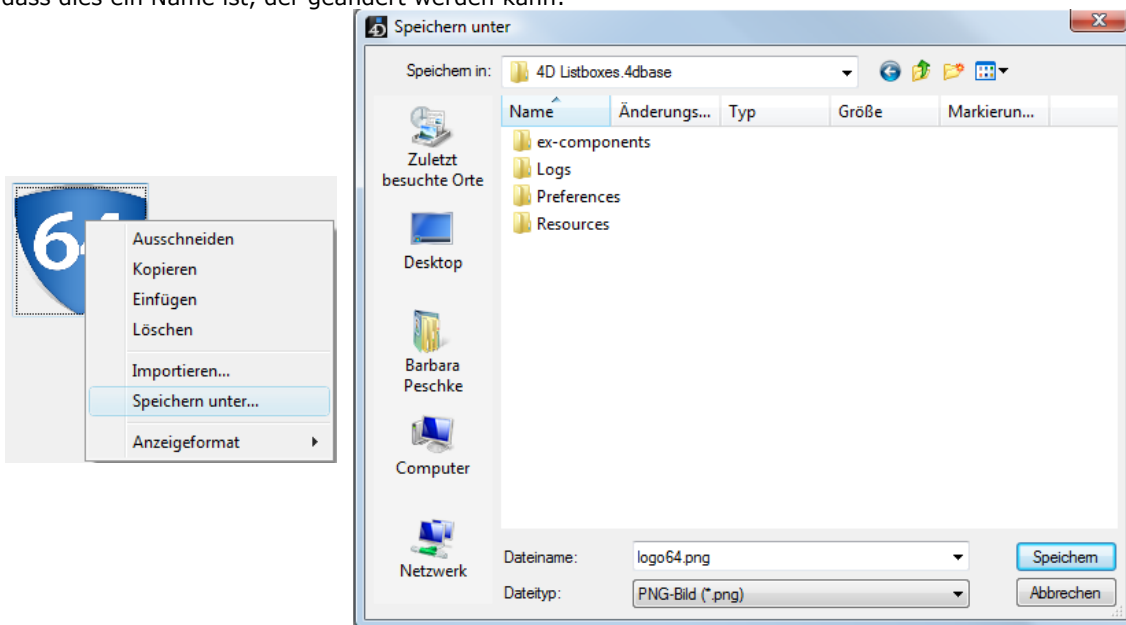
- Per Programmierung über den Befehl **SET PICTURE FILE NAME**. Damit weisen Sie dem Bild einen standardmäßigen Dateinamen zu. Über den Befehl **Get picture file name** erhalten Sie den Standardnamen eines Bildes.
- Automatisch, wenn der Inhalt einer Bilddatei in ein Datenfeld vom Typ Bild importiert wird; entweder über das Kontextmenü oder über den Befehl **READ PICTURE FILE**. In diesem Fall speichert 4D den Namen der Bilddatei.

Sehen Sie hierzu folgendes Beispiel::

1. Der Anwender importiert die Datei mit Namen *logo64.png* in ein Datenfeld vom Typ Bild:



2. Dann sichert der Anwender den Inhalt der Bilddatei: Der Name *logo64.png* erscheint im Dialogfenster. Beachten Sie, dass dies ein Name ist, der geändert werden kann:



Objekt

4D Objektfelder speichern verschiedene Arten von Eigenschaft/Wert Paaren ohne vordefiniertes Datenschema. Die gespeicherte Datenstruktur zwischen verschiedenen Datensätzen ist nicht zwingend dieselbe. So kann z.B. ein Objektfeld [Person]Adresse je nach Stadt, Land, etc. verschiedene Eigenschaften haben:

```
record1 = {"street1":"Cotton Treasure Grounds", "street2":"Place Corners", "state":"MD", ...} record2 = {"street1":"Umber Road", "Number":"28", "state":"MO", ...}
```

Die Struktur von 4D Objekten basiert auf dem klassischen Prinzip von "Eigenschaft/Wert" Paaren. Die Syntax dieser Objekte basiert auf JSON, folgt ihr aber nicht komplett.

- Ein Eigenschaftname ist immer ein Text, z.B. "Name" (bis zu 255 Zeichen, unterscheidet zwischen Groß- und Kleinschreibung).
- Der Wert einer Eigenschaft kann einer der folgenden Typen sein:
 - Zahl (Zahl, Ganzzahl, etc.)
 - Text
 - Null
 - Boolean
 - Datum (Datumstyp oder ISO Datumsformat - siehe **Seite Kompatibilität**, Option "Verwende Datumstyp statt ISO Datumsformat in Objekten")

- Zeit (gespeichert als Anzahl von Sekunden)
- Objekt (Objekte können in mehrere Stufen verschachtelt sein)
- Bild (*)
- Collection

Ein Objektfeld kann bis zu 2 Gigabyte groß sein. Arbeiten Sie mit einem Datensatz, der ein Objektfeld enthält, wird das gesamte Objekt in den Speicher geladen. Objektfelder lassen sich wie Felder vom Typ Text, Bild oder BLOB in der Datendatei speichern (innerhalb oder außerhalb des Datensatzes) oder außerhalb der Datendatei; weitere Informationen dazu finden Sie im Abschnitt **Daten extern speichern**.

Zum Verwalten von Objektfeldern können Sie die Objektnotation verwenden (siehe **Objektnotation verwenden**) oder Befehle aus dem Kapitel **Objekte (Sprache)**, wie z.B. **OB Get** und **OB SET**.

Mit spezifischen Befehlen wie **QUERY BY ATTRIBUTE**, **QUERY SELECTION BY ATTRIBUTE**, **DISTINCT ATTRIBUTE VALUES** oder **DISTINCT ATTRIBUTE PATHS** können Sie Suchen und Bearbeitungen in Objektfeldern ausführen.

Da Objektfelder auf Text basieren, wird der Inhalt eines Objektfeldes in einem 4D Formular standardmäßig als Text angezeigt und in JSON formatiert.

(*) Wenn als Text im Debugger dargestellt oder in JSON exportiert, zeigen die Eigenschaften des Objekts Bild "[Objekt Bild]" an. Beachten Sie, dass danach Sichern des Datensatzes den String "[Objekt Bild]" in der Eigenschaft sichert.

Hinweis: Zum Arbeiten mit JSON Objekten können Sie die Befehle im Kapitel **JSON** verwenden.

Warum Felder vom Typ Objekt verwenden?

Mit Datenfeldern vom Typ Objekt lassen sich dynamische Felder ohne Schema definieren, d.h. sie sind sozusagen benutzerdefinierte bzw. eigene Felder. In 4D haben Sie die Wahl zwischen einem Datenmodell mit oder ohne Schema. Beide Fälle erlauben rasche indizierte Suchen.

Darüberhinaus können Objektfelder standardmäßige Datenmodelle vereinfachen. So sorgt z.B. ein Objektfeld in einer klassischen Tabelle "Kontakte" dafür, dass nicht dutzende Felder mit allen möglichen Werten angelegt werden - und in 90% der Fälle meist unbenutzt bleiben. Das Informationsmodell wird von Fall zu Fall on-the-fly erstellt.

Werte von Objektfeldern setzen oder erhalten

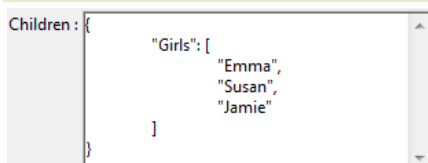
Werte von Objektfeldern werden wie standardmäßige Objekte der Programmiersprache über Befehle aus dem Kapitel **Objekte (Sprache)** oder über Objektnotation (siehe **Objektnotation verwenden**) verwaltet. Zum Beispiel:

```
// Um einen Wert zu setzen
OB SET([Persons]Identity_OB;"First Name";$firstName)
OB SET([Persons]Identity_OB;"Last Name";$lastName)

// Um einen Wert zu erhalten
$firstName:=OB Get([Persons]Identity_OB;"First Name")
$lastName:=OB Get([Persons]Identity_OB;"Last Name")
```

Arrays werden ebenfalls unterstützt, zum Beispiel:

```
ARRAY TEXT($arrGirls;3)
$arrGirls{1}:="Emma"
$arrGirls{2}:="Susan"
$arrGirls{3}:="Jamie"
OB SET ARRAY([Persons]Children;"Girls";$arrGirls)
```



Objektfelder sichern

Um Änderungen an Attributen eines Objektfeldes zu sichern, müssen Sie die Änderung in den meisten Fällen 4D vor dem Sichern des Datensatzes explizit mitteilen. Dazu weisen Sie das Objektfeld erneut sich selbst zu:

```
[Persons]Identity_OB:=[Persons]Identity_OB //Updaten des Feldinhalts erzwingen
SAVE RECORD([Persons])
```

Dieser Schritt ist notwendig, da eine Objektreferenz an verschiedenen Stellen in der Anwendung und auch in anderen Objektfeldern verwendet werden kann. Beim Sichern des Datensatzes kann die 4D Programmiersprache nicht erkennen, ob ein Attribut des Objektfeldes geändert wurde.

In folgenden Fällen müssen Sie das Feld explizit neu zuweisen, um eine Änderung des Attributs direkt zu sichern:

- Wenn Sie Attribute über Objektnotation verwalten:

```
[Person]Info.firstName:="Jane"
[Person]Info:=[Person]Info //zwingend, um die Bearbeitung zu sichern
SAVE RECORD([Person])
```

- Wenn Sie Attribute, die nicht auf der ersten Ebene liegen, mit dem Befehl **OB SET** verwalten:

```
OB SET([Person]Info.Children[0];"firstName";"Jenny")
[Person]Info:=[Person]Info //zwingend, um die Bearbeitung zu sichern
SAVE RECORD([Person])
```

Hinweis: Die erneute Zuweisung entfällt, wenn Attribute, die auf der ersten Ebene liegen, mit dem Befehl **OB SET** bearbeitet werden:

```
OB SET([Rect]Desc;"x";"50";"y";"50";"color";"blue") //Auf Attribute auf der ersten Ebene zugreifen
SAVE RECORD([Rect]) //In diesem Fall muss das Feld nicht zugewiesen werden.
```

Formeln mit Objektfeldern

Objektfelder lassen sich in Formeln verwenden. Dazu verwenden Sie den standardmäßigen Formeleditor oder den Befehl **EXECUTE FORMULA**. In diesem Kontext lassen sich Objektfelder dann aber nur über folgende Funktionen verwalten:

- **OB Get**
- **OB Is empty**
- **OB Is defined**
- **OB Get type**

Sie können z.B. folgende Suchformel ausführen:

```
OB Get([Rect]Desc;"color")="blue"
```

Objektfelder in Formularen anzeigen und eingeben

Objektfelder werden standardmäßig als Textfelder in 4D Formularen angezeigt. In diesen Bereichen müssen Objektdaten entweder *undefiniert* oder als JSON Text definiert sein, andernfalls wird ein Fehler zurückgegeben.

Haben Sie z.B. das Feld [Rect]Desc als Objektfeld definiert, können Sie schreiben:

```
CREATE RECORD([Rect])
[Rect]Name:="Blue square"
OB SET([Rect]Desc;"x";"50";"y";"50";"color";"blue")
SAVE RECORD([Rect])
```

Wird das [Rect]Desc Feld in Ihr Formular eingebunden, erscheint folgender Inhalt:

Name: Blue square

Desc: {
 "x": "50",
 "y": "50",
 "color": "blue"
 }

Sie können angezeigte Werte direkt im Textfeld anzeigen oder Objektdaten direkt mit der standardmäßigen Objekt Notation eingeben. Drücken Sie die **Tabulatortaste**, werden sie direkt in JSON formatiert:

Name: Pink rect

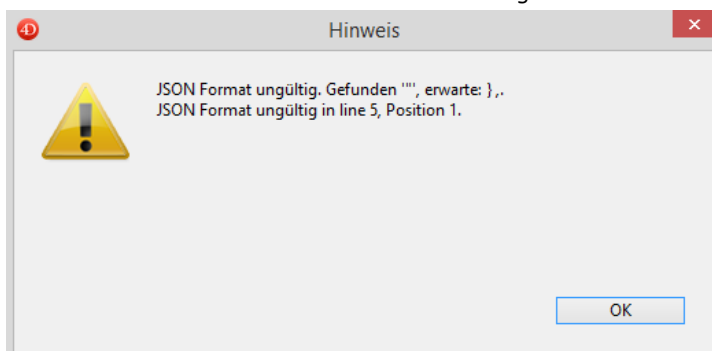
Desc: {x:100,y:1000,color:pink}

[Tab]

Name: Pink rect

Desc: {
 "x": 100,
 "y": 1000,
 "color": "pink"
 }

Machen Sie direkte Eingaben jedoch mit Bedacht, da fehlende oder falsch gesetzte Zeichen in JSON zu einem Fehler beim Parsen führt und die bearbeiteten Daten nicht gesichert werden:



Es ist in der Regel präziser, Inhalt von Objektfeldern über Befehle der Kapitel **Objekte (Sprache)** und **JSON** zu verwalten.

Aktuelle Einschränkungen

Die meisten der standardmäßigen 4D Funktionalitäten unterstützen Objektfelder genauso wie andere Datentypen. Einige fortgeschrittene Bereiche des Programms sind dagegen im aktuellen Release noch nicht für Objektfelder verfügbar. Diese

Bereiche werden schrittweise erweitert und sind dann in kommenden Releases auch für Objektfelder verfügbar.

Teilweise Unterstützung durch 4D Formeln

Folgende Features oder Befehle unterstützen Objektfelder teilweise über 4D Formeln:

Feature/Befehl

Sucheditor

Sortiereditor

Import-/Exporteditor (automatische Unterstützung für 4D Anwendungsformat, Textformate benötigen Formulare)

4D Write

4D Write Pro

4D View

4D Tags

PROCESS 4D TAGS (mit Objektfeldern außerhalb von 4D Formeln)

Keine Unterstützung

Folgende 4D Features bzw. Befehle unterstützen Objektfelder nicht:

Feature/Befehl

Schnellberichteditor

Etiketteneditor

Diagramme

PHP

SDK Plugin

SQL Data Definition Language (CREATE TABLE)

SQL Data Manipulation Language (SELECT, INSERT, UPDATE)

SQL EXPORT DATABASE und **SQL EXPORT SELECTION**

DISTINCT VALUES

RELATE ONE

RELATE MANY

ORDER BY

SCAN INDEX

Zusätzlich zu Feldname und -typ (siehe **4D Datenfeldtypen**) bestimmen Feldeigenschaften die Darstellung sowie Konditionen für Eingabe, Anzeigen, Ändern und Speichern von Daten. Jedes Datenfeld lässt sich individuell einstellen. Sie stellen die Feldeigenschaften im **Inspektorfenster** ein. Für bestimmte Eigenschaften können Sie auch das Kontextmenü verwenden. Es erscheint, wenn Sie mit der rechten Maustaste auf ein Datenfeld klicken.

Hinweis: Sie können die Eigenschaften eines Datenfeldes ändern, für das bereits Daten eingegeben wurden. Bestimmte Änderungen berücksichtigen vorhandene Daten. Wählen Sie z.B. das Attribut **Einmalig**, zeigt 4D eine Meldung und erlaubt keine Aktivierung dieses Attributs, wenn Daten in diesem Feld doppelte Werte enthalten. Ändern Sie dagegen Eigenschaften der Dateneingabe, beeinträchtigt das nicht bereits vorhandene Daten. Wählen Sie z.B. das Attribut **Zwingend**, werden nur Eingaben nach dieser Änderung geprüft; vorher eingegebene Daten können leere Felder enthalten.

Definition


Der Bereich "Definition" des **Inspektorfenster** konfiguriert die grundlegenden Eigenschaften für das Feld. Einige Eigenschaften sind nur für bestimmte Feldtypen verfügbar.

Farbe

Sie können jedem Feld eine Farbe zuweisen. Mit Farben können Sie Felder nach ihren Attributen oder ihrem Zweck klassifizieren. Sie können z.B. einmalige Felder oder zwingende Felder jeweils in einer anderen Farbe kennzeichnen. Sie können auch jeder Tabelle (siehe **Farbe der Tabellendarstellung**) und jeder Verknüpfung (siehe **Definition**) eine eigene Farbe zuweisen.

Hinweis: Die Farbe für Felder im Struktureditor hat keine Auswirkung auf die Farbe von Feldern in Formularen (siehe **Farben für Hintergrund und Rahmen**).

Die Farbe für ein Feld wird auf den Feldnamen angewandt. Um eine Farbe zu setzen, markieren Sie das bzw. die gewünschten Datenfelder und wählen eine Farbe auf folgende Weise:

- In der Werkzeugpalette des Editors mit der Schaltfläche **Farbe** 
- Im Kontextmenü der Felder mit dem Eintrag **Farbe**
- Mit der Option **Farbe** im **Inspektorfenster**.

Über die Option **Automatik** können Sie wieder zur ursprünglichen Farbe des Datenfeldes zurückgehen.

Unsichtbar

Sie können ein Datenfeld für den Anwendungsmodus und für Plug-Ins ausblenden, wenn Sie für dieses Datenfeld die Eigenschaft **Unsichtbar** wählen. Die Eigenschaft **Unsichtbar** blendet das Datenfeld für den Benutzer aus. Ein Datenfeld mit dieser Eigenschaft erscheint nicht in den 4D Standard-Editoren und Dialogfenstern des Anwendungsmodus. Davon sind folgende Editoren und Dialogfenster betroffen:

- Alle Sucheditoren (siehe **Datensätze suchen**)
- **Sortiereditor**
- **Etikketteneditor**
- **Schnellberichteditor**
- Dialogfenster für **Daten importieren und exportieren**
- **Formeleditor**

An allen diesen Stellen kann der Benutzer das Datenfeld weder sehen noch auswählen. Er kann z.B. ein ausgeblendetes Datenfeld nicht in einem mit dem Schnellberichteditor erstellten Bericht verwenden.

Hinweis: Wenn Benutzer diese Editoren einsetzen, können sie ihre Spezifikationen in Dateien auf der Platte speichern, z.B. Kriterien einer Suche oder Sortierung. Dafür werden auch unsichtbare Datenfelder verwendet. Außerdem können Benutzer die Namen ausgeblendeter Datenfelder in das Dialogfenster **Formeleditor** schreiben.

Ausgeblendete Datenfelder werden im Fenster des Struktureditors kursiv dargestellt.

Einmalig

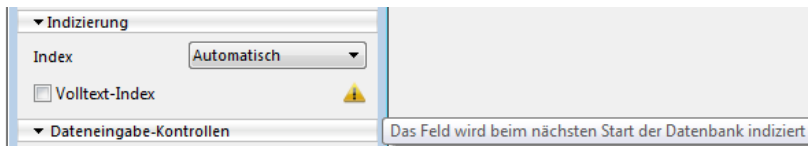
Die Eigenschaft **Einmalig** benutzen Sie, wenn Sie sicher sein wollen, dass jeder Datensatz einen unterschiedlichen, einmaligen Wert in diesem Datenfeld hat. **Einmalig** sollte für das Datenfeld definiert werden, welches jeden Datensatz in der Tabelle eindeutig kennzeichnet. Diese Eigenschaft ist sinnvoll für Datenfelder, in denen Personalnummer, Sozialversicherungsnummer und Auftragsnummer o.ä. gespeichert werden.

Einmalig verhindert, dass leere Werte dupliziert bzw. aktuelle Werte doppelt eingegeben werden. Ein leeres Datenfeld kann nicht in einen anderen Datensatz dupliziert werden.

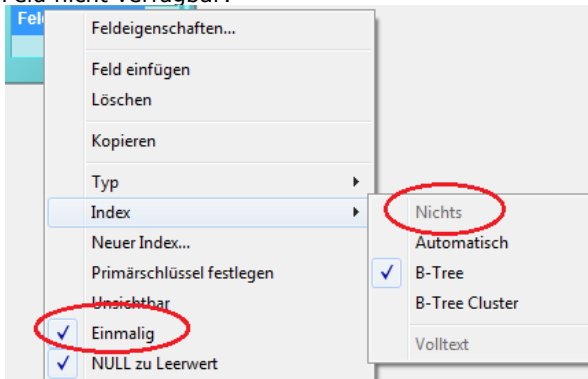
Die Informationsleiste im Struktureditor zeigt an, ob die Eigenschaft **Einmalig** für ein Datenfeld ausgewählt ist. Weitere Informationen dazu finden Sie im Abschnitt **Struktureditor**.

In 4D müssen Felder mit der Eigenschaft **Einmalig** indiziert werden:

- Wählen Sie für ein Feld die Eigenschaft **Einmalig**, zeigt ein Icon im Inspektorfenster an, dass ein Index erstellt wird (falls noch keiner existiert):



- Der Index für ein einmaliges Feld lässt sich nicht entfernen, d.h. Optionen wie **Nichts** oder **Entfernen** sind für dieses Feld nicht verfügbar:



Wollen Sie den Index zu einem einmaligen Feld entfernen, müssen Sie zuvor die Eigenschaft **Einmalig** deaktivieren.

Mit 4D Mobile Service veröffentlichen

Diese Option steuert, ob das Feld für 4D Mobile Anfragen zugänglich ist, die via REST an die 4D Datenbank gesendet werden. Standardmäßig sind alle Tabellen und Felder in 4D Mobile veröffentlicht. Aus Sicherheitsgründen wollen Sie u.U. bestimmte Felder in Ihren Tabellen nicht über 4D Mobile veröffentlichen, wie z.B. das Feld [Angestellte]Gehalt.

Dazu deaktivieren Sie die Option **Mit 4D Mobile Service veröffentlichen** für das entsprechende Feld.

Hinweis: Sie können diese Option auch für eine ganze Tabelle setzen, siehe **Tabelleneigenschaften**.

Beachten Sie, dass ein Feld über 4D Mobile nur zugänglich ist, wenn das auch für die übergeordnete Tabelle gilt. Ist sie nicht veröffentlicht, gilt das auch für alle dazugehörigen Felder, unabhängig welchen Status sie haben. So können Sie die 4D Mobile Veröffentlichung für eine Tabelle je nach Bedarf aktivieren/deaktivieren und die Einstellung der einzelnen Felder unverändert lassen.

Diese Option lässt sich im Rahmen der "4D Mobile" Funktionalität verwenden. Weitere Informationen dazu finden Sie im Handbuch **4D Mobile**.

Eingabe von NULL verweigern

Das Attribut **Eingabe von NULL verweigern** verhindert das Speichern von NULL Werten.

Ist diese Option für ein Datenfeld markiert, lässt sich hier kein NULL Wert speichern. Erhält ein Feld mit diesem Attribut einen NULL Wert, wird ein Fehler generiert.

Diese low-level Eigenschaft entspricht exakt dem Attribut NOT NULL von SQL. Wollen Sie in Ihrer 4D Datenbank die Möglichkeit haben, NULL Werte zu verwenden, empfehlen wir, dafür ausschließlich die SQL Sprache von 4D einzusetzen.

Hinweis: In 4D können Datenfelder auch das Attribut „Zwingend“ haben (siehe unten). Beide Konzepte sind ähnlich, ihre Reichweite ist jedoch unterschiedlich: Das Attribut "Zwingend" dient zur Kontrolle bei der Dateneingabe, während das Attribut "Eingabe von NULL Werten verweigern" auf Engine-Ebene der Datenbank arbeitet.

NULL Werten leere Werte zuordnen

Die Eigenschaft **NULL Werten leere Werte zuordnen** sorgt dafür, dass 4D jeden Nullwert, der im Datenfeld auftritt, als standardmäßigen 4D Leerwert betrachtet. Diese Option ist standardmäßig markiert.

Aus Kompatibilitätsgründen werden NULL Werte, die in Tabellen einer 4D Datenbank gespeichert sind, beim Verwalten mit der 4D Programmiersprache automatisch in Standardwerte konvertiert. Nehmen wir z.B. folgende Anweisung:

```
myAlphavar:=[mytable]MyAlphafield
```

... enthält das Datenfeld MyAlphafield einen Wert NULL, enthält die Variable myAlphavar "" (Leerstring).

Die Standardwerte richten sich nach dem Datentyp:

- Für Daten vom Typ Alpha und Text: ""
- Für Daten vom Typ Zahl, Ganzzahl und Lange Ganzzahl: 0
- Für Daten vom Typ Datum: "00.00.00"
- Für Daten vom Typ Zeit: "00:00:00"
- Für Daten vom Typ Boolean: Falsch
- Für Daten vom Typ Bild: Leeres Bild
- Für Daten vom Typ Blob: Leeres BLOB

Andererseits greift diese Funktionsweise in der Regel nicht bei Operationen auf Ebene der 4D Datenbank-Engine, z.B. Suchläufe. Eine Suche nach einem „leeren“ Wert, z.B. MeinWert=0, findet keine Datensätze, die den Wert NULL enthalten, und umgekehrt. Sind beide Werte, also Standard- und NULL Wert in den Datensätzen für dasselbe Datenfeld vorhanden, kann das bestimmte Operationen beeinträchtigen oder zusätzlichen Code erfordern.

Um diese Unstimmigkeiten zu vermeiden, können Sie die Option **NULL Werten leere Werte zuordnen** wählen, die das Steuern in der Programmiersprache von 4D standardisiert. Datenfelder mit NULL Werten werden dann systematisch als Feld mit Standardwerten betrachtet.

Diese Eigenschaft wird auf einer sehr niedrigen Ebene der Datenbank Engine berücksichtigt. Sie arbeitet hauptsächlich mit der 4D Funktion **Is field value Null**.

Hinweis: Alphanumerische Felder im UUID Format, die nicht generiert werden, werden nicht als NULL gewertet (**UUID Format**).

Autoincrement

Die Eigenschaft **Autoincrement** ermöglicht, für Datensätze einmalige Kennnummern zu erzeugen. Diese Eigenschaft ist für Datenfelder vom Typ Ganzzahl, Lange Ganzzahl und Ganzzahl 64 bit möglich. Weitere Informationen dazu finden Sie im Abschnitt **4D Datenfeldtypen**.

Ein Datenfeld mit dieser Eigenschaft zählt automatisch weiter, wenn ein Datensatz in der Tabelle angelegt wird. Die dem Feld zugewiesene Nummer passt zum "internen Zähler" der Tabelle, den 4D verwaltet. Auf diesen internen Zähler lässt sich über die 4D Befehle **Sequence number** und **Get database parameter** zugreifen, sowie über den Marker #N. Weitere Informationen dazu finden Sie im Abschnitt **Standardwerte**.

Die zugewiesenen Nummern sind dauerhaft und werden nie wiederverwendet, selbst wenn Datensätze gelöscht werden. Nummern, die während einer Transaktion erzeugt und dann wieder gelöscht werden, gehen verloren.

Es ist jedoch möglich, den internen Zähler einer Tabelle über den 4D Befehl **SET DATABASE PARAMETER** wiederherzustellen.

Diese Eigenschaft entspricht dem SQL Attribut SQL AUTO_INCREMENT und lässt sich mit dieser Programmiersprache spezifizieren (siehe Beispiel unten). Für Felder mit diesem Attribut erscheint die Bezeichnung AUTOINCREMENT auch im Bereich "SQL" des Inspektorfensters

Der folgende Code erzeugt in der Tabelle "Tiere" ein "id" Feld vom Typ Ganzzahl:

```
ARRAY TEXT($names;6)
$names{1}:="Hund"
$names{2}:="Katze"
$names{3}:="Pinguin"
$names{4}:="Schlange"
$names{5}:="Schmetterling"
$names{6}:="Maus"
Begin SQL
CREATE TABLE Tiere(
  id INT NOT NULL AUTO_INCREMENT ,
  name VARCHAR(30)NOT NULL,
  PRIMARY KEY(id));
INSERT INTO Tiere (name)VALUES(:$names);
End SQL
```

Der folgende Code:

```
SELECT * FROM Tiere;
```

... gibt diese Werte zurück:

id	name
1	Hund
2	Katze
3	Pinguin
4	Schlange
5	Schmetterling
6	Maus

Speichern im Datensatz, in der Datendatei, außerhalb der Datendatei

Diese Option ist nur für Datenfelder vom Typ Text, BLOB, Bild und Objekt verfügbar. Sie können für Daten dieser Felder einen Speicherort setzen. Es gibt folgende Optionen:

- **Im Datensatz:** Die Daten werden mit jedem Datensatz gespeichert. Im allgemeinen sollten Sie diese Option nicht für Felder mit umfangreichen Daten wählen. Diese Option ist jedoch für Textfelder standardmäßig gewählt, da diese Speicherart bei Verwenden von Standard B-Tree-Indizes erforderlich ist. Das Menü zum Wählen des Index ist ausgeblendet, wenn Textdaten außerhalb von Datensätzen gespeichert werden.
- **In der Datendatei:** Die Daten werden innerhalb der Datendatei, aber außerhalb des Datensatzes gespeichert. Diese Option ist für Felder vom Typ BLOB, Bild und Objekt standardmäßig gewählt. Das optimiert die Arbeitsweise der Anwendung bei umfangreichen Daten, da Texte, Bilder, BLOBs und Objekte außerhalb der Datensätze gespeichert werden.
- **Außerhalb der Datendatei:** Die Daten werden in separaten Dateien außerhalb der .4DD Datei gespeichert. Weitere Informationen dazu finden Sie im Abschnitt **Daten extern speichern**.
Hinweis zur Kompatibilität: Diese Option wird nur für Datensätze berücksichtigt, die danach neu für die Tabelle erstellt oder geändert werden. Markieren Sie diese Option für eine Tabelle, die bereits Datensätze enthält, werden sie nicht geändert und die Tabelle arbeitet mit einer Mischung aus interner/externer Speicherung.
Wollen Sie das externe Speichern auch auf die vorhandenen Datensätze anwenden, müssen Sie die Daten mit der Option **Erzwinge Aktualisierung aller Datensätze** komprimieren. Weitere Informationen dazu finden Sie im Abschnitt **Seite Komprimieren**.

Max. Größe bei interner Speicherung

Wie bereits im vorigen Absatz beschrieben, werden zur Optimierung Daten aus Feldern vom Typ BLOB, Bild, Text und Objekt standardmäßig außerhalb der Datensätze oder außerhalb der Datendatei gespeichert. Bearbeiten Ihre Anwendung BLOBs,

Bild-, Text- oder Objektdaten unterschiedlicher Größe, ist es hilfreich, wenn Sie den Speicherort von der Größe abhängig machen können, um die Performance zu optimieren.

Diese Einstellung können Sie im Inspektorfenster für Felder definieren. Der im Bereich **Max. Größe bei interner Speicherung** eingegebene Wert gibt den Grenzwert in Bytes an, bis zu dem die Daten im Feld innerhalb des Datensatzes gespeichert werden. Geben Sie z.B. für ein Datenfeld vom Typ Bild den Wert 30 000 ein, wird ein Bild mit 20 KB innerhalb, ein Bild mit 40 KB außerhalb des Datensatzes gespeichert. Standardmäßig ist der Wert 0 vorgegeben, d.h. alle Daten werden außerhalb der Datensätze gespeichert.

UUID Format

Diese Eigenschaft ist für Datenfelder vom Typ Alpha verfügbar. Sie gibt an, dass das Feld UUID Identifier speichert. Die gespeicherten Daten müssen dann konform zum UUID Format sein (Kombination aus 32 Buchstaben [A-F, a-f] und Zahlen [0-9]). Sie können dafür die Eigenschaft **Auto UUID**, den 4D Befehl **Generate UUID** oder einen beliebigen eigenen Algorithmus verwenden.

Versuchen Sie, eine Zeichenkette zu speichern, die nicht dem UUID Format in diesem Feld entspricht, wandelt 4D diese automatisch um. Diese Operation wird auch auf den Inhalt nicht-alfanumerischer Datenfelder angewandt, die in UUID Felder umgewandelt werden: Beim Laden der Datensätze werden die Werte umformatiert und erneut gespeichert.

Felder mit der Eigenschaft **UUID Format** lassen sich in Formularen anzeigen und bleiben eingebbar. Ihr Inhalt erscheint in Großbuchstaben. Zur Anzeige von Kleinbuchstaben müssen Sie über eine Variable gehen.

Hinweise:

- Felder mit UUID Format lassen sich nicht mit Volltextindizes oder Auswahllisten kombinieren.
- Sie können eine Verknüpfung zwischen zwei Feldern ziehen, wenn beide das UUID Format haben. Sie können dagegen nicht ein standardmäßiges Alpha Feld mit einem Feld mit UUID Format verbinden.
- Ein UUID Feld, das initialisiert (generiert) ist und den Wert NULL hat, gibt einen leeren String zurück. Ein nicht-generiertes UUID Feld ist nicht NULL und gibt "000..." zurück (die Anzahl der Nullen entspricht der Anzahl der Zeichen). Die Eigenschaft "NULL Werten leere Werte zuordnen" wird bei nicht-generierten UUID Feldern nicht berücksichtigt (Anzeige von "000").

Auto UUID

Diese Option ist nur aktiv, wenn Sie die Eigenschaft **UUID Format** gewählt haben.

Damit wird, immer wenn ein Datensatz erstellt wird, im Feld automatisch eine UUID Nummer erzeugt.

Diese Nummer wird in folgenden Kontexten automatisch berechnet:

- Beim Erstellen des Datensatzes
- Beim Laden des Datensatzes, wenn sein UUID Feld einen Nullwert enthält. Das passiert insbesondere bei Datensätzen, die erstellt und gesichert wurden, bevor das UUID Feld in der Tabelle hinzugefügt wird und die Option "Setze NULL Werte auf leere Werte" inaktiv ist.

In allen Fällen muss der Datensatz natürlich gesichert werden, damit das automatisch generierte UUID im Feld gesichert wird.

Hinweis: Beim Datenimport generiert 4D keine neue Nummer, selbst wenn diese Eigenschaft markiert ist, sondern verwendet die importierten Werte und wandelt sie bei Bedarf um, wenn das Format nicht gültig ist. Ist der Wert des importierten Feldes dagegen leer, wird automatisch eine UUID zugewiesen.

QuickTime Komprimierung

Hinweis zur Kompatibilität: Diese Option wird nur zur Wahrung der Kompatibilität beibehalten. Sie hat standardmäßig keine Auswirkung. Um sie zu nutzen, müssen Sie QuickTime in Ihrer Anwendung aktivieren (siehe **Überblick über Bilder**).

Suchen und Sortieren in Text ohne Tags

Diese Eigenschaft ist für Datenfelder vom Typ Text oder Alpha verfügbar. Ist diese Option markiert, berücksichtigen Such- und Sortierläufe auf die im Datenfeld gespeicherten Daten keine evtl. enthaltenen Stil-Tags.

Diese Option bezieht sich auf die Möglichkeit, verschiedene Stilarten innerhalb eines Textbereichs im Formular anzuwenden. Weitere Informationen dazu finden Sie im Abschnitt **Text mit Stil (Rich Text)**.

Stilelemente werden durch Einfügen von HTML Tags in den Text angelegt. Diese Tags werden beim Anzeigen des Textbereichs interpretiert.

Stil-Tags werden mit den Daten gespeichert. Schreiben Sie zum Beispiel "Wochen ende" in ein Textfeld, speichert 4D "Wochen ende". Diese Formatierung ist für den Benutzer auf Formularebene unsichtbar. Such- und Sortierläufe erfordern dagegen eine spezifische Einstellung, damit 4D die Stil Tags ignoriert. Die Suche nach "Wochenende" findet das Wort nur, wenn Sie für das Feld im Inspektorfenster die Option **Text Suchen/Sortieren ohne Stil-Tags** markiert haben.

Hinweis: Mit dieser Option entspricht eine Suche für *derWert* in den Daten von *dasFeld* folgender Anweisung in 4D:

```
QUERY BY FORMULA(OBJECT Get plain text(dasFeld)="derWert")
```

Indizierung

Die Eigenschaft **Index** ist für alle Datenfeldtypen mit Ausnahme von BLOB und Bildern verfügbar. Der **Volltext-Index** ist für Datenfelder vom Typ Alpha, Text und Bild verfügbar.

Über **Indizes** können Sie Bearbeiten und Suchen in den Daten beschleunigen. Weitere Informationen dazu finden Sie im Abschnitt **Indizes erstellen und ändern**.

Dateneingabe Kontrolle

Sie können für Datenfelder und eingebbare Objekte auf Formularebene Dateneingabe Kontrollen definieren. Sie beschränken dann die Eingabe durch den Benutzer auf eine bestimmte Form.

Zwingend

Ist für ein Datenfeld die Eigenschaft **Zwingend** markiert, muss der Benutzer bei der Dateneingabe in dieses Datenfeld einen Wert eingeben. 4D akzeptiert keinen Datensatz, der ein leeres Datenfeld enthält, wenn die Eingabe zwingend ist. Diese Eigenschaft bewahrt die Integrität der Datensätze. Sie ist für Datenfelder sinnvoll, die wichtige Informationen für Ihre Datenbank enthalten. Ein Datenfeld, das jeden Datensatz eindeutig kennzeichnet, ist hierfür ein gutes Beispiel. Das kann die Nummer der Sozialversicherung, Rechnungsnummer oder die Personalnummer sein.

Sie können **Zwingend** auch für ein Datenfeld in einem einzelnen Formular vorgeben. Wählen Sie die Eigenschaft im Struktureditor, können Sie diese für ein einzelnes Formular nicht deaktivieren. Sie können sie jedoch in einem Formular für ein Datenfeld anwenden, welches diese Eigenschaft im Struktureditor nicht hat. Weitere Informationen dazu finden Sie im Abschnitt **Attribute Eingebbar und Zwingend und Feldeigenschaften**.

Hinweis: In 4D können Felder auch die Eigenschaft **Eingabe von NULL verweigern** haben (siehe oben). Beide Konzepte sind ähnlich, sind jedoch von unterschiedlicher Reichweite: **Zwingend** ist eine Dateneingabe Kontrolle, während **Eingabe von NULL verweigern** auf Engine-Ebene arbeitet.

Nicht änderbar

Ist die Eigenschaft **Nicht änderbar** für ein Datenfeld vorgegeben, akzeptiert 4D den anfänglich in dieses Datenfeld eingegebenen Wert, erlaubt jedoch dem Benutzer nicht, diesen Wert zu ändern, sobald der Datensatz gespeichert ist. Der Benutzer kann einen Eintrag in ein solches Datenfeld nur beim ersten Erstellen bearbeiten. Sobald er diesen Datensatz speichert, lässt sich der Wert nicht mehr bearbeiten. Er lässt sich dann nur über eine Methode oder im Designmodus durch Aufheben dieser Eigenschaft ändern.

Sie benutzen **Nicht änderbar** für Datenfelder, die zum Prüfen dienen, wie z.B. Empfangsdatum, Zahlungstermin usw. Diese Eigenschaft wird auch oft für Feld mit dem Attribut einmalig verwendet.

Hinweis: Diese Eigenschaft gilt nur für Datenfelder im Standardeingabeformular. In den anderen Fällen, also Eingabe in Liste, in Unterformularen im Eingabe- oder Ausgabeformular, lässt sich der Wert des Datenfeldes weiterhin ändern.

Nur anzeigen

Der Benutzer kann in ein Datenfeld, für das die Eigenschaft **Nur anzeigen** markiert ist, keine Werte über die Tastatur eingeben. Für dieses Datenfeld müssen Sie einen voreingestellten Wert verwenden oder eine Methode schreiben, die einen Wert in dieses Datenfeld eingibt. Ein Datenfeld mit der Eigenschaft **Nur anzeigen** ist sinnvoll für Werte, die der Benutzer der Anwendung nicht ändern soll, wie z.B. berechnete Summen oder eine von einer Methode vergebene laufende Nummer. Sie können in einem bestimmten Formular ein beliebiges Datenfeld nicht-eingebbar machen. Weitere Informationen dazu finden Sie im Abschnitt **Attribute Eingebbar und Zwingend und Feldeigenschaften**.

Mehrzeilig

Diese Option ist nur für Datenfelder vom Typ Text verfügbar. Ist sie markiert, wird das Textfeld automatisch gestaltet, so dass es mehrere Zeilen enthalten kann. Es erhält folgende Merkmale:

- Höhe entspricht mehreren Zeilen
- Horizontaler Rollbalken
- Während der Ausführung erzeugt die Zeilenschaltung einen Zeilenumbruch

Ist diese Option nicht markiert, erscheinen Textfelder in Formularen wie alphanumerische Felder, d.h. sie bestehen aus einer einzigen Zeile und haben keinen Rollbalken. In diesem Fall springen Sie mit der Zeilenschaltung in das nächste Datenfeld im Formular.

Sie können die Standarddarstellung des Feldes jederzeit im Formulareditor verändern.

Auswahlliste

Die Eigenschaft **Auswahlliste** verwenden Sie, wenn Sie für die Eingabe von Daten in das Datenfeld eine Auswahlliste anzeigen wollen. Dafür müssen Sie vorher im Listeneditor eine Auswahlliste anlegen (siehe **Auswahllisten**).

Über Auswahllisten können Sie Eingaben in das Datenfeld standardisieren und Schreibfehler vermeiden. Sie benutzen eine Auswahlliste für ein Datenfeld mit einer begrenzten Anzahl gültiger oder gebräuchlicher Eingaben. Die Verwendung einer Auswahlliste hindert den Benutzer nicht daran, einen anderen Wert einzugeben, d.h. einen Wert, der nicht in der Auswahlliste steht.

Sie können die Auswahlliste auch einem Datenfeld in einem bestimmten Formular zuweisen. Wenn Sie die Auswahlliste nur einem Formular zuweisen, erscheint sie nicht in anderen Editoren, wie z.B. dem Sucheditor und in den Dialogfenstern.

Weitere Informationen dazu finden Sie im Abschnitt **Eingabekontrollen**.

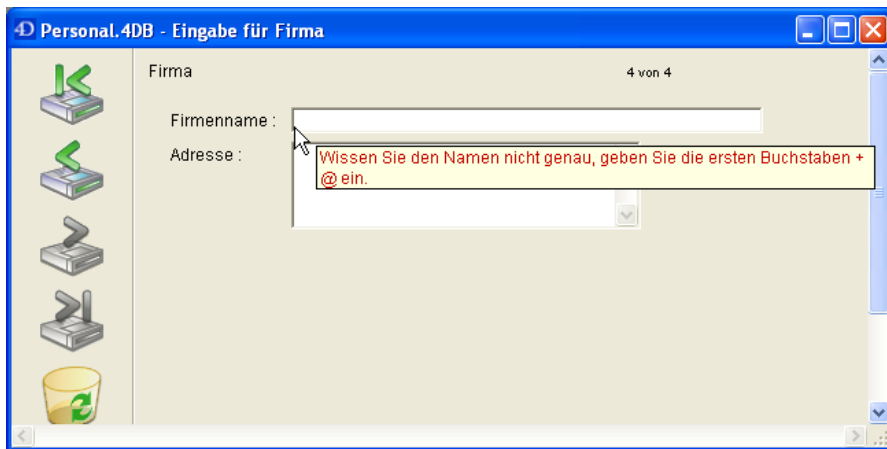
Hinweis: Auf Formularebene können Sie auch Auswahllisten mit erforderlichen oder ausgeschlossenen Werten einrichten.

Markieren Sie die Eigenschaft **Auswahlliste**, wird das zugeordnete DropDown-Menü aktiviert. Hier können Sie dem Datenfeld eine bereits vorhandene Auswahlliste zuweisen oder auf den Bereich [...] klicken, um den Listeneditor zum Erstellen einer Liste zu öffnen (siehe **Auswahllisten**).

Hilfetipps

Mit einem Tipp, den Sie einem Datenfeld hinzufügen, geben Sie Benutzern weitere Informationen über das Datenfeld. Sind die Tipps aktiviert, erscheint der Tipp – egal, in welchem Formular das Datenfeld liegt – sobald der Benutzer den Mauszeiger auf das Datenfeld setzt. Ein Tipp erscheint auf allen Plattformen.

Zieht der Benutzer den Mauszeiger über das Datenfeld, erscheint der Tipp unter dem Datenfeld. Hierzu ein Beispiel:



Hinweis: Über den Selektor Tips enabled des Befehls **SET DATABASE PARAMETER** lassen sich Hilfetipps für die gesamte Anwendung deaktivieren oder aktivieren.

Sie können Hilfemeldungen den Feldern auch auf 2 andere Arten zuweisen:

- Auf Formularebene, so dass er nicht in den anderen Formularen erscheint. Weitere Informationen dazu finden Sie im Abschnitt **Hilfemeldungen**.
- Über den Befehl **OBJECT SET HELP TIP** für den aktuellen Prozess.

Kommentar

Im Bereich Kommentar in der Inspektorpalette können Sie zusätzliche Information über das Datenfeld speichern, die dann für alle Entwickler zur Verfügung steht.

Beachten Sie, dass jede Tabelle und jede Verknüpfung einen eigenen Bereich für Kommentare hat.

SQL

Der Bereich SQL der Inspektorpalette enthält verschiedene Informationen über das Datenfeld zur Verwendung mit der SQL Programmiersprache.

Der Bereich zeigt an, ob der Name des Datenfeldes für SQL nicht-erlaubte Zeichen enthält. SQL erlaubt z.B. im Gegensatz zu 4D keine Namen mit Leerzeichen.

Bei Datenfeldern erscheinen auch die SQL Attribute wie Typ und Eigenschaften.

Daten extern speichern

Sie können wählen, wo Sie Daten aus einem Datenfeld vom Typ BLOB, Bild, Text bzw. Objekt speichern wollen. Sie können den Inhalt dieser Felder, zusätzlich zu den internen Speicheroptionen (im Datensatz oder in der Datendatei), auch außerhalb der Datendatei abspeichern. In diesem Fall werden Ihre Daten als externe Dateien gespeichert, die 4D verwaltet und auch über Programme von Drittherstellern steuerbar sind - vorausgesetzt, Sie sorgen für den Erhalt der Datenintegrität.

Dies läuft für den Benutzer unsichtbar ab: Der Zugriff auf die Daten bleibt gleich, egal wo sie gespeichert sind.

Durch externes Speichern der Daten können Sie die Funktionsweise der Anwendung optimieren. So können Sie beispielsweise umfangreiche Daten transportieren oder über das Betriebssystem auf Text bzw. Bilder innerhalb der Datenbank zugreifen, selbst wenn sie nicht geöffnet ist.

Automatischer oder eigener Modus

Es gibt zwei Möglichkeiten, um Daten extern zu speichern:

- **Automatischer Modus:** In diesem Modus erstellt und verwaltet 4D einen Standardordner, der in einer bestimmten Art strukturiert ist und alle extern gespeicherten Daten enthält. Diese Daten werden im Hintergrund gesteuert und haben die gleiche Funktionsweise und automatischen Aktionen wie intern gespeicherte Daten. Sie können den automatischen Modus folgendermaßen steuern:
 - Externes Speichern im Struktureditor konfigurieren (Die Einstellungen werden mit der Struktur der Datenbank gesichert). Weitere Informationen dazu finden Sie im Abschnitt .
 - Über den Befehl **SET EXTERNAL DATA PATH** mit einer Konstante im Parameter *Pfad* (die Einstellung gilt für die Arbeitssitzung).
- **Eigener Modus:** In diesem Modus wählen Sie selbst, wo die externen Dateien zu Datenfeldern bzw. Datensätzen gespeichert werden. 4D behält dann nur den Link zwischen dem Feld und seinen Daten und greift im Nur-Lesen Modus darauf zu, einige Mechanismen der Datenbank sind nicht länger verfügbar. Sie aktivieren den eigenen Modus über den Befehl **SET EXTERNAL DATA PATH** und geben im Parameter *Pfad* einen anderen Ordner als den Standardordner an.

Nachfolgende Tabelle zeigt die Funktionsweise und automatischen Aktionen im automatischen und im eigenen Modus:

	Automatischer Modus	Eigener Modus
Ordner zum Speichern einrichten	Name und Speicherort werden von 4D gesetzt; nur ein Standardordner für die gesamte Datenbank	Name und Speicherort frei wählbar; kann für jedes Feld anders sein
Externe Dateien erstellen, laden und sichern	Automatisch	Automatisch
Externe Datei löschen, wenn der Datensatz gelöscht wird	Ja	Nein
Externe Datei löschen, wenn dem Feld Nullwert zugewiesen wird	Ja	Nein
Automatische Integration beim Sichern der Datenbank	Ja	Ja (externer Pfad)
Automatische Unterstützung, wenn das Logbuch integriert wird	Ja	Ja (externer Pfad)
Unterstützung von Standardindizes (Textfelder)	Nein	Nein
Unterstützung von Volltext-Indizes (Text- und Bildfelder)	Ja	Nein
Unterstützung von Transaktionen	Ja	Nein

Speicherort der externen Dateien

Außerhalb der Datendatei gespeicherte Daten werden nach folgenden Kriterien organisiert:

- Die Daten werden für jeden Datensatz in einer externen Datei mit Namen xxx.txt (Textfeld), xxx.blob (Blob- oder Objektfeld) oder xxx.jpg/xxx.tiff... (Bildfeld) gespeichert, wobei xxx ein einmaliger Identifier (UUID) ist, den 4D verwaltet.
- Im automatischen Modus werden alle externen Daten in einen Ordner mit Namen *<DatabaseName>.ExternalData* gelegt und *<DatabaseName>* wird durch den Namen der Strukturdatei der Anwendung ersetzt. Dieser Ordner wird neben die Datendatei (.4DD Datei) der Anwendung gelegt. Innerhalb dieses Ordners erstellt 4D pro Tabelle mit externen Daten einen Ordner mit Namen *Tabelle+Tabellennummer*, dann pro externes Datenfeld einen Unterordner mit Namen *Feld+Feldnummer*. Die ersten 100 Einträge des Feldes werden auf der obersten Ebene dieses Ordners gespeichert, die darauffolgenden in Unterordnern, beginnend mit Nr. 2. Jeder Unterordner kann bis zu 100 Elemente enthalten. Wird z.B. der Inhalt von Feld Nr. 5 von Tabelle Nr.3 in der Anwendung "Accounting" außerhalb der Datei gespeichert,

erstellt 4D hierfür folgende Baumstruktur:

Accounting.4DD

[Accounting.ExternalData]

Table3

Field5

Data_1B7F3A 56F6544B45951EFA60426D5ABC.txt

Data_1B7F3A 56F6544B45951EFA60426D5CCC.txt

...

2

Data_2ADBFBA478AAE4409DA9C2D13C90A53B.txt

Data_32F8A30B87EE7E4BBC802468D553DC43.txt

...

- Im eigenen Modus bestimmen Sie Name und Speicherort des Ordners selbst und lassen sich für den extern gespeicherten Inhalt jedes Datenfeldes über den Befehl **SET EXTERNAL DATA PATH** separat angeben. Diese Einstellung wird nicht in der Struktur der Anwendung gespeichert.

Dateien erstellen und aktualisieren

Im automatischen Modus werden Daten von Datenfeldern in einer externen Datei gespeichert, wenn der Datensatz auf die Festplatte gesichert wird - falls erforderlich nach Bestätigen der Transaktion:

- Existiert die externe Datei nicht, wird sie im automatischen Modus angelegt, im eigenen Modus wird ein Fehler zurückgegeben
- Existiert bereits eine externe Datei, ersetzt 4D sie durch eine neue
Um die bisherige Datei beizubehalten, können Sie entweder einen anderen Pfad angeben (über den Befehl **SET EXTERNAL DATA PATH**) oder über den Befehl **RELOAD EXTERNAL DATA** den Inhalt des Feldes aus seiner externen Datei in den Speicher laden, bevor er erneut auf der Festplatte gesichert wird. Diese Option ist hilfreich, wenn die Datei nach dem Laden des Datensatzes von einer anderen Anwendung geändert wurde.

Im eigenen Modus (Pfad definiert durch den Befehl **SET EXTERNAL DATA PATH**) speichert 4D nur den Pfadnamen der Datei, wenn der Datensatz auf der Festplatte gesichert wurde. Der Entwickler muss die Datei selbst verwalten (Erstellen und Ändern).

Synchronisation und Replikation

Der Speicherplatz für die Daten ist in jeder Anwendung ein lokaler Parameter. Er kann beim Synchronisieren oder Replizieren der Daten (siehe **Replication via SQL** in der lokalen und in der remote Anwendung unterschiedlich sein. In diesem Fall passt das Speichern zum Parameter der jeweiligen Anwendung; Synchronisieren oder Replizieren ändert diese Parameter nicht.

Beispiel: Ein Datenfeld vom Typ Bild wird in der remote Anwendung außerhalb der Datendatei gespeichert; dasselbe Datenfeld wird in der lokalen Anwendung innerhalb der Datendatei gespeichert. Beim Replizieren werden alle Daten, die diesem Datenfeld in der lokalen Anwendung hinzugefügt wurden, in der remote Anwendung weiterhin außerhalb der Datendatei gespeichert.

Extern auf die Dateien zugreifen

Extern gespeicherte Dateien lassen sich auch von anderen Programmen als 4D im Lese-/Schreibmodus öffnen (Betriebssysteme, Text- oder Grafikedatoren, usw.). Das sollte jedoch mit Bedacht geschehen, da es u.U. die Funktionsweise der Anwendung beeinträchtigen kann:

- Wird eine extern gespeicherte Datei durch das Betriebssystem oder ein third-party Programm gelöscht, umbenannt oder verschoben, wertet 4D das dazugehörige Feld als Wert Null. Im automatischen Modus wird die Datei erneut angelegt (sofern sie nicht Null ist), wenn der Datensatz gesichert wird.
Im eigenen Modus wird bei Ausführen des Befehls **SET EXTERNAL DATA PATH** ein Fehler generiert.
- Verwenden Sie Indizes und werden die gespeicherten Dateien über ein third-party Programm geändert, ohne die dazugehörigen Datensätze erneut zu speichern, werden die Indizes nicht aktualisiert.

Hinweis: Externe Textdateien werden in UTF-8 ohne BOM-Format gesichert. Werden sie über ein third-party Programm geöffnet und dann mit einem BOM gesichert, kann 4D sie weiterhin öffnen, sie werden dann jedoch wieder ohne BOM gesichert.

Beachten Sie, dass Laden eines Datensatzes im Nur-Lesen Modus nicht die externen Dateien für die Datenfelder dieses Datensatzes sperrt. Die Dateien sind über 4D oder ein anderes Programm weiterhin änderbar, selbst wenn 4D ihren Inhalt in den Speicher lädt.

Namensregeln für Tabellen und Datenfelder

Für Tabellen und Felder gelten folgende Regeln:

- Der Name kann bis zu 31 Zeichen lang sein.
- Er muss mit einem Buchstaben beginnen.
- Er kann beliebige Kombinationen aus Buchstaben oder Zahlen, Leerzeichen oder Unterstrichen enthalten.
- Folgende Zeichen sind in Tabellennamen nicht erlaubt; das erzeugt einen Fehler im Struktureditor: () + - / * " ; = & | # > < ^ ' { } % DIAMOND (0x00D7), CUBE (0x00B3), SQUARE (0x00B2), PLUS-MINUS (0x00B1)
- Sie sollten im allgemeinen Zeichen vermeiden, die in 4D oder anderen Sprachen wie XML, ODBC zu Fehlinterpretation führen könnten. Das sind insbesondere Punktierungszeichen (Komma, Doppelpunkt, etc.) oder diakritische Zeichen (ä, ö, ü,) und scharfes s (ß).
- 4D schneidet Namen ab, die länger als 31 Zeichen sind, und entfernt Leerzeichen am Anfang oder Ende des Namens.
- Verwenden Sie nicht zweimal denselben Namen für sichtbare Objekte. Sie erhalten dann eine Meldung, dass ein anderes Objekt bereits denselben Namen hat und die Eingabe nicht möglich ist.
- Verwenden Sie keine reservierten Namen für Tabellen und Datenfelder. Das sind z.B. Befehlsnamen wie *Date*, *Time*, Schlüsselwörter wie *If*, *For* oder Konstanten.

Tipp: In 4D können Sie in Objektnamen Leerzeichen verwenden. Namen ohne Leerzeichen haben jedoch den Vorteil, dass Sie diesen Namen im Methodeneditor durch Doppelklick auswählen können. Deshalb sollten Sie anstelle der Leerzeichen Unterstriche verwenden.

Hinweise:

- Für SQL gelten zusätzliche Einschränkungen: Nur die Zeichen `_0123456789abcdefghijklmnopqrstuvwxyz` sind erlaubt. Namen dürfen keine SQL Schlüsselwörter wie *Command*, *Attribute* enthalten. Der SQL Bereich in der Inspektorraute zeigt eine Meldung, wenn ein Name eine SQL Regel nicht beachtet. Weitere Informationen dazu finden Sie im Abschnitt **UNREGISTER CLIENT**.
- Achten Sie darauf, dass die Zeichen mit JavaScript kompatibel sind, wenn Sie über einen **4D Mobile** Link von einer Wakanda Applikation aus auf Ihre Tabellen und Felder zugreifen wollen (siehe [Programming and Writing Conventions](#) in der Wakanda Dokumentation).

Indizes erstellen und ändern

Sie können Datenfelder, nach denen Sie häufig suchen und sortieren, mit einem Index versehen. Sie können z.B. die Datenfelder Nachname, Firmenname oder Produktname indizieren, wenn Sie nach besonderen Datensätzen suchen oder die Datensätze nach diesen Datenfeldern sortieren. Sie benutzen diese Eigenschaft auch für Datenfelder, die Verknüpfungen zwischen Tabellen herstellen. Weitere Informationen finden Sie im Abschnitt **Verknüpfungen erstellen und ändern**.

Ist ein Index zugeordnet, erzeugt 4D eine interne Indextabelle für das Datenfeld. Damit kann 4D in dem Datenfeld schnelle Such- und Sortierläufe durchführen. Wenn 4D in einem nicht indizierten Datenfeld eine Suche oder Sortierung vornimmt, wird jeder Datensatz sequentiell durchlaufen und der Wert geprüft. Mit einem Index kann 4D suchen und sortieren, ohne jeden Datensatz zu laden.

Sie können Datenfelder vom Typ Alpha, Text, Datum, Zeit, Ganzzahl, Lange Ganzzahl, Ganzzahl 64 Bit, Zahl, Fließkomma, Bild und Objekt indizieren. Wenn Sie Datensätze hinzufügen oder löschen, aktualisiert 4D eine Indextabelle automatisch. Weisen Sie den Index einem bestehenden Datenfeld zu, indiziert 4D automatisch die bestehenden Daten, sobald Sie den Designmodus verlassen. Sie können beliebig viele Datenfelder indizieren.

Jede Indextabelle kann enthalten bis zu:

- 128 Milliarden Schlüssel für Alpha, Text und Float Indices;
- 256 Milliarden Schlüssel für andere Indextypen (skalare Daten).

Indizieren Sie nicht jedes Datenfeld. Denn Indizes machen die Anwendung größer, sie benötigt mehr Speicherplatz auf Ihrer Festplatte. Viele Indizes verlängern auch die Zeit zum Speichern eines Datensatzes, da 4D die Indextabelle mit jeder Eingabe aktualisiert.

Indizierte Datenfelder erscheinen im Strukturfenster in **Fettschrift**.

Indexarten

Ein Standardindex dient zum Beschleunigen von Datenbankoperationen und bezieht sich auf einen generischen Index im Gegensatz zu textbezogenen oder kombinierten Indizes. Es gibt drei Auswahlmöglichkeiten:

- **Standardindex:** Dieser Index ist einem einzelnen Datenfeld zugeordnet und sorgt für rasche Durchführung von Standardoperationen der Datenbank, wie Suchen und Sortieren. Sie können noch die interne Architektur bestimmen (außer für Objektfelder): B-Tree oder Cluster B-Tree.
- **Zusammengesetzter Index:** Dieser Index speichert kombinierte Werte zwischen zwei oder mehr Datenfeldern, nach denen oft zusammen gesucht wird, z.B. Nachname + Vorname.
- **Volltext-Index:** Diese Indexart ist nur für Datenfelder vom Typ alphanumerisch, Text und Bild verfügbar. Damit können Sie rasch innerhalb von Text suchen oder bei Bildern nach zugewiesenen Schlüsselwörtern.

Standardindex

Ein Standardindex dient zum Beschleunigen von Datenbankoperationen und bezieht sich auf einen generischen Index im Gegensatz zu textbezogenen oder kombinierten Indizes. Es gibt zwei Auswahlmöglichkeiten:

- **B-tree:** Index vom Typ Standard B-Baum. Dieser vielseitige Indextyp erfüllt alle gängigen Anforderungen.
- **Cluster B-tree:** Index vom Typ B-Baum mit Cluster. Diese Architektur ist effizienter, wenn der Index nur wenig Schlüsselwörter (keys) enthält, z.B. wenn häufig derselbe Wert in den Daten vorkommt.

Wenn Sie eine Index Architektur wählen, bietet 4D auch die Option **Automatisch**. In diesem Fall wählt 4D automatisch die Architektur, die zum gewählten Datentyp passt.

Hinweis: Ein B-Tree Index, der einem Datenfeld vom Typ Text zugeordnet ist, speichert maximal die ersten 1024 Zeichen des Feldes. Folglich schlagen Suchläufe nach Strings mit mehr als 1024 Zeichen fehl.

Wenn Sie eine Index Architektur wählen, bietet 4D auch die Option **Automatisch**. In diesem Fall wählt 4D automatisch die Architektur, die zum gewählten Datentyp passt.

Automatisch ist die einzige verfügbare Option für Datenfelder vom Typ Objekt. In diesem Fall werden alle Attributpfade automatisch indiziert.

Zusammengesetzter Index

Ein zusammengesetzter Index speichert zusammengesetzte Werte aus zwei oder mehr Feldern für jede Eingabe. Ein klassisches Beispiel ist der zusammengesetzte Index aus den Datenfeldern für Vor- und Nachname. Die Suche nach "Peter Schmid" wird von daher optimiert, im Vergleich zur Standardsuche, die erst nach "Schmid" und dann nach "Peter" sucht.

4D nutzt bei Such- und Sortierläufen automatisch die Vorteile zusammengesetzter Indizes. Gibt es z.B. den Index "Stadt+Postleitzahl", wird er bei der Suche vom Typ "Nachname=Becker und Stadt=München und Postleitzahl=812@" verwendet.

Im Struktureditor können Sie zusammengesetzte Indizes nur im Dialogfenster für Indizes erstellen. Weitere Informationen dazu finden Sie im nachfolgenden Abschnitt "Index erstellen".

Volltext-Index

Feldern vom Typ Text, alphanumerisch und Bild können Sie einen Volltext-Index zuweisen.

- Bei dieser Indexart wird der Text in einem Datenfeld Wort für Wort indiziert. Alle Wörter werden indiziert, selbst wenn sie nur aus 1 oder 2 Buchstaben bestehen. Das beschleunigt eine spätere Volltextsuche beträchtlich. Sie können

Datenfeldern vom Typ Text oder alphanumerisch sowohl einen Standard- als auch einen Volltext-Index zuordnen (beim Speichern innerhalb der Datensätze). 4D verwendet dann den zum Kontext passenden Index.

- Weisen Sie diese Indexart einem Bildfeld zu, beschleunigt das die Suche nach Schlüsselwörtern zu Bildern, d.h. Metadaten beträchtlich. Volltext-Indizes für Bilder basieren ausschließlich auf Metadaten vom Typ **IPTC/Keywords**. Diese Typen werden von den Formaten TIFF und JPEG unterstützt. Beachten Sie, dass BMP, PNG und GIF diese nicht unterstützen.

4D aktualisiert die Volltext-Indizes für Bilder automatisch, immer wenn das Datenfeld vom Typ Bild gespeichert wird, d.h. wenn ein Datensatz erstellt oder geändert wird, Daten importiert werden, usw.. Werden Metadaten vom Typ **IPTC/Keywords** im Bild gefunden, indiziert 4D diese automatisch. Sie müssen nicht den Befehl **SET PICTURE METADATA**, aufrufen, um sie in den Index für das Bildfeld einzufügen.

Über den Befehl **DISTINCT VALUES** erhalten Sie die Liste der Schlüsselwörter in einem Volltext-Index.


Sie können Volltext-Indizes für Bild oder Text über den % Operator verwenden: Dieser muss in Formeln zum Suchen bzw. Sortieren gesetzt werden, um einen bestimmten Indexwert zu verwenden:

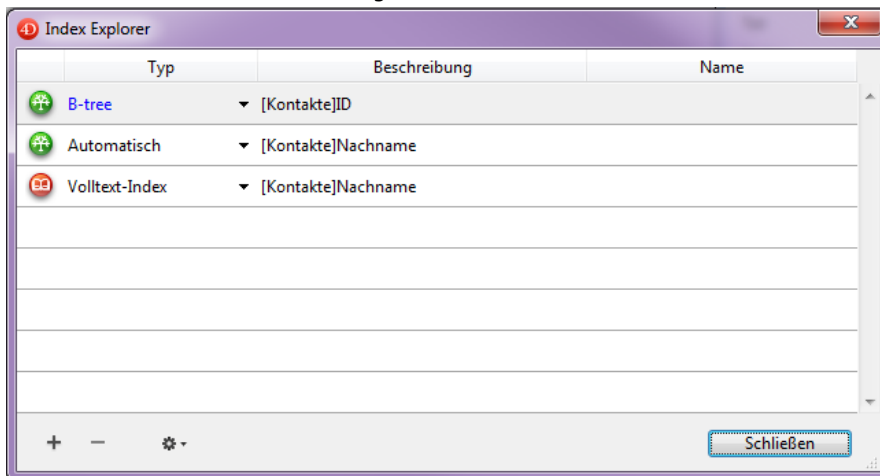
```
QUERY([PICTURES];[PICTURES]Photos %"Katzen")  
// Suche nach Fotos mit dem Schlüsselwort Katzen
```

Das funktioniert auf dieselbe Weise für alle Such- und Sortierläufe über Befehle **QUERY BY FORMULA**, **QUERY SELECTION**, **ORDER BY**, etc.

Weitere Informationen zur Arbeitsweise des Operators % finden Sie im Abschnitt **Vergleichsoperatoren** des Handbuchs *4D Programmiersprache*.


Index Liste


Die Schaltfläche  in der Werkzeugleiste des Struktureditors öffnet das Fenster Index Explorer. Hier erscheint die Liste aller Struktur-Indizes mit ihren Eigenschaften:



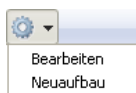
In der Index Liste sehen Sie die Haupteigenschaften der Indizes:

- **Typ:** Indextyp, jeder Typ (B-tree, Cluster B-tree, Volltext-Index) wird mit einem eigenen Icon angezeigt. Wollen Sie den Indextyp verändern, klicken Sie am unteren Rand auf den Pfeil neben der Schaltfläche und wählen im PopUp-Menü den Eintrag **Bearbeiten**. Auf dem Bildschirm erscheint ein Dialogfenster, in dem Sie einen anderen Typ wählen können.
- **Beschreibung:** Tabelle und Felder. Bei einem zusammengesetzten Index zeigt die Liste alle Felder dieses Index.
- **Name:** Indexname. Diese Eigenschaft wird insbesondere von den Befehlen der Programmiersprache verwendet. Doppelklicken Sie in diesen Bereich, um einen Indexnamen zu ändern oder hinzuzufügen.

Die Schaltfläche  zeigt das Dialogfenster Index.

Die Schaltfläche  löscht den gewählten Index (Ein Dialog zum Bestätigen erscheint).

Haben Sie einen Index markiert, sind am unteren Rand im PopUp-Menü der Schaltfläche zwei weitere Befehle verfügbar:



- **Bearbeiten:** öffnet ein Dialogfenster mit den Eigenschaften des gewählten Index. Sie können dieses Fenster auch durch Doppelklick auf eine Zeile der Liste aufrufen (außer Namensbereich).
- **Neuaufbau:** löscht den gewählten Index und baut ihn neu auf. Zuvor erscheint ein Dialog zur Bestätigung.

Index erstellen

Das Erstellen eines Index ist je nach Typ unterschiedlich. Sie können auch wählen, ob Sie den Index direkt erstellen oder das Dialogfenster zum Erstellen des Index verwenden:

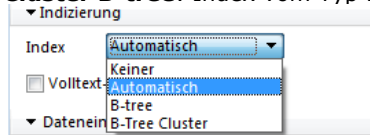
Um einen **Standardindex** direkt anzulegen:

1. Wählen Sie ein Datenfeld und dann in der Inspektoralette im Bereich Index den gewünschten Indextyp
ODER
Klicken Sie mit der rechten Maustaste auf das Datenfeld und wählen dann im **Kontextmenü>Index** den gewünschten

Indextyp.

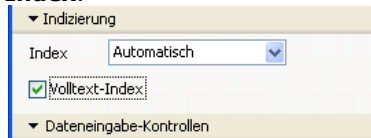
Es gibt folgende Optionen (außer für Felder vom Typ Objekt):

- **Automatisch:** 4D wählt automatisch die interne Architektur, die zum gewählten Datentyp passt.
- **Keiner:** Erstellt keinen Index oder entfernt einen vorhandenen Index.
- **B-tree:** Index vom Typ Standard B-Baum
- **Cluster B-tree:** Index vom Typ B-Baum mit Cluster



Um einen **Volltext-Index** direkt anzulegen:

1. Wählen Sie ein Datenfeld vom Typ alphanumerisch, Text oder Bild und dann in der Inspektorraute die Option **Volltext-Index**.



ODER

Klicken Sie mit der rechten Maustaste auf das Datenfeld und wählen dann im **Kontextmenü>Index** den Eintrag **Volltext**.

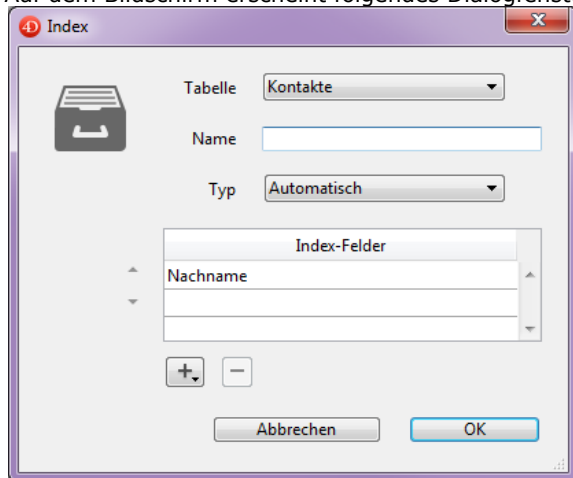
Um einen **zusammengesetzten Index** oder eine andere Indexart über das Dialogfenster *Index* anzulegen:

1. Wählen Sie im Kontextmenü der Tabelle **Neuer Index** oder in der Werkzeugleiste des Struktureditors im DropDown-Menü **Hinzufügen** die Option **Index**.

ODER


Wählen Sie mehrere Felder mit gedrückter **Strg-Taste** (Windows) oder **Befehlstaste** (OS X), klicken dann mit der rechten Maustaste auf ein Datenfeld und wählen im Kontextmenü **Neuer zusammengesetzter Index**.

Auf dem Bildschirm erscheint folgendes Dialogfenster:




Es gibt folgende Vorgaben:

- **Tabelle:** Liste aller Tabellen der Datenbank. Hier wählen Sie die Tabelle, der Sie einen Index zuordnen wollen.
- **Name:** Hier tragen Sie den Namen für den Index ein. Diesen Namen verwenden die Befehle der 4D Programmiersprache.
- **Typ:** Hier wählen Sie den Indextyp aus. Behalten Sie die Option "Automatisch" bei, wählt 4D automatisch den Indextyp nach dem Inhalt des Feldes aus.
- **Index Felder:** Hier geben Sie ein oder mehrere Felder an, die Sie dem Index zuordnen wollen. Je nach der aktuellen Auswahl im Editor kann standardmäßig schon ein Feld enthalten sein.

Um ein Feld hinzuzufügen, klicken Sie auf die Schaltfläche  unter dem Eingabebereich. Auf dem Bildschirm erscheinen die Felder der ausgewählten Tabelle. Wählen Sie die Felder, die Sie dem Index zuordnen wollen.

- Um einen zusammengesetzten Index zu erstellen, fügen Sie die dafür vorgesehenen Felder der Reihe nach hinzu. Ist die Liste fertig, können Sie die Felder per Drag&Drop oder über die Pfeile am linken Rand umsortieren.
- Erstellen Sie einen zusammengesetzten Index mit Primärschlüssel-Feldern, müssen Sie die Felder im Primärschlüssel und im Index in derselben Reihenfolge setzen.
- Haben Sie den Indextyp "Volltext-Index" gewählt, stehen nur Datenfelder vom Typ Alpha oder Text zur Verfügung. Sie müssen dann auch mehrere Felder für den Index wählen.

Um ein Feld aus dem Index zu löschen, wählen Sie dieses in der Liste aus und klicken unter dem Eingabebereich auf die Schaltfläche .

Ist der Index fertig konfiguriert, klicken Sie auf die Schaltfläche **OK**, damit er erstellt wird.

Index im Struktureditor löschen

Sie können Indizes, die nicht mehr benötigt werden, jederzeit löschen. Das ist direkt im Struktureditor oder im Index Explorer möglich. Weitere Informationen zum Index Explorer finden Sie im oberen Abschnitt "Index Liste".

Um einen Standardindex zu löschen:

1. Wählen Sie das Feld mit dem Index, den Sie löschen wollen und dann in der Inspektorpalette im Bereich Index die Option **Keiner**
ODER
2. Klicken Sie mit der rechten Maustaste auf den Feldnamen mit dem Index und wählen im **Kontextmenü>Index** den Eintrag **Nichts**.

Um einen Volltext-Index zu löschen:

1. Wählen Sie das Feld mit dem Index, den Sie löschen wollen und deaktivieren in der Inspektorpalette im Bereich Index die Option **Volltext-Index**
ODER
2. Klicken Sie mit der rechten Maustaste auf den Feldnamen mit dem Index und deaktivieren im **Kontextmenü>Index** den Eintrag Volltext.

Ein zusammengesetzter Index lässt sich nur im Dialogfenster *Index* über die Schaltfläche  löschen.

Datenfeld neu indizieren

Sie können ein Datenfeld jederzeit neu indizieren, d.h. je nach Datenbestand die zugeordneten Indizes neu aufbauen. Das ist z.B. sinnvoll zur Wartung der Anwendung.

Wählen Sie dazu in der Index-Liste im PopUp-Menü am unteren Rand den Eintrag **Neuaufbau**.

Beachten Sie, dass das Ändern der Sprache (siehe [Text-Vergleiche](#)) oder Wartungsoperationen wie Komprimieren (siehe [Seite Komprimieren](#)) auch eine neue Indizierung auslösen.

Verknüpfungen erstellen und ändern

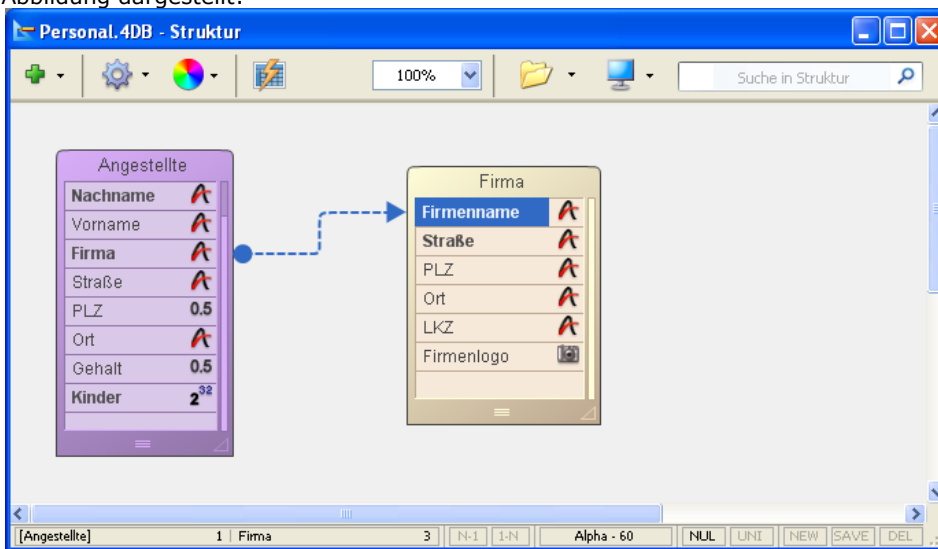
Sie können Verknüpfungen über den Struktureditor oder über die SQL Befehle von 4D erstellen. Dieser Abschnitt beschreibt, wie Sie Verknüpfungen manuell in der Designumgebung anlegen. Weitere Informationen zu SQL Anweisungen in 4D finden Sie im Handbuch *4D SQL Reference*.

Verknüpfung erstellen

Um eine Verknüpfung zwischen zwei Tabellen zu erstellen, ziehen Sie manuell eine Linie zwischen zwei Datenfeldern oder klicken Sie in der Werkzeugleiste des Struktureditors auf das Pluszeichen.

Um im Strukturfenster eine Verknüpfung zu erstellen:

1. Ziehen Sie den Mauszeiger über das Datenfeld in der Viele-Tabelle.
2. Halten Sie die Maustaste gedrückt und ziehen zur Tabelle, die Sie verknüpfen wollen.
Wenn Sie den Mauszeiger bewegen, markiert 4D das Datenfeld und zeichnet die Verknüpfung, wie in der folgenden Abbildung dargestellt:



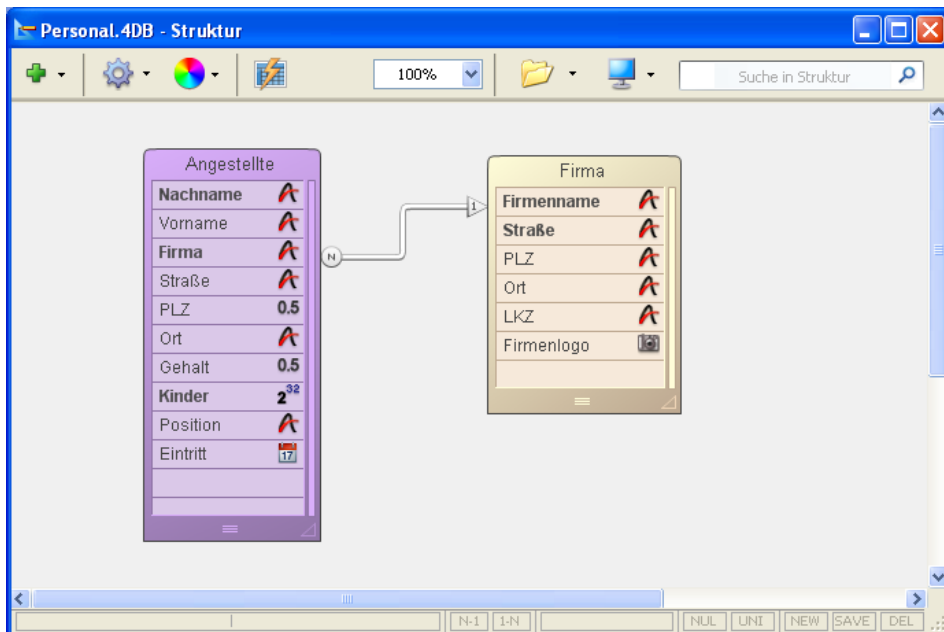
3. Ziehen Sie zum Primärschlüsselfeld in der Eine-Tabelle und lassen die Maustaste los.

Um eine Verknüpfung über das Pluszeichen der Werkzeugleiste zu erstellen:

1. Wählen Sie zwei Felder vom selben Typ aus zwei verschiedenen Tabellen aus:
Die Reihenfolge der Auswahl bestimmt die Richtung der Verknüpfung. Das erste gewählte Datenfeld ist das Fremdschlüsselfeld (Viele-Feld), das zweite das Primärschlüsselfeld (Eine-Feld).
2. Wählen Sie im DropDown-Menü des Pluszeichens den Eintrag **Verknüpfung**.



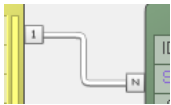
4D zeigt die Verknüpfung als Pfeil zwischen den beiden Tabellen, Anfang und Ende werden mit den Symbolen N und 1 gekennzeichnet:



Das **Inspektorfenster** zeigt die Eigenschaften der Verknüpfung (siehe **Verknüpfungseigenschaften**).

Über SQL erstellte Verknüpfungen

Die Darstellung von Verknüpfungen ist anders, wenn sie via SQL erstellt wurden. Hier erscheinen Anfang und Ende als Quadrate:



Verknüpfung löschen

Sie können eine Verknüpfung jederzeit löschen. Gehen Sie folgendermaßen vor:

1. Wählen Sie die entsprechende Verknüpfung
Ist die Verknüpfung gewählt, werden die Umrisse blau.
2. Klicken Sie mit der rechten Maustaste auf die Verknüpfung und wählen im Kontextmenü den Eintrag **Löschen**
ODER
Drücken Sie die **Löschen-** oder **Rückschritttaste**.

In beiden Fällen erscheint ein Dialogfenster, um die Aktion zu bestätigen oder zu annullieren. Wenn Sie die Aktion bestätigen, entfernt 4D die Verknüpfung, die Tabellen sind nicht länger miteinander verknüpft. Bestehende Indizes im Viele- und Eine-Feld werden nicht gelöscht.

Verknüpfung erneut zeichnen

Sie können eine Verknüpfung jederzeit wiederherstellen. Sie haben z.B. die Verknüpfung versehentlich zwischen den falschen Datenfeldern gezogen oder wollen eine Verknüpfungseigenschaft ändern. In 4D können Sie eine Verknüpfung durch Ziehen der Verknüpfungslinie einfach wieder neu zeichnen.

Bei komplexen Strukturen können Sie Datenfelder in den Quell- und Zieltabellen einer Verknüpfung über die Einträge **Quellfeld auswählen** und **Zielfeld auswählen** des Kontextmenüs wählen.

Um eine bestehende Verknüpfung zwischen zwei Datenfeldern zu verändern, doppelklicken Sie im Struktureditor auf die Verknüpfung. 4D zeigt die Verknüpfungseigenschaften in der Inspektoralette, so dass Sie die gewünschten Veränderungen ausführen können.

Um eine Verknüpfung mit einem anderen Datenfeld der Eine-Tabelle neu zu zeichnen, beginnen Sie die Verknüpfung vom Viele-Feld aus zu ziehen.

Um eine Verknüpfung mit einem anderen Datenfeld der Viele-Tabelle neu zu zeichnen, entfernen Sie zuerst die unerwünschte Verknüpfung und ziehen dann die Verknüpfung vom Viele-Feld aus neu.

Verknüpfungsarten

Die gängigste Verknüpfungsart geht von der Viele-Tabelle zur Eine-Tabelle — genannt Viele-zu-Eine. Sie können aber auch Viele-zu-Viele und Eine-zu-Eine Verknüpfungen einrichten. Alle Verknüpfungen können manuell oder automatisch sein.

Automatische und manuelle Verknüpfungen

Verknüpfungen können automatisch oder manuell sein.

Bei einer automatischen Verknüpfung wählt 4D den entsprechenden Datensatz oder die Datensätze, sobald ein Datensatz in einer verknüpften Tabelle aktualisiert wird. Der bzw. die so bestimmten Datensätze lassen sich ansehen, drucken, ändern oder für eine Suche oder Sortierungen nutzen. Es ist keine Programmierung erforderlich.

Bei einer manuellen Verknüpfung kontrollieren Sie, ob 4D den entsprechenden Datensatz oder die Datensätze in den Speicher lädt. Zur Durchführung dieser Kontrolle benutzen Sie Methoden. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Verknüpfungen**.

Eine manuelle Verknüpfung benutzen Sie z.B., um die Leistung spezifischer Anwendungen zu optimieren, bei denen Sie nicht alle entsprechenden Datensätze jedesmal laden wollen. Wenn Ihre Struktur z.B. drei oder mehr Tabellen verknüpft, wollen Sie kontrollieren, wann verknüpfte Datensätze in den Speicher geladen werden. Eine manuelle Verknüpfung benutzen Sie auch, wenn Sie zwei Tabellen mit zwei separaten Verknüpfungen verbinden wollen. Zwischen zwei Tabellen kann nur eine automatische Verknüpfung bestehen. Manuelle Verknüpfungen zwischen zwei Tabellen können beliebig viele existieren.

Viele-zu-Eine Verknüpfungen

Erstellen Sie eine Verknüpfung zwischen zwei Tabellen, wird die Tabelle mit dem Primärschlüssel der Verknüpfung Eine-Tabelle und die Tabelle mit dem Fremdschlüssel der Verknüpfung Viele-Tabelle genannt. Die Tabellen werden als Eine-Tabelle und Viele-Tabelle bezeichnet, da ein Datensatz in der Eine-Tabelle mit vielen Datensätzen in der Viele-Tabelle verknüpft ist und viele Datensätze in der Viele-Tabelle mit einem Datensatz in der Eine-Tabelle. Diese Art von Tabellenverknüpfung heißt *Viele-zu-Eine* Verknüpfung.

Bei der Verknüpfung von Angestellten und Firmen ist die Tabelle [Firma] die Eine-Tabelle und die Tabelle [Angestellte] die Viele-Tabelle. Ein Firmendatensatz ist verknüpft mit mehreren Angestellten, d.h. mit allen Angestellten, die in dieser Firma arbeiten und mehrere Angestellte sind mit einer Firma verknüpft, d.h. mit der Firma, in der sie arbeiten. Es gibt z.B. einen Datensatz für Acme in der Tabelle [Firma] aber viele Datensätze von Angestellten in der Tabelle [Angestellte], die bei Acme angestellt sind.

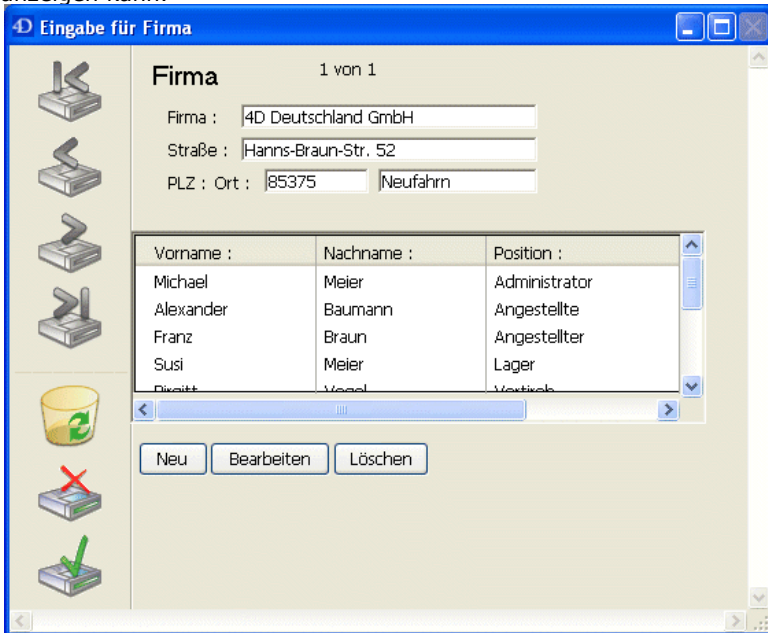
Die folgende Abbildung zeigt, wie der Firmenname in einem Datensatz der Tabelle [Angestellte] einen Datensatz in der Tabelle [Firma] bestimmt, so dass der Datensatz der Tabelle [Angestellte] die Adresse der Firma anzeigt.

Eingabe für Angestellte	
Angestellte 6 von 10	
Vorname :	Annette
Nachname :	Schwab
Position :	Marketing
Firma :	4D Deutschland GmbH
Straße :	Hanns-Braun-Str. 52
PLZ :	85375
Ort :	Neufahrn
Telefon :	0816595190

Umgekehrt, wenn in der Tabelle [Firma] ein Datensatz aktuell wird, erstellt 4D eine Auswahl von Datensätzen der Tabelle [Angestellte] und zeigt diese im Formular. Da die Verknüpfung mehrere Datensätze der anderen Tabelle angibt, können Name und Position vieler Angestellter angezeigt werden. Nur die aktuell im Formular angezeigten Datensätze werden in den Speicher geladen.

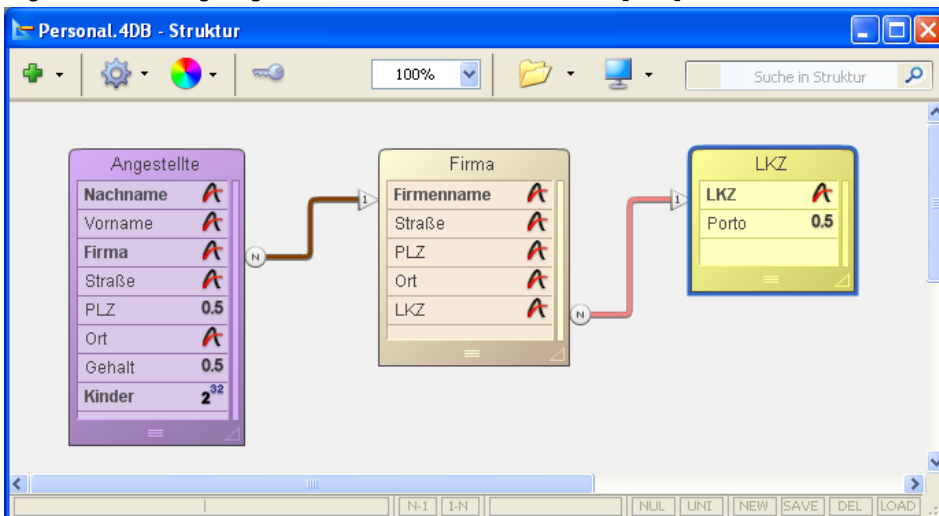
Die folgende Abbildung zeigt, wie ein Firmenname in einem Datensatz der Tabelle [Firma] mehrere Datensätze in der Tabelle [Angestellte] bestimmt, so dass der Datensatz der Tabelle [Firma] eine Liste der in der Firma beschäftigten Angestellten

anzeigen kann.



Der Unterschied zwischen der Eine-Tabelle und der Viele-Tabelle entspricht genau einer spezifischen Verknüpfung. Eine Tabelle kann in einer Verknüpfung die Eine-Tabelle und in einer anderen die Viele-Tabelle sein. Eine Tabelle in einer Verknüpfung kann nur einen Primärschlüssel, jedoch aber mehrere Fremdschlüssel haben.

Nehmen wir z.B. an, Sie verschicken eine Warenprobe an jeden Mitarbeiter in Ihrer Tabelle [Angestellte]. Sie fügen eine Tabelle [LKZ] mit den Landeskennzahlen und dem Porto für jede Landeskennzahl hinzu. Mit dieser Struktur können Sie für jeden Angestellten ein Adressenetikett drucken, welches bereits das Porto für die Postbeförderung des Paketes enthält. Die folgende Abbildung zeigt die Datenbank mit der Tabelle [LKZ]:



Das Datenfeld mit den Landeskürzeln in der Tabelle [LKZ] ist der Primärschlüssel, d.h. die Tabelle [LKZ] ist die Eine-Tabelle. Das Datenfeld mit den Landeskürzeln in der Tabelle [Firma] ist das verknüpfte Datenfeld. Da es ein Fremdschlüssel ist, kann es nicht-eindeutige Werte enthalten. Das Datenfeld mit den Landeskürzeln wird für nah beieinander liegende Firmen identische Landeskürzel enthalten. Die Tabelle [Firma] ist deshalb die Viele-Tabelle, verknüpft mit der Tabelle [LKZ].

Ob es sich bei einer Tabelle um die Eine-Tabelle oder eine Viele-Tabelle handelt, hängt somit von der Verknüpfung mit der anderen Tabelle ab. Die Tabelle [Firma] ist die Viele-Tabelle in der Verknüpfung mit der Tabelle [LKZ] und die Eine-Tabelle in der Verknüpfung mit der Tabelle [Angestellte].

Eine-zu-Eine Verknüpfungen

Eine-zu-Eine-Verknüpfungen werden nur in besonderen Fällen benutzt, da Tabellen, die im Verhältnis Eine-zu-Eine verknüpft sind, auch in der Tabelle selbst kombiniert werden können.

Einige Gründe für Eine-zu-Eine-Verknüpfungen sind:

- Sie haben in Ihrer Datenbank umfangreiche Datenfelder mit Text, BLOB oder Bild. Solche Felder verlangsamen die Datenbank, sobald ein Datensatz aktualisiert und BLOBs in den Speicher geladen werden. Wenn Sie diese Felder in eine andere Tabelle einfügen, werden sie nur bei Bedarf geladen.
- Sie haben eine sehr große Anzahl Datenfelder und wollen sie in logische Gruppen aufteilen. Separate Tabellen machen eine Datenbank schneller und leichter einsetzbar.
- Sie wollen den Zugriff auf bestimmte Datenfelder begrenzen. Wenn Sie einzelne Tabellen benutzen, können Sie für jede Tabelle unterschiedliche Zugriffsrechte vergeben.

Viele-zu-Viele Verknüpfungen

In manchen Fällen müssen Sie viele Datensätze in einer Tabelle mit vielen Datensätzen in einer anderen Tabelle verknüpfen. Dies wird als eine Viele-zu-Viele-Verknüpfung bezeichnet.

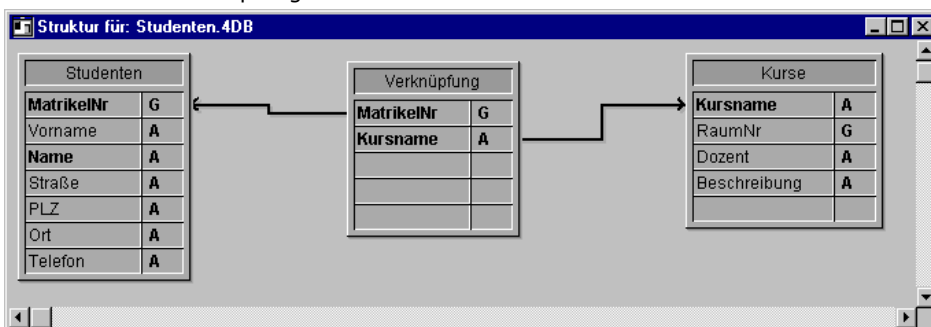
Ein Beispiel dafür ist eine Datenbank zum Verwalten von Kursen. Angenommen, diese Anwendung enthält die beiden Tabellen [Studenten] und [Kurse]. Ein Student kann sich in viele Kurse einschreiben und ein Kurs kann viele Studenten haben. Sie wollen sich alle Kurse ansehen, in die sich ein Student eingeschrieben hat, und auch alle Studenten sehen, die sich pro Kurs eingeschrieben haben.

Andere Beispiele für Viele-zu-Viele-Verknüpfungen sind:

- **[Lieferanten] und [Produkte]:** Jeder Lieferant liefert viele Produkte und jedes Produkt kann von mehreren Lieferanten geliefert werden.
- **[Mitarbeiter] und [Kostenträger]:** Jeder Mitarbeiter arbeitet für viele Kostenträger und für jeden Kostenträger können viele Mitarbeiter tätig sein.
- **[Filme] und [Schauspieler]:** Jeder Film beschäftigt mehrere Schauspieler und jeder Schauspieler kann in mehreren Filmen auftreten.

4D kann automatisch Viele-zu-Viele -Verknüpfungen erstellen. Der Schlüssel hierzu ist eine Zwischentabelle, die mit anderen Tabellen verknüpft ist, die Viele-zu-Viele -Verknüpfungen benutzen. Sie erstellen dann Eingabe- und Ausgabeformulare, die alle erforderlichen Erfassungen der Datensätze und die Datenanzeige durchführen.

Die folgende Abbildung zeigt eine Anwendung für Einschreibungen mit drei Tabellen, nämlich [Studenten], [Kurse] und [Verknüpfung]. Diese Datenbankstruktur wird für den gesamten Abschnitt benutzt, um zu zeigen, wie eine automatische Viele-zu-Viele-Verknüpfung abläuft:



Die Tabelle [Studenten] ist eine Eine-Tabelle. Sie enthält einen Datensatz für jeden Studenten, Namen, und Anschrift. Das Datenfeld *MatrikelNr* bestimmt jeden Studenten eindeutig.

Die Tabelle [Kurse] ist ebenfalls eine Eine-Tabelle. Sie enthält einen Datensatz für jeden Kurs, mit dem Kursnamen und dem Dozenten. Das Datenfeld *Kursname* bestimmt jeden Kurs eindeutig.

Eine Zwischentabelle, nämlich [Verknüpfung] ist für beide anderen Tabellen die Viele-Tabelle. Sie enthält Datensätze für viele Studenten und für viele Kurse. Formulare dieser Tabelle werden für die Eingabe von Daten in beide anderen Tabellen benutzt und für die Anzeige von Informationen in jeder der anderen Tabellen.

Mit Hilfe der drei Tabellen werden die Daten effizient gespeichert. Der komplette Datensatz eines Studenten wird nur einmal gespeichert. Zu jedem Kurs gehört ein Datensatz, nur einmal gespeichert. Die Datensätze, die Studenten mit Kursen verknüpfen, werden für jede Einschreibung einmal gespeichert. Sämtliche Informationen stehen aber in jeder beliebigen Zusammenstellung zur Verfügung.

Daten in Viele-zu-Viele Verknüpfungen eingeben

Für Eingabe und Anzeigen von Informationen der beiden anderen Tabellen benutzen Sie die Zwischentabelle - in diesem Beispiel die Tabelle [Verknüpfung]. Jeder Datensatz, den Sie in die Tabelle [Verknüpfung] eingeben, ist mit den beiden anderen Tabellen verknüpft, nämlich mit einem Studenten und einem Kurs. Die Datensätze der Tabelle [Verknüpfung] enthalten nur die beiden Teilinformationen, welche die Verknüpfung darstellen: Die MatrikelNr des Studenten und den Kursnamen.

Hier sehen Sie ein Beispiel für die Eingabe eines neuen Datensatzes in die Tabelle [Verknüpfung]:

Dieser Datensatz zeigt den Studenten Bernhard Winkler, eingeschrieben in einen Datenbankkurs. Dieser Datensatz verbindet direkt die Informationen der anderen beiden Tabellen.

Ein ähnlicher Datensatz besteht für jeden Kurs, in den sich der Student eingeschrieben hat. Nur die Datenfelder mit der MatrikelNr und dem Kursnamen sind tatsächlich in der Tabelle [Verknüpfung] gespeichert. Jeder Datensatz katalogisiert einen einzelnen Studenten in einem einzelnen Kurs.

Hinweis: Wenn ein Datensatz in die Tabelle [Verknüpfung] eingegeben wird, so als wenn Sie einen Datensatz erstellen, wird hierdurch automatisch eine Auswahl von Datensätzen in den verknüpften Tabellen erstellt. Die Auswahl besteht aus den entsprechenden Datensätzen für den Studenten und den Kurs. Wenn Sie zu einer dieser Tabellen umschalten, wird nur ein einzelner Datensatz angezeigt. Um alle Datensätze anzuzeigen, wählen Sie im Menü **Datensätze** den Befehl **Alle anzeigen**.

Die folgende Abbildung zeigt das Eingabeformular für diesen Datensatz. Beachten Sie, dass es Datenfelder der Tabelle [Studenten] und der Tabelle [Kurse] enthält.

Sie geben nur in die Datenfelder *MatrikelNr* und *Kursname* Daten ein. Wenn Sie eine MatrikelNr eingeben, findet 4D die Informationen über den Studenten in der verknüpften Tabelle [Studenten] und zeigt diese in den Datenfeldern Vorname, Name und Anschrift. Genauso, wenn Sie einen Kursnamen eingeben, findet 4D die Kursinformation in der Tabelle [Kurse] und zeigt diese im Eingabeformular an.

Informationen in einem Unterformular anzeigen

Sie können die Informationen aus diesen drei Tabellen in Unterformularen anzeigen. Im Datensatz für den Studenten zeigen Sie alle Kurse an, in die der Student sich eingeschrieben hat. Im Kursdatensatz zeigen Sie alle Studenten an, die sich in einen Kurs eingeschrieben haben.

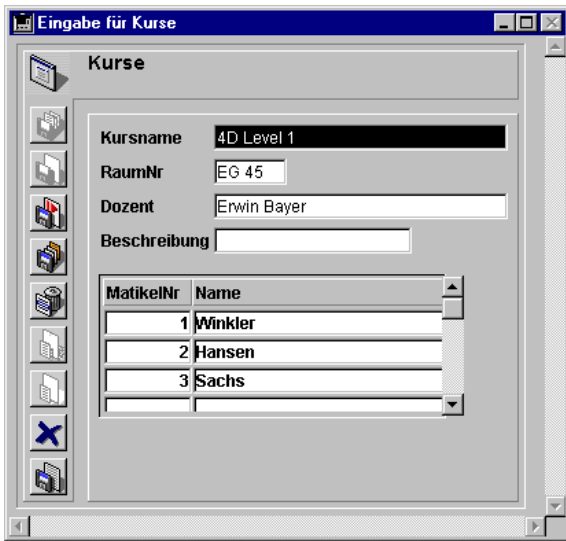
Um in einem Studentendatensatz die Kurse anzuzeigen, benutzen Sie ein Unterformular. Weitere Informationen dazu finden Sie im Abschnitt **Unterformular erstellen**.

Kursname	RaumNr
4D Level 1	EG 45
4D Level 2	EG 77

Der oben abgebildete Datensatz ist in der Tabelle [Studenten]. Er zeigt im oberen Teil die Informationen über den Studenten. Die Informationen über die beiden Kurse, in die er sich eingeschrieben hat, stammen aus der verknüpften Tabelle, welche die Informationen über die Einschreibung enthält.

Beachten Sie, dass dieses Unterformular für die Tabelle [Verknüpfung] gilt, nicht für die Tabelle [Kurse]. Die verknüpfte Tabelle enthält die Datensätze, welche den Datensatz über den Studenten mit den Kurs-Datensätzen verknüpft. Das Unterformular enthält das Datenfeld für den Kursnamen aus der Tabelle [Kurse]. Durch die Verknüpfung zwischen der verknüpften Tabelle und der Tabelle [Kurse] kann 4D automatisch den korrekten Kursnamen anzeigen.

Hier sehen Sie einen Datensatz mit den in einen Kurs eingeschriebenen Studenten:



Es handelt sich um einen Datensatz aus der Tabelle [Kurse]. Er zeigt die Informationen zu dem Kurs und in einer Liste die für diesen Kurs eingeschriebenen Studenten. Die Informationen zu den Studenten stammen ebenfalls aus der verknüpften Tabelle. Diese Tabelle enthält die Datensätze, welche die Kurse mit den in diese Kurse eingeschriebenen Studenten verknüpft.

Im obigen Beispiel für Unterformulare können Sie in jedes der Datenfelder Datensätze eingeben. Wenn Sie z.B. in einen Kursdatensatz einen neuen Studenten eingeben wollen, setzen Sie den Tabulator auf den letzten in dem Unterformular gezeigten Studentendatensatz und drücken unter Windows die Tastenkombination **Strg+Shift+/,** auf Mac OS **Befehl+Shift+/,** um einen neuen Datensatz zu erstellen (Sie können diese Tastenkombination in den Datenbank-Eigenschaften ändern, siehe). Wenn Sie den passenden Kursnamen eingeben, werden die weiteren Informationen in dem Datensatz erstellt.

Datenbank-Verknüpfungen analysieren

Die Verknüpfungen in einer Datenbank spielen eine wichtige Rolle für Operationen in der Datenbank. Sie steuern den Informationsfluss zwischen den Tabellen.

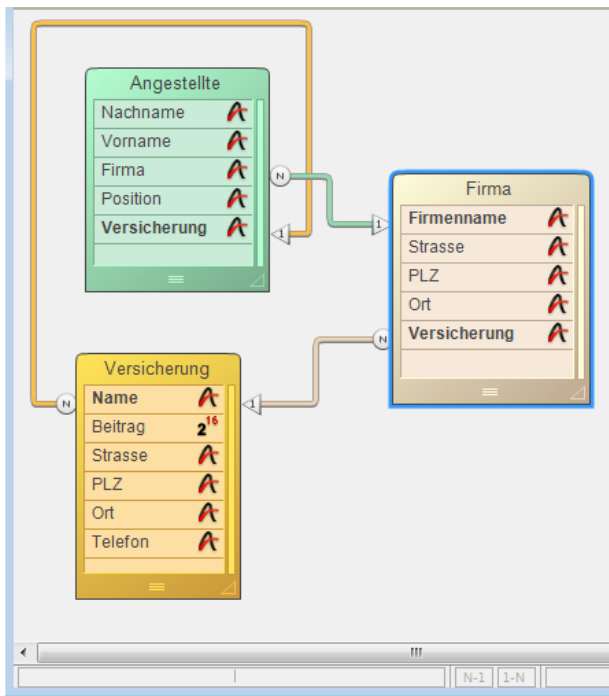
Wenn bei einer automatischen Verknüpfung ein Datensatz von der Festplatte in ein Eingabeformular geladen wird, wird der entsprechende Datensatz oder die Datensätze aus der verknüpften Tabelle ausgewählt. Wenn eine Verknüpfung aus einer verknüpften Tabelle nur einen Datensatz auswählt, wird dieser Datensatz von der Festplatte geladen. Wenn eine Verknüpfung aus einer verknüpften Tabelle mehr als einen Datensatz auswählt, wird für diese Tabelle eine neue aktuelle Auswahl erstellt und der erste Datensatz dieser aktuellen Auswahl wird von der Festplatte geladen. Der gerade geladene Datensatz wird als aktueller Datensatz dieser Tabelle bezeichnet.

In den Beispielen dieses Kapitels wurden die Verknüpfungen zwischen nicht mehr als drei Tabellen erstellt. In der Praxis bestehen Verknüpfungen oftmals zwischen mehreren Tabellen und werden, wie in einer Kette, alle nacheinander aktiviert. Jedesmal, wenn eine Verknüpfung aktiviert wird, erstellt 4D eine Auswahl von Datensätzen in der verknüpften Tabelle und lädt einen Datensatz. Dieser Datensatz wird für die Tabelle der aktuelle Datensatz und – falls für diese Tabelle eine automatische Verknüpfung besteht – erstellt 4D eine Auswahl und lädt einen aktuellen Datensatz in die nächste verknüpfte Tabelle in der Kette usw.

Wurden die Tabellenverknüpfungen nicht korrekt erstellt, kann der Informationsfluss durcheinander geraten oder abbrechen. Im folgenden sehen Sie Verknüpfungsstrukturen, die Sie vermeiden müssen.

Rekursive Verknüpfungen

Bei einer rekursiven Verknüpfung sind die Verknüpfungen so angelegt, dass sich die Übertragung von Informationen in einer Schleife totläuft. Die folgende Abbildung zeigt eine rekursive Verknüpfung, bei der die Tabelle [Angestellte] mit der Tabelle [Firma] verknüpft ist, die mit der Tabelle [Versicherung] verknüpft ist, von der eine Verknüpfung zurück zur Tabelle [Angestellte] geht.



Sobald ein Datensatz in die Tabelle [Angestellte] geladen wird, lädt 4D den verknüpften Datensatz aus der Tabelle [Firma]. Dieser wird der aktuelle Datensatz für die Tabelle [Firma], der als Folge den verknüpften Datensatz Versicherung aus der Tabelle [Versicherung] lädt.

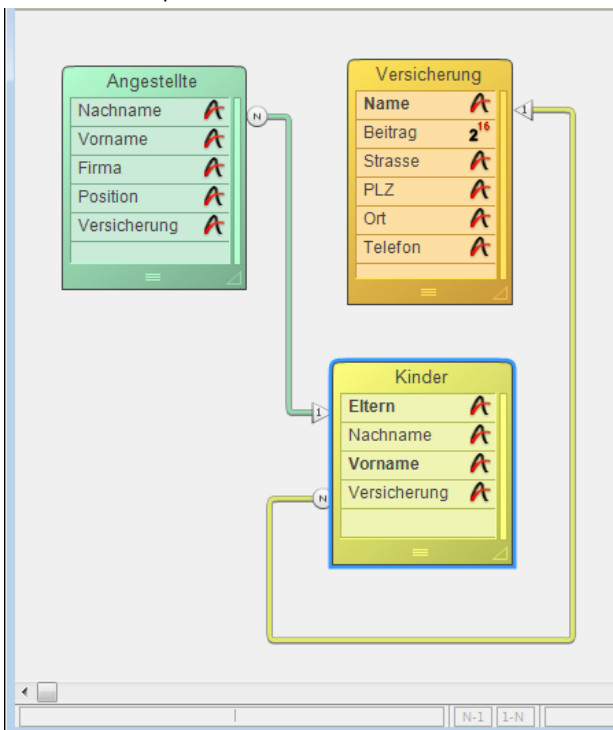
Falls die Tabellen-Verknüpfungen eine Fortsetzung zuließen, würden die mit dieser Versicherung verknüpften Datensätze, d.h. alle Angestellten, die bei dieser Firma versichert sind, in der Tabelle [Angestellte] ausgewählt werden. Der erste Datensatz dieser Auswahl würde zum aktuellen Datensatz werden. Beachten Sie, dass dieser aktuelle Datensatz nicht der gleiche ist, wie der aktuelle Datensatz, bei dem dieser Ablauf gestartet ist. In dieser Situation hat 4D keine Möglichkeit, festzustellen, welcher Datensatz tatsächlich der aktuelle Datensatz ist.

Findet 4D diese Art rekursiver Verknüpfung, werden die Tabellenverknüpfungen an der letzten Tabelle der Kette gestoppt. Bei diesem Beispiel wird die Verknüpfung zwischen der Tabelle [Versicherung] und der Tabelle [Angestellte] nicht ausgeführt.

Mehrfache Verbindungen zur gleichen Tabelle

Ein ähnlicher Konflikt zwischen aktuellen Datensätzen entsteht, wenn mehr als eine Verbindung zur gleichen Tabelle besteht. Da Sie nicht gleichzeitig mehr als einen aktuellen Datensatz haben können, können Sie eine automatische Tabellen-Verknüpfung, in der zwei oder mehr Tabellen mit der gleichen Tabelle verknüpft sind, nicht verwalten.

Die folgende Abbildung zeigt eine Datenbankstruktur, in der eine Tabelle und eine ihr zugeordnete Tabelle mit der gleichen Tabelle verknüpft sind



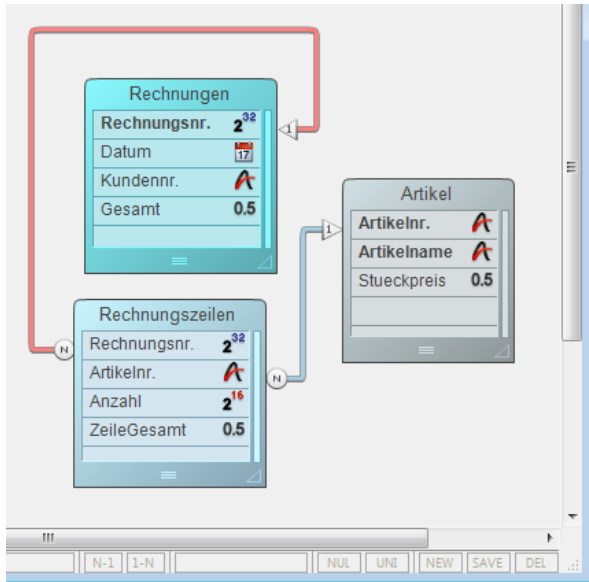
Gleichzeitig besteht aber eine Verknüpfung zwischen der Tabelle [Kinder] und der Tabelle [Versicherung]. Dies bedeutet, dass ein weiterer verknüpfter Datensatz in der Tabelle [Versicherung] geladen wird, der auf dem aktuellen Datensatz, dem ersten Datensatz in der Tabelle [Kinder] basiert. Wenn die Versicherung des Kindes mit der Versicherung der Eltern nicht übereinstimmt, führt diese Verknüpfungsstruktur zu Problemen.

Bei diesem Beispiel stoppt 4D den weiteren Ablauf der Verknüpfungen nicht. Es werden beide Verknüpfungen durchgeführt, aber nicht gleichzeitig. Wenn Sie diese Art von Struktur benutzen wollen, müssen Sie die Tabellenverknüpfungen manuell aktivieren und die Befehle der 4D Programmiersprache benutzen.

Ein weiteres Beispiel für eine relationale Struktur, die von automatischen Verknüpfungen nicht durchgeführt werden kann, ist eine Struktur, in der eine Tabelle mehr als eine Verknüpfung zu einer anderen Tabelle hat. Jedesmal, wenn der Benutzer ein verknüpftes Datenfeld in der einen Tabelle ändert, kann der aktuelle Datensatz in der anderen Tabelle wechseln. In dieser Situation lässt sich nicht sagen, welche Verknüpfung aktiviert werden soll.

Verknüpfungen von Mehrfach-Datensätzen

Da es in einer Tabelle nur einen aktuellen Datensatz gibt, werden die Verknüpfungen nicht für alle Datensätze einer Auswahl hergestellt. Nehmen wir als Beispiel die Datenbank *Rechnungen* mit folgender Struktur:



Wenn ein Datensatz in der Tabelle [Rechnungen] benutzt wird, wird in der Tabelle [Rechnungszeilen], die alle Zeilen für diese Rechnung enthält, eine Auswahl von Datensätzen erstellt. Der entsprechende Datensatz in der Tabelle [Artikel] wird jedoch nur für die erste Angabe in der Tabelle [Rechnungszeilen] ausgewählt. Die Auswahl der Tabelle [Artikel] enthält nicht die Informationen über alle Artikel der Rechnung, sondern nur die erste Angabe.

Wenn Sie jedoch [Rechnungszeilen] in ein Unterformular von [Rechnungen] bringen, ruft 4D jede Rechnungszeile einmal auf und aktiviert die Verknüpfung für jede von ihnen.

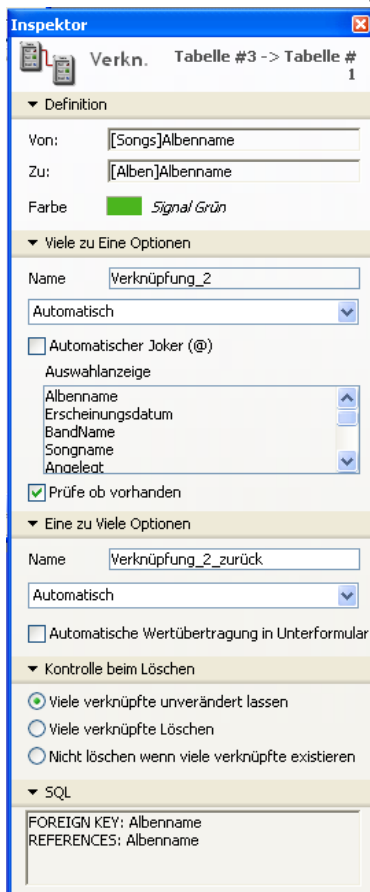
Verknüpfungseigenschaften

Eine Verknüpfung zeichnen Sie immer von der Viele-Tabelle zu der Eine-Tabelle.

Die verknüpften Datenfelder müssen identische oder kompatible Feldtypen haben. Folgende Feldtypen können Sie als Primär- oder verknüpfte Datenfelder einsetzen:

- String, d.h. Alpha und Text
- Zahl, d.h. Zahl, Ganzzahl, Lange Ganzzahl, Ganzzahl 64 bit oder Fließkomma
- Zeit
- Boolean
- Datum

Primärschlüsselfelder sind in der Regel indiziert, das ist jedoch nicht zwingend. Die Eigenschaften für die Verknüpfung legen Sie im **Inspektorfenster** fest. Einige Eigenschaften können Sie auch über das Kontextmenü definieren. Es erscheint durch rechten Mausklick auf die Verknüpfung.



Definition

Im Bereich Definition bestimmen Sie die Primärschlüsselfelder und die verknüpften Datenfelder:

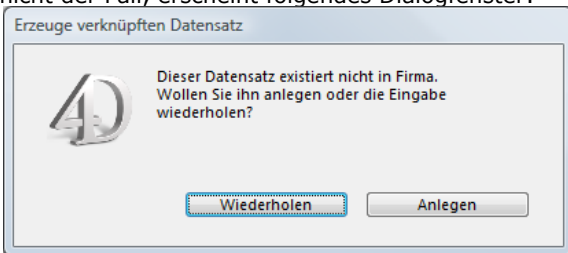
- **Von:** Das Datenfeld "Von" ist das verknüpfte Datenfeld in der Viele-Tabelle für diese Verknüpfung. Im Kontextmenü lautet es "Quellfeld."
- **Zu:** Das Datenfeld "Zu" ist das Primärschlüsselfeld in der Eine-Tabelle. Im Kontextmenü lautet es "Zielfeld." Sie zeichnen die Verknüpfungslinie vom verknüpften Datenfeld in der Viele-Tabelle zum Primärschlüsselfeld in der Eine-Tabelle.
- **Farbe:** Hier können Sie eine Farbe für die Verknüpfungslinie im Struktureditor auswählen. Die Farbe der Verknüpfungslinien beeinflusst keine andere Eigenschaft der Verknüpfung. Sie können die Farbe auch über das Kontextmenü der Verknüpfung festlegen.

Viele-zu-Eine Optionen

Die Optionen Viele-zu-Eine bestimmen, was passiert, wenn ein Datensatz aus der Viele-Tabelle geöffnet wird:

- **Name:** Hier können Sie einen Namen eintragen (optional).
Hinweis: Bei einem 4D Mobile Link wird mit diesem Namen im Modell auf Wakanda Seite ein entsprechendes relationales Attribut erstellt (siehe **Verknüpfungen verwenden** im Handbuch *4D Mobile*). Dafür müssen die JavaScript Regeln berücksichtigt werden.

- **Manuell/Automatisch:** Hier legen Sie fest, ob die Verknüpfung von der Viele-Tabelle zur Eine-Tabelle automatisch oder manuell ist. Haben Sie automatisch gewählt, passiert folgendes: Wird im Anwendungsmodus ein Datensatz aus der Tabelle [Angestellte] geöffnet, wird die verknüpfte Firma in der Tabelle [Firma] auch geöffnet. So erhält der Benutzer Informationen über die Firma, in welcher der Angestellte arbeitet. Standardmäßig ist die Option manuell gewählt. Sie müssen dann selbst das Laden und Ausgrenzen des verknüpften Eine-Datensatzes mit Befehlen der Programmiersprache durchführen. Weitere Informationen dazu finden Sie im Abschnitt **Automatische und manuelle Verknüpfungen**. Sie können die Option **Automatische Viele-Verknüpfung** auch über das Kontextmenü der Verknüpfung festlegen.
- **Automatischer Joker:** Ist dieses Optionsfeld aktiv, wird für jeden in das verknüpfte Datenfeld der Viele-Tabelle eingegebenen Wert unsichtbar das Jokerzeichen @ eingesetzt, wenn der Benutzer die Tabulatortaste drückt oder außerhalb des Datenfeldes klickt. Tippt er nur einen Teil ein, sucht 4D nach einem passenden Wert in der verknüpften Eine-Tabelle. Gibt es nur einen passenden Wert, wird die Eingabe vervollständigt. Passen mehrere Werte, zeigt 4D eine Liste mit den möglichen Werten zur Auswahl an.
- **Prüfe ob vorhanden:** Ist dieses Optionsfeld aktiv, erscheint ein Dialogfenster, in dem der Benutzer den verknüpften Eine-Datensatz neu anlegen kann, falls er noch nicht vorhanden ist. Tragen Sie einen Wert in ein verknüpftes Datenfeld der Viele-Tabelle ein, überprüft 4D, ob in der verknüpften Eine-Tabelle ein passender Datensatz vorhanden ist. Ist das nicht der Fall, erscheint folgendes Dialogfenster:



So kann der Benutzer einen passenden Datensatz in der Eine-Tabelle zum Datensatz in der Viele-Tabelle anlegen. Nehmen wir z.B. an, Sie haben eine Datenbank für Rechnungen mit einer Tabelle [Rechnung] und einer Tabelle [Kunden]. Geben Sie in die Tabelle [Rechnung] eine Rechnung ein und der dazugehörige Kunde ist in der Tabelle [Kunden] noch nicht angelegt, fragt 4D beim Bestätigen des Datensatzes in der Tabelle [Rechnung], ob Sie den noch nicht vorhandenen Datensatz in der Tabelle [Kunden] erstellen wollen.

Soll dieses Dialogfenster nicht erscheinen, deaktivieren Sie das Optionsfeld **Prüfe ob vorhanden**. Das ist sinnvoll, wenn Sie das Erstellen des verknüpften Eine-Datensatzes über eine Methode steuern.

- **Joker-Auswahlliste**

In der Liste der Joker-Felder können Sie ein zusätzliches Feld auswählen, das in der Liste der Werte erscheint und zwar, wenn der Benutzer bei der Dateneingabe in das verknüpfte Feld den Joker eingibt.

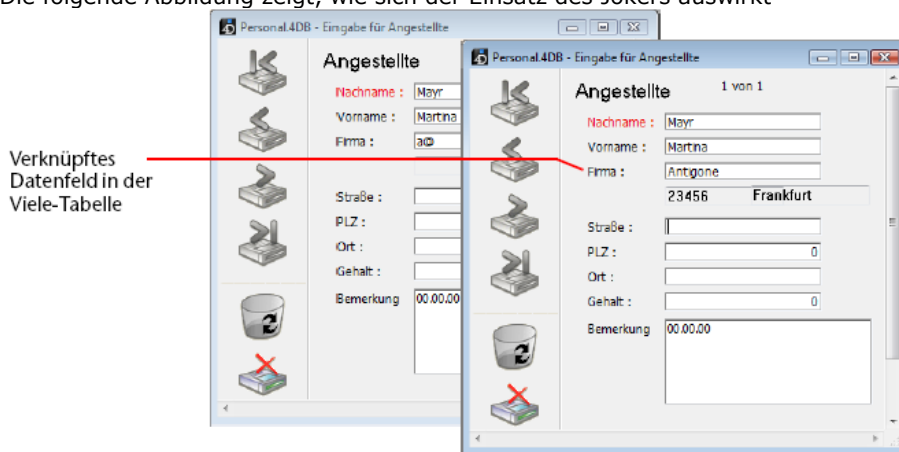
Das funktioniert folgendermaßen: Der Benutzer kann sich Werte in der Eine-Tabelle Werte ansehen, während er Daten in das verknüpfte Datenfeld der Viele-Tabelle eingibt. Hierzu verwendet er einfach im verknüpften Datenfeld das Jokerzeichen @. 4D sucht dann in der verknüpften Eine-Tabelle nach der entsprechenden Eingabe.

Sie können das Jokerzeichen auf zweierlei Art einsetzen: Um eine Teileingabe zu vervollständigen oder um eine Liste gültiger Eingaben anzuzeigen. Wenn die Liste angezeigt wird, kann der Benutzer die Eingabe aus der Liste wählen. Zum verknüpften Datenfeld kann ein zusätzliches Feld, das Joker Auswahlfeld, angezeigt werden.

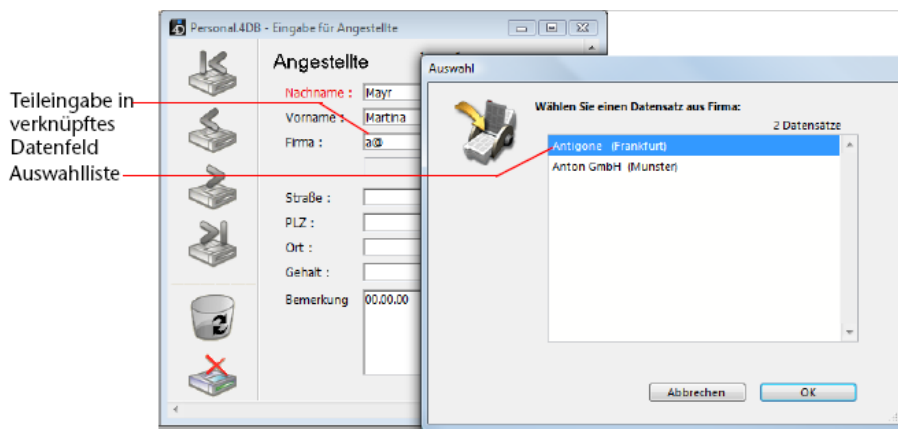
Nehmen wir z.B. an, der Benutzer erstellt einen Datensatz in der Tabelle [Angestellte]. Anstatt in das Firmenfeld Anton GmbH zu schreiben, tippt der Benutzer A@ und drückt die Tabulatortaste, um in das nächste Datenfeld zu gehen. Da @ bei 4D das Jokerzeichen ist, bedeutet diese Eingabe "Dieser Wert beginnt mit "A", gefolgt von irgend etwas".

4D sucht in der verknüpften Tabelle nach dem Datensatz, der zu dieser Eingabe passt. Wird ein Datensatz gefunden, vervollständigt es die Eingabe und wählt das nächste Datenfeld in der Reihenfolge der Dateneingabe.

Die folgende Abbildung zeigt, wie sich der Einsatz des Jokers auswirkt



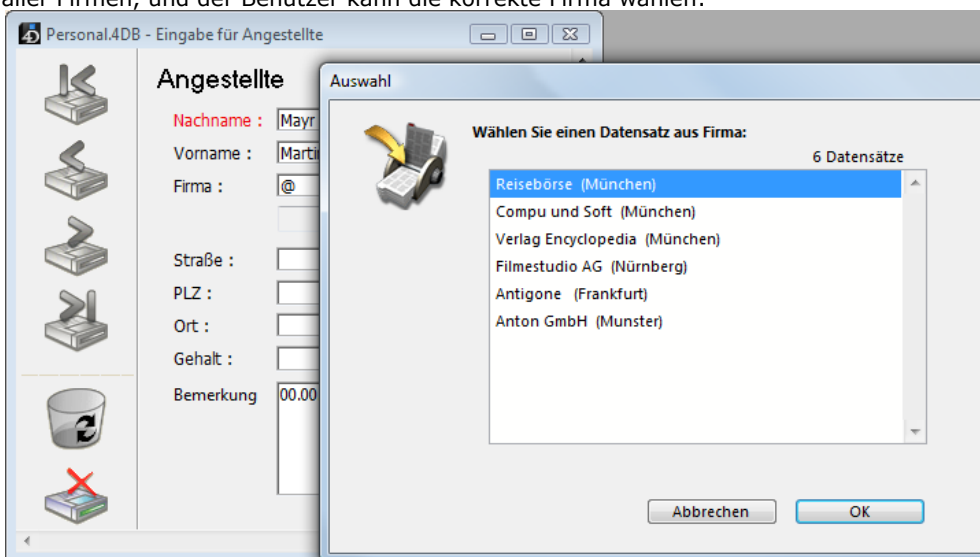
Findet 4D mehr als einen Eintrag, zeigt es eine Liste der Einträge, unter denen der Benutzer den richtigen auswählen kann. Die folgende Abbildung zeigt eine solche Liste:



Sie können als Hilfe für den Benutzer, welche Firma er wählen soll, ein zweites Datenfeld der Liste festlegen. Dieses zweite Datenfeld ist das Datenfeld der Joker-Auswahlliste, das Sie in der Inspektorpalette beim Erstellen der Verknüpfung festgelegt haben.

Nachfolgende Abbildung zeigt eine Liste von Firmen, in der Firmenname und Ort angezeigt werden. Dieses Joker-Auswahlfeld hilft dem Benutzer, der beispielsweise nicht weiß, ob die Firma Adams oder Akon heißt, sich aber erinnert, dass die Firma in München ansässig ist.

Um eine Liste aller Firmen aus der Tabelle [Firma] anzuzeigen, gibt der Benutzer nur @ ein. 4D zeigt dann eine Liste aller Firmen, und der Benutzer kann die korrekte Firma wählen:

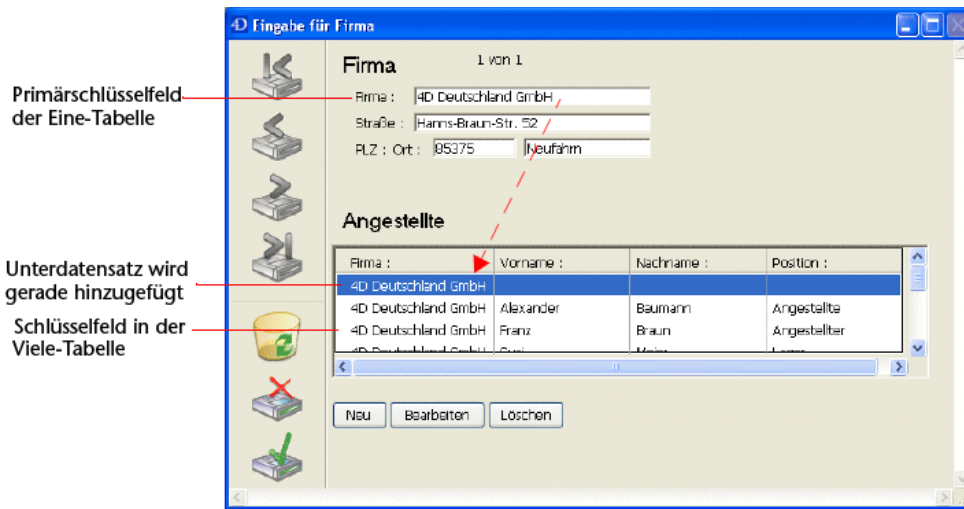


Sie können das Auswahlfenster auch vergrößern oder verkleinern.

Eine-zu-Viele Optionen

Die Option Eine-zu-Viele steuert automatische Verknüpfungen in die andere Richtung.

- **Name:** Hier können Sie einen Namen eintragen (optional).
Hinweis: Bei einem 4D Mobile Link wird mit diesem Namen im Modell auf Wakanda Seite ein entsprechendes relationales Attribut erstellt (siehe **Verknüpfungen verwenden** im Handbuch *4D Mobile*). Dafür müssen die JavaScript Regeln berücksichtigt werden.
- **Manuell/Automatisch:** Hier legen Sie fest, ob die Verknüpfung von der Eine-Tabelle zur Viele-Tabelle automatisch oder manuell ist. Haben Sie automatisch gewählt, wird automatisch die Verknüpfung hergestellt. Wird z.B. ein Datensatz der Tabelle [Firma] im Anwendungsmodus geöffnet, werden die verknüpften Datensätze der Tabelle [Angestellte] geladen. So kann 4D die Datensätze der Angestellten, die für diese Firma arbeiten, in einem Unterdatensatz anzeigen. Standardmäßig ist die Option **manuell** eingestellt. Weitere Informationen dazu finden Sie im Abschnitt **Automatische und manuelle Verknüpfungen**. Sie können die Option Automatische Verknüpfung auch über das Kontextmenü der Verknüpfung festlegen.
- **Automatische Wertübertragung:** Mit dieser Option können Sie den Wert des Primärschlüsselfeldes in der Eine-Tabelle bei der Dateneingabe automatisch dem verknüpften Datenfeld in der Viele-Tabelle zuweisen. Diese Option ist nur sinnvoll, wenn für die zurückgehende Verknüpfung die Option **automatisch** gewählt wurde. Diese Option beeinflusst die Dateneingabe, wenn ein Eingabeformular in einer Eine-Tabelle ein Unterformular in einer Viele-Tabelle hat. Weitere Informationen dazu finden Sie im Abschnitt **Unterformulare und Widgets**. Ist **Automatische Wertübertragung** aktiviert, kann der Benutzer in einem Unterformular Daten hinzufügen, d.h. in der verknüpften Viele-Tabelle, die dann den Datenfeldern der verknüpften Tabelle automatisch zugewiesen werden. In der Verknüpfung zwischen der Tabelle [Firma] und der Tabelle [Angestellte] ist [Firma] die Eine-Tabelle und [Angestellte] eine verknüpfte Viele-Tabelle. Für jede Firma gibt es einen Datensatz in der Tabelle [Firma] und mehrere Datensätze in der Tabelle [Angestellte]. Erscheint das Primärschlüsselfeld der Viele-Tabelle im Unterformular, können Sie die Auswirkung dieser Option betrachten: Der Feldwert erscheint bei jedem Hinzufügen eines Unterdatensatzes automatisch im Unterformular:



Die Option funktioniert aber auch, wenn das Fremdschlüsselfeld der verknüpften Tabelle nicht im Unterformular erscheint. Dann wird der Wert intern kopiert. Sie können das überprüfen, wenn Sie in den Seitenmodus gehen. Ist die Option **Automatische Wertübertragung** nicht markiert, werden die erstellten Unterdatensätze nicht automatisch dem passenden Datensatz in der Viele-Tabelle zugeordnet. In diesem Fall müssen Sie den Unterdatensatz selbst verknüpfen:

- Manuell, wenn das Schlüsselfeld angezeigt wird: Im obigen Beispiel genügt es, im Datenfeld Firma jedes Unterdatensatzes den Namen „4D Deutschland GmbH“ einzugeben
- Per Programmierung: Sie führen beim Erstellen des Unterdatensatzes folgenden Code aus

```
TableN]Field1 := [Table1]Field1
```

Hinweis: Ändern Sie in der Eine-Tabelle den verknüpften Feldwert, nachdem Sie diesen in der Viele-Tabelle erstellt haben, hat **Automatische Wertübertragung** keinen Einfluss. Sie müssen dann das verknüpfte Datenfeld entweder manuell oder per Programmierung zuweisen.

Löschkontrolle

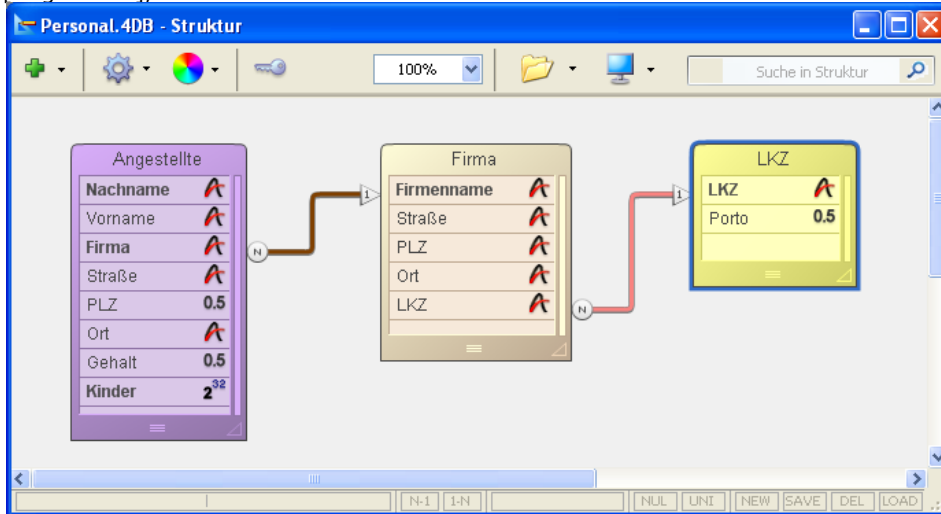
Die Optionen der Löschkontrolle regeln das Löschen von Datensätzen in der Viele-Tabelle, wenn Sie in der Eine-Tabelle einen Datensatz löschen. Im Normalfall kann der Benutzer in einer Tabelle keine Datensätze löschen, wenn diese Tabelle nicht die aktuelle Tabelle ist. Dies bedeutet z.B. wenn Sie in der Tabelle [Angestellte] Datensätze löschen wollen, müssen Sie diese erst zur aktuellen Tabelle machen. Hierzu wählen Sie die Tabelle in der Tabellenliste im Designmodus.

- **Viele verknüpfte unverändert lassen:** Ist diese Option aktiv, kann der Benutzer in der Eine-Tabelle einen Datensatz löschen, während die entsprechenden Datensätze in der Viele-Tabelle erhalten bleiben. Dies behält Datensätze in der Viele-Tabelle ohne einen entsprechenden Datensatz in der Eine-Tabelle bei. Damit sind Informationen aus der Eine-Tabelle nicht mehr verfügbar. In der Eine-Tabelle wird kein Datensatz geladen, wenn ein entsprechender Datensatz in der Viele-Tabelle zu diesem gelöschten Datensatz geladen wird.
- **Viele verknüpfte löschen:** Ist diese Option aktiv, löscht 4D automatisch alle verknüpften Datensätze in der Viele-Tabelle, wenn der Benutzer einen Datensatz in der Eine-Tabelle löscht. Diese Eigenschaft stellt sicher, dass keine Datensätze in der Viele-Tabelle "verwaisen", wenn der entsprechende Datensatz in der Eine-Tabelle gelöscht wird.
- **Nicht löschen, wenn viele verknüpfte existieren:** Ist diese Option aktiv, erlaubt 4D dem Benutzer nicht, einen Datensatz in der Eine-Tabelle zu löschen, wenn verknüpfte Datensätze in der Viele-Tabelle bestehen. Diese Eigenschaft stellt sicher, dass Datensätze nicht fälschlich gelöscht werden. Bitte beachten Sie, dass Sie in der Viele-Tabelle Datensätze löschen können, gleichgültig, welche Option Sie wählen.

Die Optionsfelder **Viele verknüpfte löschen** und **Nicht löschen, wenn viele verknüpfte existieren** verstärken die in der Datenbanktheorie sogenannte referentielle Integrität. Sobald diese besteht, stellt 4D sicher, dass jeder Datensatz in einer verknüpften Viele-Tabelle genau einem Datensatz in der verknüpften Eine-Tabelle zugeordnet wird.

Wenn Sie mit mehreren verknüpften Tabellen arbeiten, wird die Löschkontrolle wie in einer Kette für jede Verknüpfung gesetzt. Nehmen wir an, Sie haben z.B. die unten gezeigte Struktur. Wenn Sie in der Tabelle [LKZ], d.h. einer Eine-Tabelle, ein Landeskürzel löschen, und für jede Verknüpfung Viele verknüpfte löschen gewählt haben, löscht 4D zuerst die Datensätze der entsprechenden Firmen in der Tabelle [Firma] und anschließend die Datensätze aller Angestellten in der Tabelle

[Angestellte], die in dieser Firma arbeiten.



Wenn 4D widersprüchliche Einstellungen für die Löschkontrolle feststellt, lässt es Löschen nicht zu. Wenn Sie z.B. für die Verknüpfung zwischen der Tabelle [Firma] mit der Tabelle [LKZ] **Viele verknüpfte löschen** gewählt haben, für die Verknüpfung zwischen der Tabelle [Angestellte] und der Tabelle [Firma] aber **Nicht löschen, wenn viele verknüpfte existieren**, erfolgt kein Löschen und die Datensätze in der Tabelle [Firma] und der Tabelle [Angestellte] bleiben bestehen.

SQL

Der Bereich SQL der Inspektortaube zeigt Informationen, die beim Bearbeiten der Struktur mit der SQL Programmiersprache hilfreich sind. Für Verknüpfungen erscheinen die Eigenschaften FOREIGN KEY und REFERENCES.

Strukturdefinitionen exportieren und importieren

In 4D können Sie die Strukturdefinitionen der Datenbank als XML- oder HTML-Datei exportieren. Genauso können Sie mit einer im XML-Format gesicherten Strukturdefinition im Handumdrehen eine neue 4D Datenbank erstellen. Das hat folgende Vorteile:

- Strukturen lassen sich in eigenen Formaten darstellen (Berichte, Tabellen, etc.) oder in anderen Umgebungen analysieren
- Datenbanken lassen sich über Definitionsdateien erstellen

Format einer 4D Strukturdefinition

Die Strukturdefinitionen in 4D basieren auf dem XML-Format. Sie können eine Strukturdefinition mit einem einfachen Texteditor öffnen und ansehen. Das XML-Format ermöglicht auch, jede Art von Verwendung voranzuplanen, insbesondere via XSL Transformationen. Außerdem verwendet 4D eine .XSL Datei, um die Strukturdefinition in HTML-Format zu exportieren.

Eine Strukturdefinition enthält Tabellen, Felder, Indizes und Verknüpfungen zusammen mit ihren Attributen und verschiedenen Merkmalen, die für eine komplette Definition der Struktur erforderlich sind. Die interne "Grammatik" der 4D Strukturdefinitionen wird über DTD Dateien dokumentiert — die auch zum Bestätigen der XML Dateien dient. Die von 4D verwendeten DTD Dateien werden im **DTD** Ordner abgelegt, der neben der 4D Anwendung liegt. Dabei dienen die Dateien **base_core.dtd** und **common.dtd** zum Definieren der Struktur. Für weitere Informationen zu den 4D Strukturdefinitionen sehen Sie sich diese Dateien und die darin enthaltenen Kommentare an.

Strukturdefinition exportieren

Mit 4D können Sie eine Struktur im XML- oder HTML-Format exportieren. Wählen Sie das für Ihre Bedürfnisse am besten geeignete Format:

- **XML Format:** Diese XML-Datei können Sie mit einem einfachen Text-editor öffnen und ansehen oder in unterschiedlicher Weise verwenden, z.B. für eigene XSL Transformation, Import oder Analyse in einer anderen Software. Wählen Sie dieses Format, wenn Sie die Strukturdefinition zum Erstellen einer neuen Datenbank verwenden wollen.
- **HTML Format:** Dieses Format ermöglicht, die Struktur als Bericht darzustellen, den Sie über einen Browser ansehen und ausdrucken können.

Um eine Strukturdefinition in XML zu exportieren:

1. Wählen Sie im 4D Menü **Datei/Ablage** den Eintrag **Exportieren > Strukturdefinition in XML Datei**. Auf dem Bildschirm erscheint der Standarddialog **Speichern unter**.
2. Geben Sie Name und Ort des Exports an und bestätigen das Dialogfenster.

Um eine Strukturdefinition in HTML zu exportieren:

1. Wählen Sie im 4D Menü **Datei/Ablage** den Eintrag **Exportieren > Strukturdefinition in HTML Datei**. Auf dem Bildschirm erscheint das Dialogfenster *Ordner suchen*. Hier bestimmen Sie den Ort zum Speichern der HTML-Dateien.
2. Klicken Sie am unteren Rand auf die Schaltfläche **Neuen Ordner erstellen** oder wählen Sie einen vorhandenen Ordner aus. 4D erstellt am angegebenen Ort automatisch den Ordner *MeineStruktur.4DB Structure export* mit den exportierten Elementen. *MeineStruktur* ist der Name der Strukturdatei. Das Ergebnis des Exports erscheint direkt in einem Fenster des Standardbrowsers. Die Strukturdefinitionen im HTML-Format erscheinen wie folgt

Structure Definition of MyMusic Database
[Expand all](#) / [Collapse all](#)

[-] Albums

■ Triggers

On Load	On Save New Record	On Delete	On Save

■ Table (9 fields)

Field name	Type	Indexed	Unique	Not Null	Never Null	Invisible	Enterable	Modifiable	Mandatory	Multiline	Compressed
1. Album Title	A Alpha 45	B-tree			✓		✓	✓			
2. Musician	A Alpha 45	B-tree			✓		✓	✓			
3. Format	A Alpha 15	B-tree			✓		✓	✓			
4. Music Category	A Alpha 15	B-tree			✓		✓	✓			
5. Year Recorded	2 ¹⁶ Integer				✓		✓	✓			
6. Date Purchased	17 Date				✓		✓	✓			
7. Purchase Price	0.5 Real				✓		✓	✓			
8. Notes	T Text				✓		✓	✓			
9. Performed by	A Alpha 45				✓		✓	✓			

■ Relations

From	To
[Albums]Musician	[Musicians]Musician Name

Die XSL Transformation anpassen

Um die HTML-Seiten der Strukturdefinition zu erstellen, führt 4D standardmäßig XSL Transformationen mit der Datei "Structure_to_html.xml" aus, die im Unterordner /Resource/language.lproj des Programms liegt.

Hinweis: Existiert diese Datei nicht, ist im Exportdialog das Exportieren in HTML nicht verfügbar.

Sie können diese Umwandlungen mit einer eigenen XSL-Stilvorlage beliebig an Ihre Bedürfnisse anpassen. Dazu erstellen Sie eine Datei mit Namen "Structure_to_html.xml" (Sie können die Standarddatei duplizieren) und legen diese auf dieselbe Ebene wie die Datei .4db. 4D verwendet dann diese Datei, um die Strukturdefinition im HTML-Format zu erstellen.

Datenbank aus einer Strukturdefinition

Mit Strukturdefinitionen, die Sie im XML Format exportieren, können Sie sofort neue identische Datenbanken erstellen. Die Strukturdatei lässt sich sozusagen als Strukturvorlage betrachten, die Sie beliebig duplizieren können.








Sie können die XML Strukturdefinition unverändert verwenden oder zuvor über einen XML Editor verändern. Sie können jede Art von Mechanismus zum Erzeugen von Strukturen per Programmierung in Betracht ziehen.

Da das interne Format der XML Dateien der 4D Strukturdefinition öffentlich ist (siehe Abschnitt oben "Format einer 4D Strukturdefinition"), können Sie diese Art Datei auch aus anderen Datenbankumgebungen oder mit jedem Design-Programm erstellen, um daraus automatisch 4D Datenbanken zu erzeugen.

Um eine Datenbank aus einer Strukturdefinition zu erstellen:

- Wählen Sie im Menü **Datei/Ablage** von 4D den Befehl **Neu > Struktur von Strukturdefinition**. Ein Standard-Öffnen Dialog erscheint, in dem Sie die zu öffnende Definitionsdatei festlegen können. Sie müssen ein XML-Format wählen, das die "Grammatik" der 4D Strukturdefinitionen berücksichtigt (das Programm bestätigt die Datei via DTD).
- Wählen Sie eine XML Datei der Strukturdefinition und klicken dann auf die Schaltfläche **OK**. Auf dem Bildschirm erscheint eine Dialogbox, in der Sie Name und Speicherort der zu erstellenden Datenbank definieren.
- Wählen Sie Name und Speicherort der zu erstellenden Datenbank und klicken dann auf die Schaltfläche **Speichern**. Ist die XML Datei gültig, schließt 4D die aktuelle Datenbank (sofern das zutrifft) und erstellt anhand der Strukturdefinitionsdatei eine neue Struktur und zeigt das Explorer Fenster an. Standardmäßig wird auch eine Datendatei angelegt.

ORDA

-  Überblick
-  Glossar
-  Datastore
-  Dataclasses
-  Entities
-  Entity-Selections
-  Entity sperren

Was ist ORDA?

ORDA steht für **Object Relational Data Access**. Es ist eine erweiterte Technologie, die es erlaubt, eine Datenbank als Objekt zu verwenden - durch die Sprache oder mit Widgets der Benutzeroberfläche.

Die Relationen werden transparent in das Konzept einbezogen, in Kombination mit *lazy loading*, was dem Entwickler alle typischen Schwierigkeiten bei der Datenauswahl oder -übertragung erspart.

Bei ORDA erfolgt der Datenzugriff über eine Abstraktionsschicht, genannt *Datastore* (Datenspeicher). Das ist ein Objekt, das eine Schnittstelle zum Datenbankmodell und den Daten über 4D Objekte bietet. Tabellen werden auf *Dataclass-Objekte* abgebildet, Felder und Relationen sind *Dataclass-Attribute* und Datensätze sind vergleichbar mit *Entities*. Eine Übersicht finden Sie unter [Glossar](#).

Warum ORDA verwenden?

Anstatt Informationen als Tabellen, Datensätze und Felder darzustellen, verwendet ORDA einen alternativen Ansatz, der die Daten genauer auf reale Konzepte abbildet.

Stellen Sie sich die Fähigkeit vor, eine relationale Struktur zu denormalisieren, ohne die Effizienz zu beeinträchtigen. Stellen Sie sich vor, alles über die Business-Objekte in Ihrer Anwendung so zu beschreiben, dass die Verwendung der Daten einfach und unkompliziert wird und ein vollständiges Verständnis der relationalen Struktur entfällt.

In einem *Datastore* kann eine einzelne *Dataclass* alle Elemente enthalten, die eine traditionelle relationale Datenbanktabelle bilden, aber auch Werte von verwandten übergeordneten Entities sowie direkte Verweise auf verwandte *Entities* und *Entity-Selections*.

Eine Abfrage gibt eine Liste von Entities zurück, die als *Entity-Selection* bezeichnet wird und die die Rolle des Zeilen-Sets einer SQL-Abfrage erfüllt. Der Unterschied besteht darin, dass jede Entity "weiß", wo sie im Datenmodell hingehört und ihre Beziehung zu allen anderen Entities "versteht". Das bedeutet, dass ein Entwickler in einer Abfrage nicht erklären muss, wie er die verschiedenen Informationen zueinander in Beziehung setzt, noch in einem Update, wie er geänderte Werte in die relationale Struktur zurückschreibt.

Darüber hinaus können ORDA-Objekte wie *Entity-Selections* oder *Entities* einfach an Objekte wie Listboxen oder Variablen gebunden werden. Kombiniert mit leistungsstarken Funktionen wie **This** und **Form** ermöglichen sie den Aufbau moderner und modularer Schnittstellen, die auf Objekten und Collections basieren.

Definitionen

Grundsätzlich behandelt ORDA **Objekte**. In ORDA sind alle wichtigen Konzepte, einschließlich des *Datastore*, über Objekte verfügbar. ORDA-Objekte werden bei Bedarf durch 4D Methoden erzeugt und instanziiert (Sie müssen sie nicht anlegen).

Beachten Sie jedoch, dass Sie diese normalerweise in 4D Objektvariablen speichern müssen, wie alle anderen Objekte (deklariert mit dem Befehl **C_OBJECT**). Objekte in ORDA können wie 4D Standardobjekte behandelt werden (siehe **Objekte (Sprache)**), profitieren aber automatisch von spezifischen Eigenschaften und Methoden.

In ORDA sind folgende Objekte verfügbar:

- **Datastore:** Der Datenspeicher ist das Schnittstellenobjekt zur Datenbank. Es erstellt eine Darstellung der gesamten Datenbank als Objekt und ist über den Befehl **ds** verfügbar.
- **Dataclass:** Eine Datenklasse ist vergleichbar mit einer Tabelle. Sie wird als Objektmodell verwendet und verweist auf alle Felder als Attribute, einschließlich relationaler Attribute (Attribute, die auf Beziehungen zwischen Datenklassen aufbauen). Relationale Attribute können wie beliebige Attribute in Abfragen genutzt werden.
- **Attribute:** Eigenschaften von Datenklassen sind Attributobjekte, die die zugrundeliegenden Felder oder Relationen beschreiben.
- **Entity-Selection:** Eine Entity-Auswahl verweist auf eine oder mehrere Entities aus einer Datenklasse. Sie wird in der Regel als Ergebnis einer Abfrage erstellt.
- **Entity:** Eine Entity ist vergleichbar mit einem Datensatz. Sie ist genaugenommen ein Objekt, das auf einen Datensatz in der Datenbank verweist.

Voraussetzungen

Damit Sie ORDA in Ihren 4D Datenbanken verwenden können, müssen Sie folgende Anforderungen und Spezifikationen beachten:

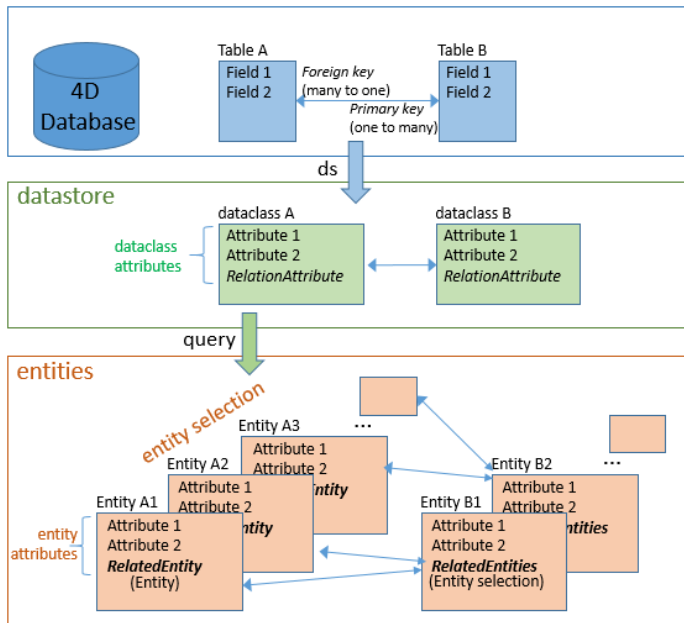
- In Anwendungen, die aus Versionen vor v17 konvertiert wurden, müssen folgende Optionen zur Kompatibilität markiert sein:
 - **Verwende Objektnotation, um auf Objekteigenschaften zuzugreifen (Unicode nötig)**
 - **Verwende Datumstyp statt ISO Datumsformat in Objekten**

Weitere Informationen dazu finden Sie auf der [Seite Kompatibilität](#).

- Tabellen ohne Primärschlüssel oder mit zusammengesetztem Primärschlüssel sind in Datastore nicht verfügbar.
- Untertabellen werden nicht unterstützt.
- Namen für Tabellen, Felder und Verknüpfungen müssen mit den standardmäßigen Namenskonventionen für Objekte übereinstimmen. Weitere Informationen dazu finden Sie im Abschnitt **Identifizier für Objekteigenschaft**.
- Wenn Sie eine Tabelle, ein Feld oder eine Verknüpfung in der 4D Struktur umbenennen, müssen Sie die 4D Datenbank neu starten, damit die Änderungen in der ORDA Schicht reflektiert werden.

Hinweis: ORDA berücksichtigt weder die Option "Unsichtbar" für Tabellen oder Felder, noch die virtuelle Struktur, die durch die Befehle **SET TABLE TITLES** oder **SET FIELD TITLES** definiert ist.

Die wichtigsten Konzepte im Überblick



Attribute

Ein Attribut ist die kleinste Speicherzelle einer relationalen Datenbank. Verwechseln Sie dabei nicht Dataclass-Attribute und Entity-Attribute:

- In einem Dataclass-Objekt ist jede Eigenschaft ein Attribut, das ein entsprechendes Feld in der entsprechenden Tabelle (gleicher Name und Typ) abbildet. Siehe auch **Verknüpfungsattribute**.
- In einem Entity-Objekt sind Attribute Eigenschaften, die Werte für diese Attribute im Datastore enthalten.

Hinweis: Attribute und Eigenschaften sind ähnliche Konzepte. "Attribut" wird eher für Dataclass-Eigenschaften verwendet, die Daten speichern, während "Eigenschaft" allgemeiner ist und einen Datenteil definiert, der in einem Objekt gespeichert ist.

AttributePath

Ein **AttributePath** ist der Pfad eines Attributs innerhalb einer bestimmten Dataclass oder Entity. Siehe auch **PropertyPath**.

Dataclass

Eine Dataclass ist ein Objektmodell, das Daten beschreibt. Tabellen in der vom Datastore bereitgestellten Datenbank werden über Dataclasses behandelt. Jede Tabelle hat eine entsprechende Dataclass mit dem gleichen Namen. Jedes Feld der Tabelle ist ein Attribut der Dataclass.

Eine Dataclass bezieht sich auf einen einzelnen Datastore.

Datastore

Ein Datastore ist das Schnittstellenobjekt, das ORDA zur Verfügung stellt, um auf eine Datenbank zu verweisen und auf deren Daten zuzugreifen. Die Hauptdatenbank, die vom Befehl **ds** zurückgegeben wird, ist als Datastore verfügbar.

Ein Datastore enthält:

- eine Verbindung zur 4D Datenbank
- einen Satz **Dataclasses** zum Arbeiten mit der Datenbank

Wenn der Datastore über den Code verwaltet wird, ist er ein Objekt, dessen Eigenschaften alle definierten Dataclasses sind. Ein Datastore verweist nur auf eine einzige Datenbank.

Hinweise:

- Ein Datastore verweist nur auf Tabellen mit einem einzigen Primärschlüssel, dagegen nicht auf Tabellen ohne Primärschlüssel oder mit zusammengesetzten Primärschlüsseln.
- Attribute vom Typ BLOB werden nicht im Datastore verwaltet.

Tiefe Kopie (deep copy)

Eine tiefe Kopie dupliziert ein Objekt und alle darin enthaltenen Referenzen. Nach einer tiefen Kopie enthält die kopierte Collection doppelte Elemente und damit neue Verweise auf alle Originalelemente. Siehe auch **Flache Kopie (shallow copy)**.

ds

ds ist der 4D Befehl, der eine Referenz auf ein Datastore-Objekt zurückgibt. Das ist die Referenz zum Datastore, der auf die 4D Hauptdatenbank zugreifen kann.

Entity

Eine Entity ist ein Objekt, das einem Dataclass-Modell entspricht. Eine Entity enthält die gleichen Attribute wie die Dataclass.

Eine Entity kann als Instanz der Dataclass angesehen werden - ähnlich wie der Datensatz in der Tabelle - die der Dataclass im dazugehörigen Datastore entspricht. Eine Entity enthält jedoch auch verknüpfte Daten. Sie dient zum Verwalten von Daten (Anlegen, Aktualisieren, Löschen).

Entity-Selection

Eine Entity-Selection ist ein Objekt. Bei der Abfrage des Datastore wird eine Entity-Selection zurückgegeben. Dies ist ein Satz Referenzen auf Entities, die sich auf dieselbe Dataclass beziehen.

Eine Entity-Selection enthält:

- einen Satz von 0 bis X Entity-Referenzen
- immer eine Eigenschaft `length`
- Die Eigenschaften `queryPlan` und `queryPath` (wenn beim Abfragen angefordert).

Eine Entity-Selection kann auch leer sein.

Lazy loading

Da Entities als Referenzen verwaltet werden, werden Daten nur bei Bedarf geladen, d.h. beim Zugriff im Code oder über Interface-Widgets. Dieses Optimierungsprinzip wird *Lazy Loading* genannt.

Member-Funktion

ORDA-Objekte wie Datastores, Dataclasses, Entity-Selections und Entities definieren *Klassen* von Objekten. Sie bieten spezifische Methoden, um direkt mit ihnen zu interagieren. Sie werden auf einer Instanz des Objekts aufgerufen. Solche Methoden werden auch *Member-Funktionen* genannt.

Zum Beispiel ist die **query()**-Methode eine Dataclass-Member-Funktion. Wenn Sie ein Dataclass-Objekt in der Variablen `$myClass` gespeichert haben, können Sie schreiben:

```
$myClass.query("name = smith")
```

Datentyp Gemischt

In dieser Dokumentation wird der Datentyp "Gemischt" verwendet, um die verschiedenen Datentypen zu bezeichnen, die in Dataclass-Attributen gespeichert werden können. Das sind:

- Zahl
- Text
- Null
- Boolean
- Datum
- Objekt
- Collection
- Bild (*)

(*) Der Typ Bild wird nicht von statistischen Methoden wie **entitySelection.max()** unterstützt.

Optimistisches Sperrverfahren (Optimistic Lock)

Im Modus "optimistic Lock" werden Entities nicht explizit gesperrt, bevor sie aktualisiert werden. Eine Entity hat einen internen Stempel, der bei jedem Speichern der Entity auf der Festplatte automatisch hochgezählt wird. Die Methoden **entity.save()** oder **entity.drop()** geben einen Fehler zurück, wenn der Stempel der geladenen Entity nicht mit dem Stempel der Entity auf der Festplatte übereinstimmt oder die Entity gelöscht wurde. Optimistische Sperrverfahren sind nur in der ORDA-Implementierung verfügbar. Das Gegenteil davon ist "pessimistic Lock".

Pessimistisches Sperrverfahren (Pessimistic Lock)

Der Modus "pessimistic Lock" bedeutet, dass eine Entity vor ihrem Zugriff über die Methode `entity.lock()` gesperrt wird. Andere Prozesse können die Entity weder aktualisieren noch löschen, bis sie entsperrt ist. Die klassische 4D Sprache erlaubt nur pessimistische Sperrverfahren. Das Gegenteil davon ist "optimistic Lock".

Eigenschaft

Siehe **Attribute**.

Hinweis: *Attribute* und *Eigenschaften* sind ähnliche Konzepte. "Attribut" wird eher verwendet, um Dataclass-Eigenschaften zu bezeichnen, die Daten speichern, während "Eigenschaft" allgemeiner ist und einen Teil von Daten definiert, der in einem Objekt gespeichert ist.

PropertyPath

Ein **PropertyPath** ist der Pfad zu einer Eigenschaft in einem bestimmten Objekt. Wenn die Eigenschaft in mehrere Ebenen verschachtelt ist, wird jede Ebene durch einen Punkt getrennt.

Verknüpfte Dataclass

Dies sind Dataclasses, die durch Verknüpfungsattribute verbunden sind.

Verknüpfungsattribute

Verknüpfungsattribute dienen zum Definieren von Verknüpfungen zwischen Tabellen (Viele-zu-Eine und Eine-zu-Viele).

- Viele-zu-Eine Verknüpfung (dataclassA bezieht sich auf ein Vorkommen von dataclassB): ein Verknüpfungsattribut ist in dataclassA verfügbar und verweist auf eine Instanz von dataclassB.
- Eine-zu-Viele Verknüpfung (ein Vorkommen von dataclassB verweist auf mehrere Vorkommen von dataclassA): ein Verknüpfungsattribut ist in dataclassB verfügbar und verweist auf mehrere Instanzen von dataclassA.

Eine Dataclass kann rekursive Verknüpfungsattribute haben.

In einer Entity kann der Wert eines Verknüpfungsattributs eine Entity oder eine Entity-Selection sein.

Verknüpfte Entities

Eine verknüpfte Entity kann als Instanz eines Verknüpfungsattributs in einer Dataclass angesehen werden.

Entity-Selections können sich auf verknüpfte Entities beziehen, gemäß den Verknüpfungsattributen der dazugehörigen Dataclasses.

Flache Kopie (shallow copy)

Eine *flache Kopie* dupliziert nur die Struktur der Elemente und behält die gleichen internen Referenzen. Nach einer flachen Kopie teilen sich zwei Collections die einzelnen Elemente. Siehe auch **Tiefe Kopie**.

Stempel

Stempel kommen bei optimistischen Sperrverfahren zum Einsatz. Alle Entities haben einen internen Zähler, den Stempel, der bei jedem Speichern der Entity hochgezählt wird. Durch automatisches Vergleichen der Stempel einer zu speichernden Entity und ihrer Version auf der Festplatte kann 4D gleichzeitiges Ändern derselben Entity verhindern.

Was ist ein Datastore?

Ein Datastore (Datenspeicher) ist das von ORDA bereitgestellte Oberflächenobjekt, um auf eine Datenbank zu verweisen und auf deren Modell und Daten zuzugreifen. In 4D v17 ist nur die lokale Datenbank, die vom Befehl **ds** zurückgegeben wird, als Datastore verfügbar.

Hinweis: Die auf den Datastore bezogenen Methoden und Eigenschaften werden im Kapitel **ORDA - DataStore** des Handbuchs *4D Programmiersprache* beschrieben.

Ein Datastore besteht aus einem **Modell** und **Daten**:

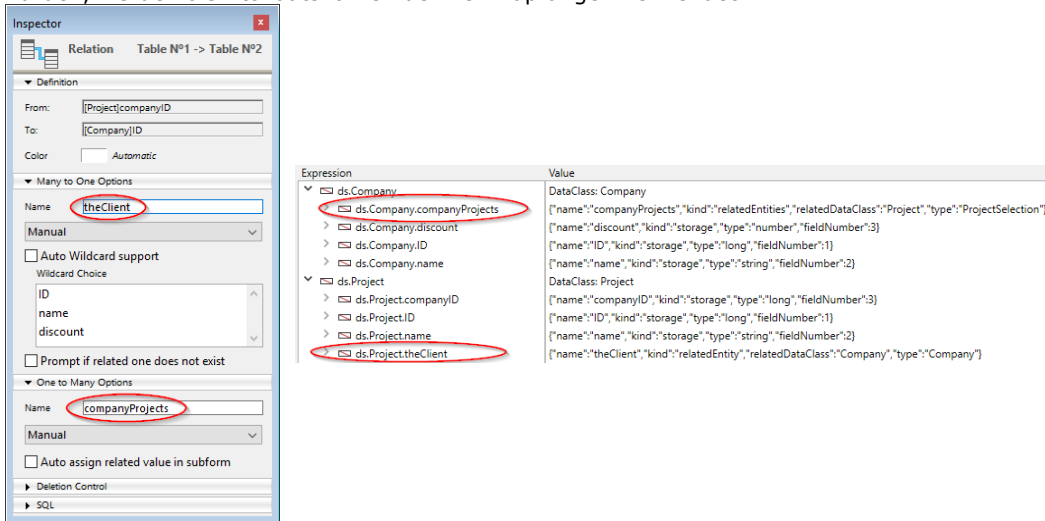
- Das Modell enthält und beschreibt alle Dataclasses, aus denen sich der Datastore zusammensetzt. Es ist unabhängig von der zugrunde liegenden Datenbank selbst.
- Daten beziehen sich auf die Informationen, die in diesem Modell verwendet und gespeichert werden sollen. Beispielsweise sind Namen, Adressen und Geburtsdaten von Mitarbeitern Datenteile, mit denen Sie in einem Datastore arbeiten können.

Wenn der Datastore über den Code behandelt wird, ist er ein Objekt, dessen Eigenschaften alle definierten Dataclasses sind. Ein Datastore bezieht sich nur auf eine einzelne lokale oder remote Datenbank.

Der ds Datastore

Der Befehl **ds** gibt eine Referenz auf den Standard-Datastore der Anwendung zurück. Wenn Sie diesen Befehl aufrufen, weist 4D automatisch jede Tabelle und jedes Feld der 4D Datenbank als Attribute des zurückgegebenen Objekts aus:

- Tabellen werden auf Dataclasses abgebildet
- Felder werden auf Speicherattribute abgebildet
- Datensätze werden auf Entities abgebildet
- Verknüpfungen werden auf Verknüpfungsattribute abgebildet - Verknüpfungsnamen, die im Struktureditor definiert wurden, werden als Attributnamen der Verknüpfungen verwendet.



Expression	Value
ds.Company	DataClass: Company
ds.Company.companyProjects	[{"name":"companyProjects","kind":"relatedEntities","relatedDataClass":"Project","type":"ProjectSelection"}]
ds.Company.discount	[{"name":"discount","kind":"storage","type":"number","fieldNumber":3}]
ds.Company.ID	[{"name":"ID","kind":"storage","type":"long","fieldNumber":1}]
ds.Company.name	[{"name":"name","kind":"storage","type":"string","fieldNumber":2}]
ds.Project	DataClass: Project
ds.Project.companyID	[{"name":"companyID","kind":"storage","type":"long","fieldNumber":3}]
ds.Project.ID	[{"name":"ID","kind":"storage","type":"long","fieldNumber":1}]
ds.Project.name	[{"name":"name","kind":"storage","type":"string","fieldNumber":2}]
ds.Project.theClient	[{"name":"theClient","kind":"relatedEntity","relatedDataClass":"Company","type":"Company"}]

Die folgenden Regeln werden bei der Konvertierung angewendet:

- Tabellen-, Feld- und Verknüpfungsnamen werden auf Objekteigenschaftsnamen abgebildet. Wenn Sie die "Punktnotation" in ORDA verwenden wollen, stellen Sie sicher, dass diese Namen den allgemeinen Regeln zum Benennen von Objekten entsprechen, wie im Abschnitt **Identifizier für Objekteigenschaft** beschrieben.
- **Hinweis:** Die Eigenschaft "Manuell" oder "Automatisch" von Verknüpfungen hat keine Auswirkung in ORDA.
- Ein Datastore bezieht sich nur auf Tabellen mit einem einzigen Primärschlüssel (siehe **Primärschlüssel**). Die folgenden Tabellen werden nicht referenziert:
 - Tabellen ohne Primärschlüssel
 - Tabellen mit zusammengesetzten Primärschlüsseln
- Attribute vom Typ BLOB werden nicht im Datastore verwaltet. Sie werden in Entities als Null zurückgegeben und können nicht zugeordnet werden.

Über das Objekt Datastore selbst

Das Datastore-Objekt selbst kann nicht als Objekt kopiert werden:

```
$mydatastore:=OB Copy(ds) //gibt Null zurück
```

Die Eigenschaften des Datastore sind dagegen aufzählbar:


```
ARRAY TEXT($prop;0)
```

```
OB GET PROPERTY NAMES(ds;$prop)
```

```
// $prop enthält die Namen aller Dataclasses
```

Definition

Dataclasses bieten Objektschnittstellen zu Datenbanktabellen. Alle Dataclasses in einer 4D Anwendung sind als Eigenschaft des Datastore **ds** verfügbar. Betrachten Sie beispielsweise die folgende Tabelle in der 4D Struktur.



Company	
ID	2 ³²
name	A
creationDate	17
revenues	0.5
extra	{ }

Die Tabelle [Company] ist automatisch als eine Dataclass im Datastore **ds** verfügbar. Sie können schreiben:

```
C_OBJECT(compClass)
compClass:=ds.Company
```

Dieser Code weist *compClass* eine Referenz auf die Dataclass Company zu.

Ein Objekt Dataclass enthält:

- Attribute
- Verknüpfungsattribute

Die Dataclass bietet eine Abstraktion der physikalischen Datenbank und erlaubt den Umgang mit einem konzeptionellen Datenmodell. Die Dataclass ist die einzige Möglichkeit, den Datastore abzufragen. Eine Abfrage erfolgt aus einer einzigen Dataclass. Sie basiert auf Attributen und Verknüpfungsattributen der Dataclasses. Die Verknüpfungsattribute sind also das Mittel, um mehrere verknüpfte Tabellen in eine Abfrage einzubinden.

Attribute der Dataclass

Attribute der Dataclass stehen als Eigenschaften ihrer jeweiligen Klassen zur Verfügung. Zum Beispiel:

```
nameAttribute:=ds.Company.name //Referenz auf Class-Attribut
revenuesAttribute:=ds.Company["revenues"] //alternativer Weg
```

Dieser Code ordnet *nameAttribute* und *revenuesAttribute* Referenzen auf die Attribute *name* und *revenues* der Klasse Company zu. Diese Syntax gibt KEINE Werte innerhalb des Attributs zurück, sondern Referenzen auf die Attribute selbst. Für Werte müssen Sie **Entities** durchlaufen.

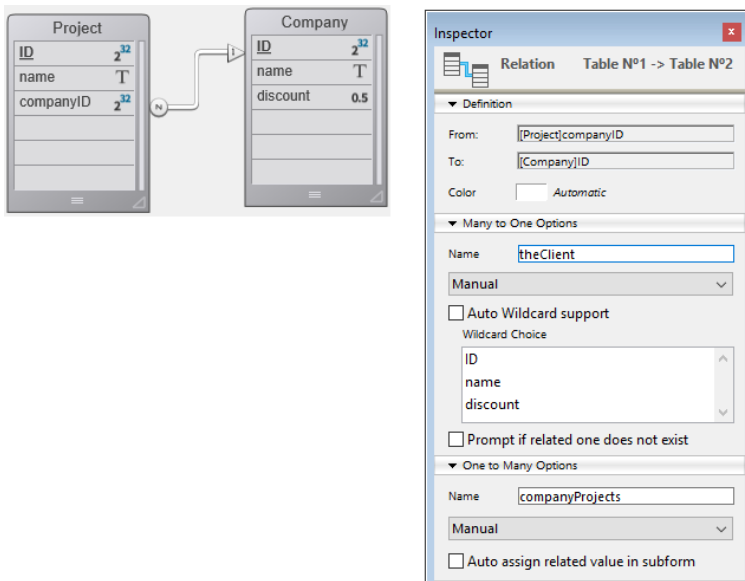
Hinweis: Dataclass-Attribute sind Objekte, die Eigenschaften haben. Weitere Informationen dazu finden Sie im Abschnitt **ORDA - DataClassAttribute**.

Speicherattribute und Verknüpfungsattribute

Es gibt verschiedene Arten von Dataclass-Attributen: Storage, relatedEntity und relatedEntities. Attribute, die skalar sind (d.h. nur einen einzigen Wert liefern), unterstützen den standardmäßigen 4D Datentyp (Lange Ganzzahl, Text, Objekt, etc.).

- Ein **Speicherattribut** entspricht einem Feld in der 4D Datenbank und kann indiziert werden. Werte, die einem Speicherattribut zugeordnet sind, werden beim Sichern als Teil der Entity gespeichert. Beim Zugriff auf ein Speicherattribut kommt sein Wert direkt aus dem Datastore. Speicherattribute sind der grundlegendste Baustein einer Entity und werden durch Namen und Datentyp definiert.
- Ein **Verknüpfungsattribut** ermöglicht den Zugriff auf andere Entities. Verknüpfungsattribute können entweder eine einzelne Entity (oder keine Entity) oder eine Entity-Selection (0 bis N Entities) ergeben. Sie basieren auf "klassischen" Verknüpfungen in der relationalen Struktur, um einen direkten Zugriff auf verknüpfte Entities zu ermöglichen. Verknüpfungsattribute sind in ORDA direkt über ihren Namen verfügbar.

Betrachten Sie beispielsweise die folgende partielle Datenbankstruktur und die Verknüpfungseigenschaften:



Alle Speicherattribute sind automatisch verfügbar:

- in der Dataclass Project: "ID", "name" und "companyID"
- in der Dataclass Company: "ID", "name" und "discount"

Zusätzlich sind automatisch die folgenden Verknüpfungsattribute verfügbar:

- in der Dataclass Project: Attribut **theClient** vom Typ "relatedEntity"; für jedes Projekt (Client) gibt es mindestens eine Company
- in der Dataclass Company: Attribut **companyProjects** vom Typ "relatedEntities"; für jede Company gibt es eine beliebige Anzahl verknüpfter Projekte

Hinweis: Die Eigenschaft "Manuell" oder "Automatisch" von Verknüpfungen hat keine Auswirkung in ORDA.

Alle Attribute der Dataclass werden als Eigenschaften der Dataclass angezeigt:

Expression	Value
ds.Company	DataClass: Company
ds.Company.companyProjects	{"name":"companyProjects","kind":"relatedEntities","relatedDataClass":"Project","type":"ProjectSelection"}
ds.Company.discount	{"name":"discount","kind":"storage","type":"number","fieldNumber":3}
ds.Company.ID	{"name":"ID","kind":"storage","type":"long","fieldNumber":1}
ds.Company.name	{"name":"name","kind":"storage","type":"string","fieldNumber":2}
ds.Project	DataClass: Project
ds.Project.companyID	{"name":"companyID","kind":"storage","type":"long","fieldNumber":3}
ds.Project.ID	{"name":"ID","kind":"storage","type":"long","fieldNumber":1}
ds.Project.name	{"name":"name","kind":"storage","type":"string","fieldNumber":2}
ds.Project.theClient	{"name":"theClient","kind":"relatedEntity","relatedDataClass":"Company","type":"Company"}

Beachten Sie, dass diese Objekte Attribute beschreiben, aber keinen Zugriff auf Daten gewähren. Das Lesen oder Schreiben von Daten erfolgt über Entity-Objekte. Weitere Informationen dazu finden Sie im Abschnitt **Entity Attribute verwenden**.

Attributnamen und Attributreferenzen

Einige ORDA-Methoden akzeptieren String-Referenzen als Attributnamen und können auch Attribut-Referenzen akzeptieren. Betrachten Sie zum Beispiel folgendes:

```
localPeople:=ds.Employee.query("zipCode = 95113")
lastName:=localPeople.toCollection("lastname")
```

Dieser Code definiert eine Entity-Selection von Personen im Postleitzahlbereich 95113. Er erzeugt dann eine Collection von Nachnamen. Anstelle eines String-Wertes in Form eines Attributs können Sie auch eine Attribut-Referenz verwenden:

```
lastNameAtt:=ds.Employee.lastname
localPeople:=ds.Employee.query("zipCode = 95113")
lastName:=localPeople.toCollection(lastNameAtt)
```

Über das Objekt Dataclass

Das Objekt Dataclass selbst kann nicht als Objekt kopiert werden:

```
$mydataclass:=OB Copy(ds.Employee) //gibt Null zurück
```

Die Eigenschaften der Dataclass sind dagegen aufzählbar:

```
ARRAY TEXT($prop;0)
OB GET PROPERTY NAMES(ds.Employee;$prop)
```

//\$prop enthält die Namen aller Dataclass-Attribute

Entities

Eine Entity ist ein **Objekt**, das als Instanz einer Dataclass angesehen werden kann (vergleichbar mit einem Datensatz der Tabelle), die mit der Dataclass in ihrem zugehörigen Datastore übereinstimmt. Eine Entity enthält jedoch auch Daten, die mit der Datenbank des dazugehörigen Datastore zusammenhängen. Die Entity dient zum Verwalten der Daten (Anlegen, Aktualisieren, Löschen). Wird eine Entity-Referenz über eine Entity-Selection erhalten, bewahrt sie Informationen über die Entity-Selection, die ein Blättern durch die Selektion ermöglicht.

Eine Entity erstellen

Es gibt zwei Möglichkeiten, eine neue Entity in einer Dataclass anzulegen:

- Da Entities Referenzen auf Datensätze der Datenbank sind, können Sie Entities erstellen, indem Sie Datensätze mit der klassischen 4D Sprache anlegen und dann mit ORDA Methoden wie `entity.next()` oder `entitySelection.first()` darauf verweisen.
- Sie können eine Entity auch mit der Methode `dataClass.new()` erstellen.

Beachten Sie, dass die Entity nur im Speicher angelegt wird. Wenn Sie diese im Datastore hinzufügen wollen, müssen Sie die Methode `entity.save()` aufrufen.

Entity-Attribute stehen direkt als Eigenschaften des Objekts Entity zur Verfügung. Weitere Informationen dazu finden Sie unter **Entity Attribute verwenden**.

Beispielsweise wollen wir eine neue Entity in der Dataclass "Employee" im aktuellen Datastore anlegen, wobei "John" und "Dupont" den Attributen `firstname` und `name` zugeordnet sind:

```
C_OBJECT($myEntity)
$myEntity:=ds.Employee.new() //Ein neues Objekt vom Typ Entity erstellen
$myEntity.name:="Dupont" // 'Dupont' dem Attribut 'name' zuordnen
$myEntity.firstname:="John" //'John' dem Attribut 'firstname' zuordnen
$myEntity.save() //Die Entity sichern
```

Hinweis: Eine Entity ist nur in dem Prozess definiert, wo sie erstellt wurde. So können Sie beispielsweise nicht eine Referenz auf eine Entity in einer Interprozessvariablen speichern und in einem anderen Prozess verwenden.

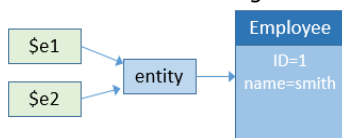
Entities und Referenzen

Eine Entity enthält eine Referenz auf einen 4D Datensatz. Verschiedene Entities können auf denselben 4D Datensatz verweisen. Da eine Entity in einer 4D Objektvariablen gespeichert werden kann, können verschiedene Variablen eine Referenz auf dieselbe Entity enthalten.

Wenn Sie den folgenden Code ausführen:

```
C_OBJECT($e1;$e2)
$e1:=ds.Employee.get(1) //Zugriff auf den employee mit ID 1
$e2:=$e1
$e1.name:="Hammer"
//beide Variablen $e1 and $e2 nutzen die Referenz auf dieselbe Entity
//$e2.name enthält "Hammer"
```

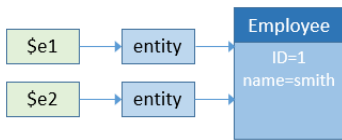
Dies wird durch die folgende Grafik veranschaulicht:



Führen Sie jetzt aus:

```
C_OBJECT($e1;$e2)
$e1:=ds.Employee.get(1)
$e2:=ds.Employee.get(1)
$e1.name:="Hammer"
//Variable $e1 enthält eine Referenz auf eine Entity
//Variable $e2 enthält eine andere Referenz auf eine andere Entity
//$e2.name enthält "smith"
```

Dies wird durch die folgende Grafik veranschaulicht:



Beachten Sie jedoch, dass sich Entities auf denselben Datensatz beziehen. In jedem Fall, wenn Sie die Methode **entity.save()** aufrufen, wird der Datensatz aktualisiert (außer im Konfliktfall, siehe **Entity sperren**).

Tatsächlich ist \$e1 und \$e2 nicht die Entity selbst, sondern eine Referenz auf die Entity. Das heißt, Sie können sie direkt an eine beliebige Funktion oder Methode übergeben und sie wirkt wie ein Zeiger, ist aber schneller als ein 4D Zeiger. Zum Beispiel:

```

For each($entity;$selection)
  do_Capitalize($entity)
End for each
  
```

Und die Funktion ist:

```

$entity:=$1
$name:=$entity.lastname
If(Not($name=NULL))
  $name:=Uppercase(Substring($name;1;1))+Lowercase(Substring($name;2;))
End if
$entity.lastname:=$name
  
```

Sie können eine Entity wie alle anderen Objekte in 4D behandeln und ihre Referenzen direkt als Parameter übergeben.

Hinweis: Mit den Entities gibt es kein Konzept "aktueller Datensatz" wie in der klassischen 4D Sprache. Sie können beliebig viele Entities gleichzeitig verwenden. Es gibt auch keine automatische Sperre auf eine Entity (siehe **Entity sperren**). Wenn eine Entity geladen wird, verwendet sie den 'Lazy loading' Mechanismus, was bedeutet, dass nur die benötigten Informationen geladen werden. Im Client/Server Betrieb kann die Entity jedoch bei Bedarf automatisch direkt geladen werden.

Entity Attribute verwenden

Entity-Attribute speichern Daten und bilden entsprechende Felder in der entsprechenden Tabelle ab. Attribute vom Typ **Speicher** können als einfache Eigenschaften des Entity-Objekts gesetzt oder abgerufen werden, während Entity vom Typ **relatedEntity** bzw. **relatedEntities** eine Entity oder eine Entity-Selection zurückgibt.

Hinweis: Weitere Informationen dazu finden Sie im Abschnitt **Speicherattribute und Verknüpfungsattribute**.

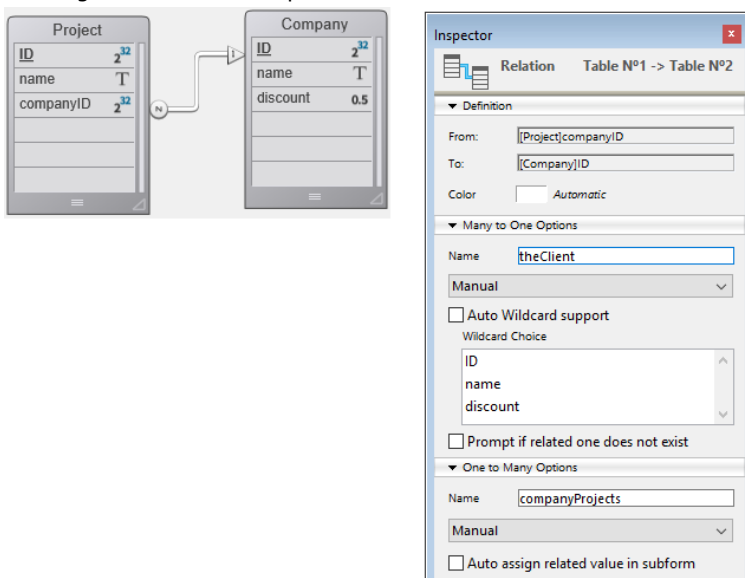
Hier ein Beispiel zum Setzen eines Speicher-Attributs:

```

$entity:=ds.Employee.get(1) //erhält ein employee Attribut mit ID 1
$name:=entity.lastname //erhält den employee Namen, z.B. "Smith"
entity.lastname:="Jones" //setzt den employee Namen
  
```

Hinweis: Bild-Attribute lassen sich nicht direkt einem gegebenen Pfad in einer Entity zuordnen.

Der Zugriff auf ein verknüpftes Attribut richtet sich nach der Art des Attributs. Nehmen wir z.B. folgende Struktur:

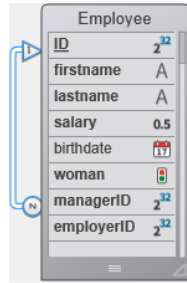
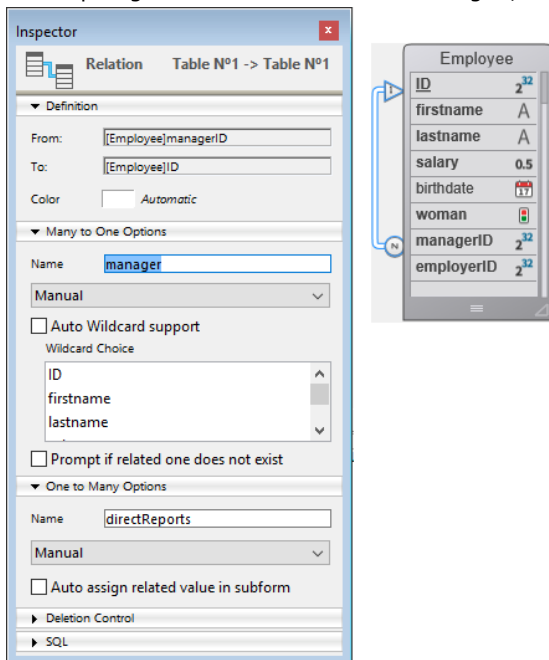


Sie können auf Daten über das/die verknüpften Objekte zugreifen:

```

$entity:=ds.Project.all().first().theClient //Die dem Projekt zugewiesene Entity Company erhalten
$EntitySel:=ds.Company.all().first().companyProjects //Die Auswahl der Projekte für die Company erhalten
  
```

Beachten Sie, dass *theClient* und *companyProjects* im obigen Beispiel primäre Verknüpfungsattribute sind und eine direkte Verknüpfung zwischen zwei Dataclasses darstellen. Dagegen lassen sich Verknüpfungsattribute auch auf Pfaden zu Verknüpfungen auf mehreren Ebenen anlegen, inkl. zirkularer Verknüpfungen. Nehmen wir z.B. folgende Struktur:



Jeder *employee* kann ein Manager sein und einen Manager haben. Um den Manager des Managers eines *employee* zu erhalten, schreiben Sie einfach:

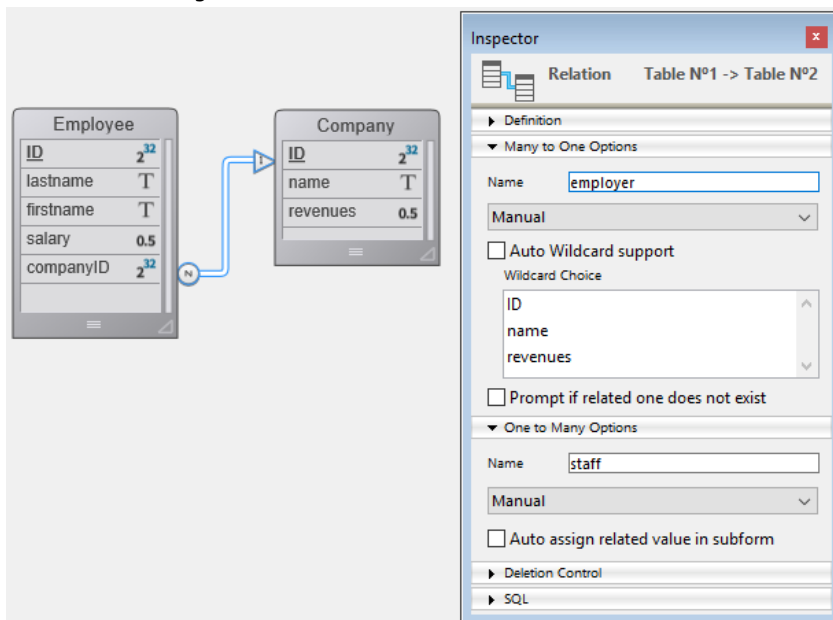
```
$myEmp:=ds.Employee.get(50)
$manLev2:=$myEmp.manager.manager.lastname
```

Den Verknüpfungsattributen Werte zuweisen

In der ORDA Architektur können Verknüpfungsattribute direkt Daten enthalten, die mit Entities verknüpft sind:

- Ein Verknüpfungsattribut vom Typ N->1 (Typ **relatedEntity**) enthält eine Entity
- Ein Verknüpfungsattribut vom Typ 1->N (Typ **relatedEntities**) enthält eine Entity-Selection

Sehen wir uns folgende vereinfachte Struktur an:



In diesem Beispiel enthält eine Entity in der Dataclass "Employee" ein Objekt vom Typ Entity im Attribut "employer" (oder einen Wert Null). Eine Entity in der Dataclass "Company" enthält ein Objekt vom Typ Entity-Selection im Attribut "staff" (oder einen Wert Null).

Hinweis: Die Eigenschaft "Manuell" oder "Automatisch" von Verknüpfungen hat keine Auswirkung in ORDA.

Um einen Wert direkt dem Attribut "employer" zuzuweisen, müssen Sie eine vorhandene Entity von der Dataclass "Company" übergeben. Zum Beispiel:

```
$emp:=ds.Employee.new() // einen employee anlegen
$emp.lastname="Smith" //einem Attribut einen Wert zuweisen
```

```
$emp.employer:=ds.Company.query("name =:1";"4D")[0] /eine Entity company zuweisen
$emp.save()
```

4D bietet eine zusätzliche Möglichkeit, um ein Verknüpfungsattribut für eine Entity N einzugeben, die mit einer Entity "1" verknüpft ist: Sie übergeben den Primärschlüssel der Entity "1" direkt, wenn dem Verknüpfungsattribut ein Wert zugewiesen wird. Damit das funktioniert, übergeben Sie dem Verknüpfungsattribut Daten vom Typ Zahl oder Text (Wert des Primärschlüssels). 4D übernimmt dann automatisch die Suche nach der entsprechenden Entity in der Dataclass. Zum Beispiel:

```
$emp:=ds.Employee.new()
$emp.lastname:="Wesson"
$emp.employer:=2 //dem Verknüpfungsattribut einen Primärschlüssel zuweisen
//4D sucht nach der Company mit dem Primärschlüssel 2 (in diesem Fall nach seiner ID)
//und weist ihn dem employee zu
$emp.save()
```

Das ist besonders hilfreich, wenn Sie große Datenmengen aus einer relationalen Datenbank importieren. Diese Art von Import enthält normalerweise eine Spalte "ID", die auf einen Primärschlüssel verweist, den Sie dann direkt einem Verknüpfungsattribut zuweisen können.

Das bedeutet auch, dass Sie Primärschlüssel in den N Entities zuweisen können, ohne dass entsprechende Entities bereits in der Datastore Class 1 angelegt wurden. Weisen Sie einen Primärschlüssel zu, der nicht in der verknüpften Datastore Class existiert, wird er trotzdem gespeichert und von 4D zugewiesen, sobald diese Entity "1" erstellt wird.

Sie können den Wert eines verknüpften Entity-Attributs "1" aus der Dataclass "N" direkt über das verknüpfte Attribut zuweisen oder ändern. Wollen Sie z.B. das Attribut *name* der verknüpften Entity *Company* einer Entity *Employee* ändern, schreiben Sie:

```
$emp:=ds.Employee.get(2) //die Entity Employee mit dem Primärschlüssel 2 laden
$emp.employer.name:="4D, Inc." //das Attribut name der verknüpften Company ändern
$emp.employer.save() //das verknüpfte Attribut sichern
//die verknüpfte Entity ist aktualisiert
```

Über das Objekt Entity

Das Objekt Entity selbst kann nicht als Objekt kopiert werden:

```
$myentity:=OB Copy(ds.Employee.get(1)) //gibt Null zurück
```

Die Eigenschaften der Entity sind dagegen aufzählbar:

```
ARRAY TEXT($prop;0)
OB GET PROPERTY NAMES(ds.Employee.get(1);$prop)
//$prop enthält die Namen aller Entity-Attribute
```

Für Exportzwecke können Sie eine Entity über die Methode **entity.toObject()** in ein reguläres Objekt konvertieren. Dann können Sie die Entity als JSON exportieren, zum Beispiel:

```
C_OBJECT($entity_json)
$entity_json:=JSON Stringify($myentity.toObject()) //gibt die Entity als JSON zurück
```

Überblick

Eine Entity-Selection ist ein Objekt, das eine oder mehrere Referenzen auf Entities der gleichen Dataclass enthält. Eine Entity-Selection kann 0, 1 oder X Entities aus der Dataclass enthalten, wobei X die Gesamtzahl der in der Dataclass enthaltenen Entities darstellen kann. Die Auswahl der Entities kann "sortiert" oder "unsortiert" erfolgen (dieser Punkt wird weiter unten erläutert).

Entity-Selections werden in der Regel über eine Abfrage (query) erstellt oder aus einem Verknüpfungsattribut zurückgegeben. Zum Beispiel:

```
brokers:=ds.Person.query("personType = broker")
```

Dieser Code gibt in *brokers* alle Personen vom Typ Broker zurück. Um auf eine Entity der Selection zuzugreifen, verwenden Sie eine ähnliche Syntax wie beim Zugriff auf ein Element in einer Collection. Zum Beispiel:

```
theBroker:=brokers[0] //Entity-Selections gehen von 0 aus
```

Die Methode **entitySelection.orderBy()** gibt eine neue Entity-Selection gemäß den definierten Sortierkriterien zurück. Zum Beispiel:

```
brokers:=brokers.orderBy("name") //gibt eine sortierte Selection zurück
```

Dieser Code gibt in *brokers* die gleiche Entity-Selection der Entities *person* zurück, jedoch sortiert nach Namen. Alternativ können Sie eine Entity-Selection auch über ein Verknüpfungsattribut zurückgeben. Zum Beispiel:

```
brokers:=ds.Person.query("personType = broker")
brokerCompanies:=brokers.myCompany
```

Dieser Code ordnet den *brokerCompanies* alle verbundenen Unternehmen der Personen in der Entity-Selection *brokers* zu, die das Verknüpfungsattribut *myCompany* annehmen. Die Verwendung von Verknüpfungsattributen sind ein leistungsstarker und einfacher Weg, um in den verknüpften Entities nach oben und unten zu navigieren.

Um wiederholte Aktionen an Entities in einer Entity-Selection durchzuführen, wie Werte bestimmter Attribute abrufen und ändern, können Sie die Struktur **For each...End for each** verwenden. Zum Beispiel:

```
C_OBJECT(emp)
For each(emp;ds.Employees.all())
  If(emp.Country="UK")
    emp.salary:=emp.salary*1,03
    emp.save()
  End if
End for each
```

Eine Entity-Selection erstellen

Sie können ein Objekt vom Typ Entity-Selection wie folgt anlegen:

- Die Entities in einer Dataclass abfragen (siehe Methode **dataClass.query()**)
- Mit der Methode **dataClass.all()** alle Entities in einer Dataclass auswählen
- Mit dem Befehl **Create entity selection** oder der Methode **dataClass.newSelection()** ein leeres Objekt Entity-Collection erstellen
- Eine der verschiedenen Methoden aus dem Thema **ORDA - EntitySelection** verwenden, die eine neue Entity-Selection zurückgibt, z.B. **entitySelection.or()**
- Ein Verknüpfungsattribut vom Typ "related entities" verwenden (siehe unten)

Sie können gleichzeitig beliebig viele verschiedene Entity-Selections für eine Dataclass anlegen und verwenden.

Beachten Sie, dass eine Entity-Selection nur Referenzen auf Entities enthält. Verschiedene Entity-Selections können Referenzen auf die gleichen Entities enthalten.

Hinweis: Eine Entity-Selection ist nur in dem Prozess definiert, wo sie erstellt wurde. So können Sie beispielsweise nicht eine Referenz auf eine Entity-Selection in einer Interprozessvariablen speichern und in einem anderen Prozess verwenden.

Entity-Selections und Attribute

Entity-Selections und Speicherattribute

Alle Speicherattribute (Text, Nummer, Boolean, Datum) sind sowohl als Eigenschaften von Entity-Selections als auch von Entities verfügbar. In Verbindung mit einer Entity-Selection gibt ein skalares Attribut eine Collection von skalaren Werten

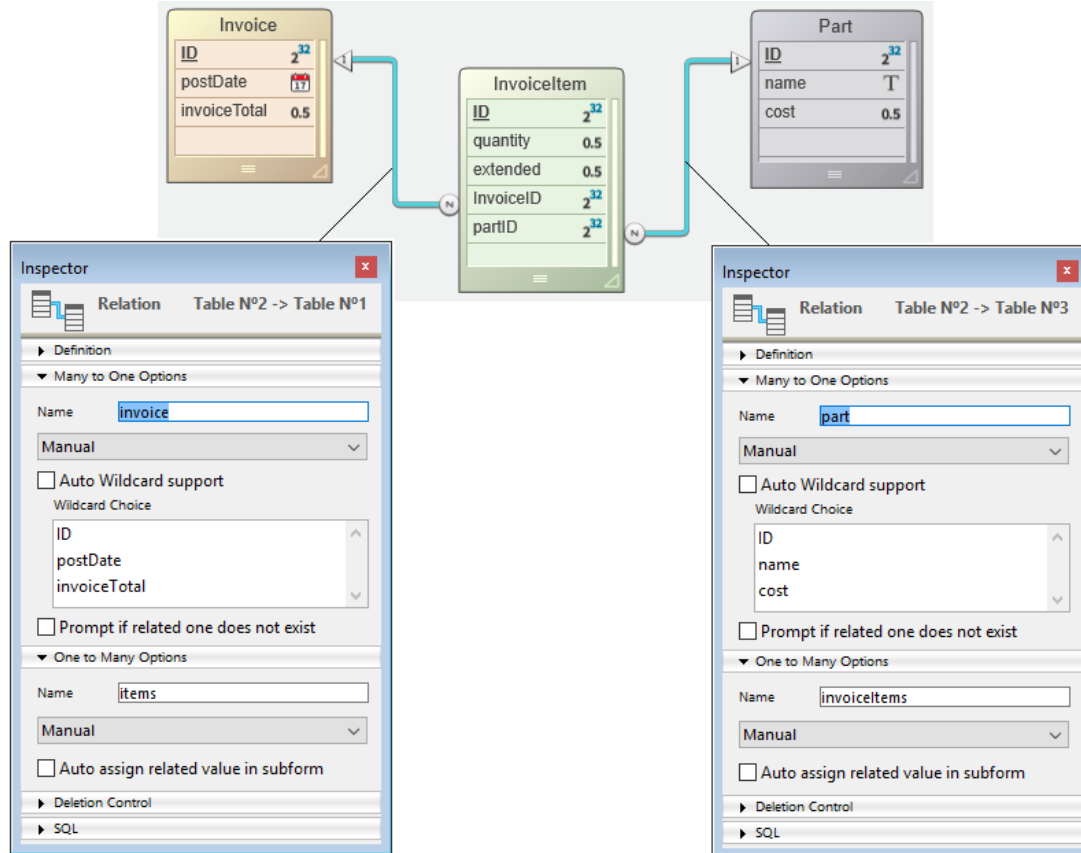
zurück. Zum Beispiel:

```
locals:=ds.Person.query("city = :1";"San jose") //Entity-Selection von Personen
localEmails:=locals.emailAddress //Collection von E-Mail Adressen (Strings)
```

Dieser Code gibt in *localEmails* eine Collection von E-Mail Adressen als Strings zurück.

Entity-Selections und Verknüpfungsattribute

Zusätzlich zu den verschiedenen Wegen für Abfragen (query) können Sie Verknüpfungsattribute als Eigenschaften von Entity-Selections verwenden, um neue Entity-Selections zurückzugeben. Nehmen wir z.B. folgende Struktur:



```
myParts:=ds.Part.query("ID < 100") //gibt Teile (parts) mit ID unter 100 zurück
$myInvoices:=myParts.invoiceItems
//Alle Rechnungen mit mindestens einem Rechnungseintrag, verknüpft mit einem Teil in myParts
```

Die letzte Zeile gibt in *\$myInvoices* eine Entity-Selection aller Rechnungen zurück, die mindestens einen Rechnungseintrag haben, verknüpft mit einem Teil (part) in der Entity-Selection *myParts*. Wird ein Verknüpfungsattribut als Eigenschaft einer Entity-Selection verwendet, ist das Ergebnis immer eine andere Entity-Selection, selbst wenn nur eine Entity zurückgegeben wird. Werden keine Entities zurückgegeben, ist das Ergebnis eine leere Entity-Selection, nicht **Null**.

Sortierte vs unsortierte Entity-Selections

Lokale Entity-Selections können **sortiert** oder **unsortiert** sein. Die Funktionsweise ist für beide Arten ähnlich, es gibt Unterschiede in der Performance und den Features. Sie können selbst bestimmen, welcher Typ sich besser für Ihre spezifischen Anforderungen eignet.

Hinweis: Entity-Selections in einem remote Datastore sind immer **sortiert**.

Vergleich der beiden Entity-Selections

- Unsortierte Entity-Selections werden auf Bit Tabellen im Speicher aufgebaut. Eine unsortierte Entity-Selection enthält ein Bit pro Entity in der Dataclass, unabhängig ob die Entity tatsächlich in der Selection ist. Jedes Bit ist gleich 1 oder 0, um anzuzeigen, ob die Entity in der Selection enthalten ist oder nicht. Es ist eine sehr kompakte Darstellung jeder Entity. Demzufolge sind Operationen mit unsortierten Entity-Selections sehr schnell und benötigen wenig Speicherplatz. Die Größe einer unsortierten Entity-Selection in Bytes entspricht immer der Gesamtanzahl von Entities in der Dataclass, dividiert durch 8. Beispiel: Sie erstellen eine unsortierte Entity-Selection für eine Dataclass mit 10.000 Entities, das beansprucht bis zu 1.250 Bytes, was ungefähr 1.2K in RAM ausmacht. Dagegen lassen sich unsortierte Entity-Selections nicht sortieren. Sie können sich nicht auf die Position von Entities innerhalb der Selection verlassen. Es kann auch nicht mehrere Referenzen auf eine Entity in der Selection geben: Jede Entity lässt sich nur einmal hinzufügen.
- Sortierte Entity-Selections werden auf Arrays vom Typ Lange Ganzzahl im Speicher aufgebaut (mit Entity Referenzen). Jede Referenz auf eine Entity beansprucht 4 Bytes im Speicher. Bearbeiten und Verwalten dieser Selections erfordert mehr Zeit und mehr Speicherplatz als unsortierte Entity-Selections.

Sie können dagegen sortiert oder unsortiert sein und Sie können sich auf die Entity Positionen verlassen. Außerdem können Sie mehrere Referenzen auf dieselbe Entity hinzufügen.

Die Übersicht zeigt die Hauptmerkmale:

Feature	Unsortierte Entity-Selection	Sortierte Entity-Selection
Bearbeitungsgeschwindigkeit	sehr schnell	langsamer
Größe im Speicher	sehr klein	größer
Sortierbar	nein	ja
Kann mehrere Referenzen auf eine Entity enthalten	nein	ja

Aus Optimierungsgründen erstellt 4D ORDA standardmäßig unsortierte Entity-Selections, außer Sie verwenden die Methode **orderBy()** oder geeignete Optionen (siehe unten). In dieser Dokumentation bezieht sich "entity selection" in der Regel auf "unsortierte Entity Collection". Andernfalls wird das spezifisch angegeben.

Sortierte oder unsortierte Entity-Selections erstellen

Wie oben erwähnt, erstellt und behandelt ORDA standardmäßig unsortierte Entity-Selections als Ergebnis von Operationen wie Anfragen oder Vergleiche wie **and()**. Sortierte Entity-Selections werden nur bei Bedarf erstellt, oder wenn sie über spezifische Optionen angefordert werden.

Sortierte Entity-Selections werden in folgenden Fällen erstellt:

- Ergebnis einer **orderBy()** auf eine Selection ((jeder Typ) oder ein **orderBy()** auf eine Dataclass
- Ergebnis der Methode **newSelection()** mit der Option `dk keep ordered`

Unsortierte Entity-Selections werden in folgenden Fällen erstellt:

- Ergebnis einer standardmäßigen **query()** auf eine Selection (jeder Typ) oder eine **query()** auf eine Dataclass
- Ergebnis der Methode **newSelection()** ohne Option
- Ergebnis einer Vergleichsmethode, unabhängig vom Typ der Input Selection: **or()**, **and()**, **minus()**

Beachten Sie, dass sich wiederholende Entity Referenzen entfernt werden, wenn eine sortierte Entity-Selection zu einer unsortierten Entity-Selection wird.

Wollen Sie eine sortierte Entity-Selection in eine unsortierte umwandeln, müssen Sie nur eine **and()** Operation darauf anwenden, wie z.B.:

```
//mySel ist eine sortierte Entity-Selection
mySel:=mySel.and(mySel)
//mySel ist jetzt eine unsortierte Entity-Selection
```

Über das Objekt Entity-Selection

Das Objekt Entity-Selection selbst kann nicht als Objekt kopiert werden:

```
$myentysel:=OB Copy(ds.Employee.all()) //gibt Null zurück
```

Die Eigenschaften der Entity-Selection sind dagegen aufzählbar:

```
ARRAY TEXT($prop;0)
OB GET PROPERTY NAMES(ds.Employee.all();$prop)
//$prop enthält die Namen der Entity-Selection Eigenschaften
//("length", "00", "01"...)
```

Entity sperren

Oftmals müssen Sie Konflikte verwalten, die auftreten können, wenn mehrere Benutzer oder Prozesse die gleichen Entities zur selben Zeit laden und versuchen zu ändern. In relationalen Datenbanken ist Datensatz-Sperren eine Vorgehensweise, um inkonsistente Aktualisierung von Daten zu vermeiden. Das Konzept dahinter ist, entweder einen Datensatz auf Nur-Lesen Modus zu setzen, so dass kein anderer Prozess ihn ändern kann, oder beim Sichern des Datensatzes zu prüfen, dass ihn kein anderer Prozess seit dem Laden geändert hat. Die erste Vorgehensweise wird als **pessimistisches Datensatz-Sperren** bezeichnet. Sie stellt sicher, dass ein geänderter Datensatz geschrieben werden kann und für andere Benutzer gesperrt ist. Die zweite Vorgehensweise wird als **optimistisches Datensatz-Sperren** bezeichnet. Sie handhabt das Schreibrecht auf den Datensatz flexibel, und weist es nur zu, wenn der Datensatz aktualisiert werden muss. Beim pessimistischen Sperren wird der Datensatz gesperrt, auch wenn er nicht aktualisiert werden muss. Beim optimistischen Sperren wird die Gültigkeit einer Änderung im Datensatz zum Zeitpunkt der Aktualisierung bestimmt.

ORDA bietet zwei Sperrmodi für Entities:

- einen automatischen "optimistischen" Modus, der für die meisten Anwendungen passt
- einen "pessimistischen" Modus, um Entities zu sperren, bevor darauf zugegriffen wird

Automatisches optimistisches Sperren

Dieser automatische Mechanismus basiert auf dem Konzept "optimistisches Sperren", das sich besonders für Belange von Web Applikationen eignet. Dafür gelten folgende Vorgehensweisen:

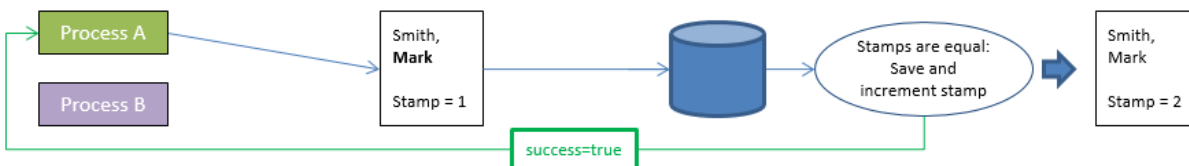
- Alle Entities lassen sich immer im Lese-/Schreibmodus laden; es gibt kein vorrangiges Sperren von Entities.
- Jede Entity hat einen internen Sperrstempel, der bei jedem Sichern erhöht wird.
- Versucht ein Benutzer oder Prozess, eine Entity mit der Methode **entity.save()** zu sichern, vergleicht 4D den Wert des Stempels der zu sichernden Entity mit dem der Entity in den Daten (im Falle einer Änderung):
 - Stimmen die Werte überein, wird die Entity gesichert und der interne Stempelwert erhöht.
 - Stimmen die Werte nicht überein, heißt das, dass zwischenzeitlich ein anderer Benutzer diese Entity geändert hat. Es wird keine Sicherung durchgeführt und ein Fehler zurückgegeben.

Nachfolgendes Diagramm illustriert optimistisches Sperren:

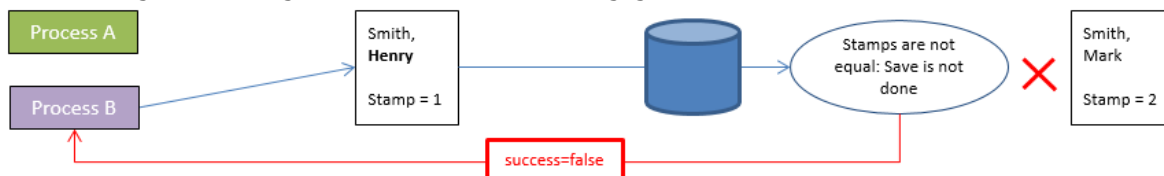
1. Zwei Prozesse laden die gleiche Entity.



2. Der erste Prozess ändert die Entity und bestätigt die Änderung. Die Methode **entity.save()** wird aufgerufen. Die 4D Engine vergleicht automatisch den internen Stempelwert der geänderten Entity mit dem der in den Daten gespeicherten Entity. Da sie zusammenpassen, wird die Entity gesichert und der Stempelwert hochgesetzt.



3. Der zweite Prozess ändert auch die geladene Entity und bestätigt die Änderungen. Die Methode **entity.save()** wird aufgerufen. Da der Stempelwert der geänderten Entity nicht mit der in den Daten gespeicherten Entity übereinstimmt, wird die Sicherung nicht durchgeführt und ein Fehler zurückgegeben.



Das lässt sich auch mit folgendem Code darstellen:

```
$person1:=ds.Person.get(1) //Referenz zur Entity
$person2:=ds.Person.get(1) //Andere Referenz zur gleichen Entity
$person1.name:="Bill"
$result:=$person1.save() // $result.success=true, Änderung gesichert
$person2.name:="William"
$result:=$person2.save() // $result.success=false, Änderung nicht gesichert
```

In diesem Beispiel weisen wir \$person1 eine Referenz auf die Entity *person* mit der Nummer 1 zu. Dann weisen wir der Variablen \$person2 eine andere Referenz auf dieselbe Entity zu. In \$person1 ändern wir den Vornamen der Person und sichern die Entity. Versuchen wir dasselbe für \$person2, prüft 4D, ob die Entity auf der Festplatte die gleiche ist wie die

zuerst zugewiesene Referenz in \$person1. Da das nicht der Fall ist, gibt 4D in der Eigenschaft *success* false zurück und sichert die zweite Änderung nicht.

Tritt diese Situation ein, können Sie z.B. die Entity über die Methode **entity.reload()** erneut von der Festplatte laden und versuchen, die Änderung auszuführen. Die Methode **entity.save()** enthält eine Option "automerge", um die Entity auch dann zu sichern, wenn durch Prozesse andere Attribute geändert wurden.

Pessimistisches Sperrverfahren (Pessimistic Lock)

Sie können Entities bei Zugreifen auf Daten direkt sperren und entsperren. Wird eine Entity von einem Prozess gesperrt, ist sie in diesem Prozess im Lese/Schreibmodus verfügbar und für alle anderen Prozesse gesperrt, d.h. sie können die Entity nur im Lesemodus laden und ihre Werte lassen sich weder bearbeiten noch sichern.

Dieses Feature basiert auf zwei Methoden der Entity Class:

- **entity.lock()**
- **entity.unlock()**

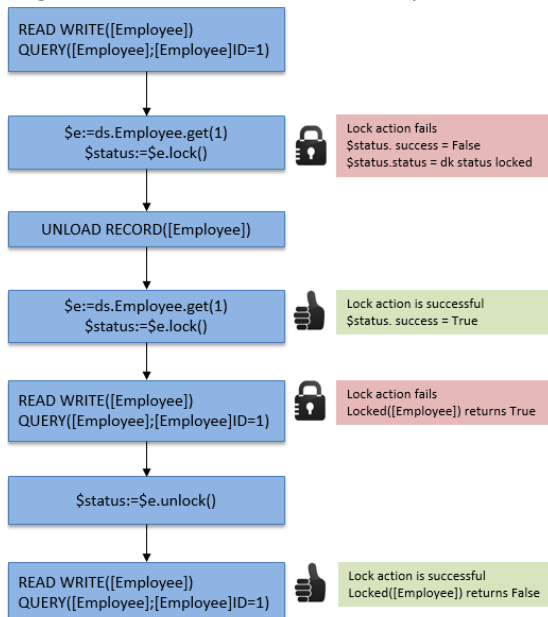
Weitere Informationen dazu finden Sie unter der Beschreibung dieser Methoden.

Gleichzeitig klassisches Sperren von 4D und pessimistisches Sperren von ORDA einsetzen

Verwenden Sie klassische und ORDA Befehle gleichzeitig zum Sperren von Datensätzen, gelten folgende Prinzipien:

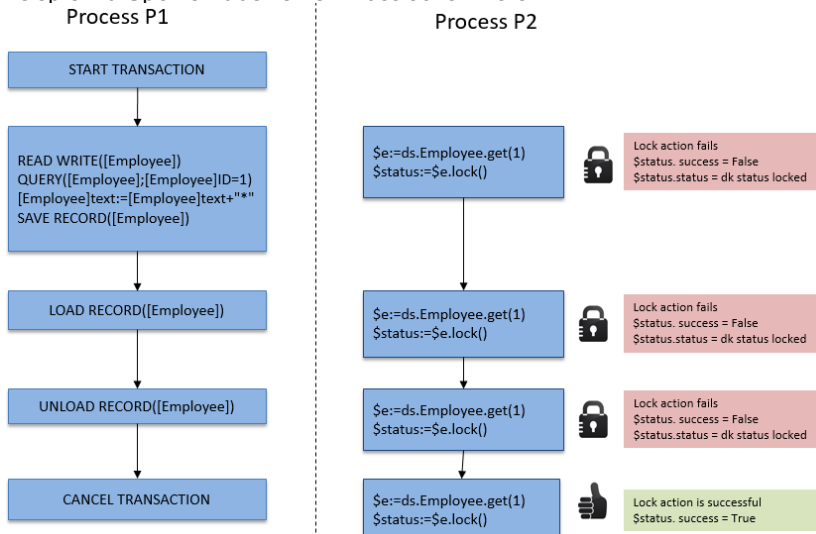
- **Datensatz-Sperren** mit klassischem 4D Befehl führt dazu, dass ORDA die zum Datensatz passende Entity nicht sperrt.
- **Entity-Sperren** mit ORDA führt dazu, dass klassische 4D Befehle den zur Entity passenden Datensatz nicht sperren.

Folgende Grafik erläutert diese Prinzipien:

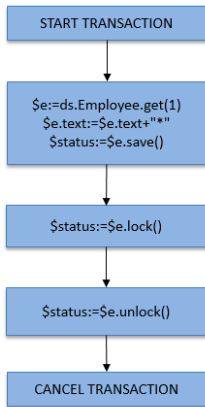


Auch **Transaktion-Sperren** wird für klassische und ORDA Befehle angewandt. In einer Anwendung mit mehreren Prozessen oder mehreren Benutzern führt Datensatz-Sperren innerhalb einer Transaktion über einen klassischen Befehl dazu, dass andere Prozesse Entities, die mit diesem Datensatz verbunden sind (oder umgekehrt) nicht sperren, bis die Transaktion bestätigt oder abgebrochen wird.

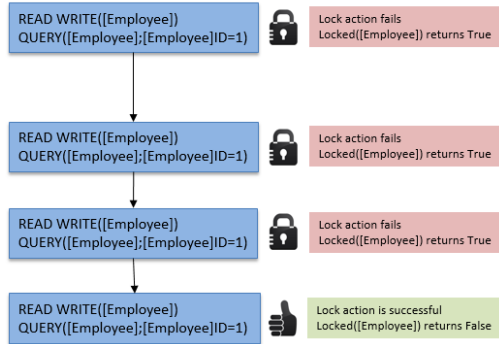
- Beispiel zu Sperren über einen klassischen Befehl:












- Beispiel zu Sperren über eine ORDA Methode:
Process P1



Process P2



Formulare verwalten

-  Überblick
-  Formular über Formularassistent erstellen
-  Leeres Formular erstellen
-  Formular bearbeiten
-  Formular umbenennen
-  Eingabe- und Ausgabeformulare festlegen
-  Formular löschen
-  Formular drucken
-  Formulare speichern

Formulare bilden die Oberfläche, über die Information eingegeben, geändert und gedruckt wird. Ein Anwender arbeitet mit den Daten in einer Anwendung über Formulare und druckt Berichte über Formulare. In eigenen Anwendungen können Sie über Formulare eigene Dialogfenster und Paletten anlegen

Jede Tabelle in Ihrer Datenbank hat im allgemeinen mindestens zwei Formulare: einmal zum Auflisten der Datensätze auf dem Bildschirm, einmal zum Anzeigen eines Datensatzes, um Daten einzugeben bzw. zu bearbeiten. Das Formular, das Datensätze auflistet, heißt Ausgabeformular oder Listenformular, das Formular, das jeweils einen Datensatz anzeigt, heißt Eingabeformular oder Detailformular. Wenn Sie Datensätze mit dem Listenformular ansehen, können Sie auf einen Datensatz doppelklicken, um ihn im aktuellen Detailformular anzusehen.

Inhalt der Formulare

In eigenen Anwendungen können Sie über die Programmiersprache steuern, welche Formulare verwendet werden sollen. Zum Beispiel könnten Sie einen Satz Formulare davon abhängig machen, ob der Vertrieb oder die Buchhaltung darauf zugreift. Sie können die Programmiersprache auch verwenden, um verschiedene Sätze von Formularen für Web Browser und 4D remote Anwender einzusetzen. Wenn Sie eine eigene Anwendung entwickeln, können Sie Formulare für eigene Dialogfenster oder Palettenfenster erzeugen. In eigenen Anwendungen können Sie über mehrfache Prozesse ermöglichen, dass Benutzer mit mehreren Formularen gleichzeitig arbeiten können.

Ein Formular kann Datenfelder aus mehr als einer Tabelle anzeigen. Sie können Datenfelder aus einer verknüpften Eine-Tabelle in ein Formular einfügen und den Anwendern ermöglichen, direkt Werte in die verknüpfte Eine-Tabelle einzugeben. Sie können auch ein Unterformular einfügen, das eine Liste der Datensätze aus einer verknüpften Viele-Tabelle anzeigt. Ein Unterformular zeigt eine Liste von Datensätzen aus einer anderen Tabelle oder einer Untertabelle in der Haupttabelle. Mit einem Unterformular kann der Anwender Datensätze in einer anderen Tabelle ansehen, eingeben und ändern. Eine Anwendung Rechnungen kann z.B. im Rechnungseingabeformular ein Unterformular enthalten, in das der Anwender Rechnungspositionen eingeben kann. Sie erscheinen zwar im Rechnungsformular, werden aber in der verknüpften Viele-Tabelle gespeichert.

Ein Formular, das für die Dateneingabe verwendet wird, kann auch mehrere Unterformular enthalten. Zum Beispiel kann eine Anwendung *Kontakte* ein Unterformular für Telefonnummern, ein weiteres für Aufgaben und ein drittes für frühere Kontakte mit der Person haben. Jedes Unterformular zeigt Datensätze aus einer anderen verknüpften Viele-Tabelle.

Ein einzelnes Formular kann einige der Datenfelder in einer Tabelle oder alle Datenfelder verwenden. Sie können zum Beispiel zwei unterschiedliche Eingabeformulare haben – eins zur Eingabe für Angestellte, ein anderes zur Eingabe für Vorgesetzte – keines der beiden enthält alle Datenfelder. Sie können bestimmte Datenfelder für die Bildschirmanzeige und wieder andere für einen gedruckten Bericht verwenden

Formulare lassen sich jederzeit ändern, unabhängig davon, ob Sie Daten in die Datenbank eingegeben haben oder nicht. Änderungen an einem Formular beeinflussen in keiner Weise die Daten, die auf der Festplatte gespeichert sind.

Jedes Formular verfügt über eine oder mehrere Seiten, in denen Datenfelder oder andere Eingabeobjekte erscheinen. Wenn Ihre Datenfelder nicht auf eine Seite passen, können Sie weitere Seiten erzeugen. Wenn Sie ein mehrseitiges Formular erzeugen, können Sie auch Schaltflächen oder eine Registerkarte hinzufügen, damit Anwender von einer Seite zur nächsten Seite wechseln können.

Jedes Formular hat eine Hintergrundseite (eine Seite Null), auf der Sie Objekte einfügen, die auf allen Seiten erscheinen. Sie verwenden die Hintergrundseite zum Einfügen von Hintergrundgrafiken, Schaltflächen, Registerkarten und anderen Grafikobjekten, die das Aussehen der Seite bestimmen, wie Rahmen und Etiketten.

Hinweis: Wenn ein mehrseitiges Formular als Ausgabeformular verwendet wird (z.B. für den Druck), erscheint nur die erste Seite.

Formulare erstellen

Mit 4D können Sie schnell Standardformulare erzeugen. Die leistungsfähigen Werkzeugen ermöglichen, Formulare mit ausgefeilter Bedienerführung zu erstellen. Ihre Formulare können genau das zur Verfügung stellen, was Sie für Ihre Anwendung brauchen. Allein durch Point- und Klick-Vorgänge können Sie ein Basisformular mit Feldern, Schaltflächen, Variablen, etc. erstellen

4D bietet zwei Werkzeuge zum Erzeugen und Ändern von Formularen: den **Formularassistenten** und den **Formulareditor**.

Formularassistent

Der Formularassistent ist Ihre Ausgangsbasis zum Erzeugen aller Formulartypen. Mit dem Formularassistenten können Sie ein neues Formular durch Auswählen der gewünschten Datenfelder aus einer Liste und der gewünschten Formularvorlage aus einer Dropdown-Liste erzeugen. Formularvorlagen bestimmen das Aussehen der Formulare. Eine Vorlage bestimmt Eigenschaften wie Formulargröße, Plattformoberfläche, Schrifteigenschaften und Schaltflächen. Weitere Informationen dazu finden Sie im Abschnitt [Formular über Formularassistent erstellen](#).

Formulareditor

Der Formulareditor ist eine objektorientierte Zeichenumgebung, in der Sie Formulare durch direktes Bearbeiten der Objekte im Formular anpassen können. Zum Beispiel können Sie Objekte verschieben oder neue hinzufügen, die der Formularassistent nicht unterstützt, mehrseitige Formulare mit Registerkarten erzeugen, Arbeitsregeln durch Beschränkungen bei der Dateneingabe vorgeben, Zugriffsrechte auf Formulare bestimmen, eine eigene Menüleiste mit einem Formular verknüpfen und Formular- und Objektmethoden schreiben, die automatisch abgearbeitet werden, wenn das Formular benutzt wird. Weitere Informationen dazu finden Sie im Abschnitt [Formular bearbeiten](#).

Tabellenformulare und Projektformulare

Mit 4D lassen sich zwei Arten von Formularen erstellen: **Tabellenformulare** und **Projektformulare**. Tabellenformulare sind einer bestimmten Tabelle zugeordnet und können so automatische Abläufe nutzen, die beim Entwickeln von datenbankgesteuerten Anwendungen hilfreich sind. Projektformulare sind unabhängige Formulare, d.h. sie sind keiner Tabelle zugeordnet. Sie dienen insbesondere zum Erstellen von Dialogfenstern für Oberflächen und Komponenten.

Mit Projektformularen können Sie Oberflächen erstellen, die besser zu den Standards von Betriebssystemen passen. Wir empfehlen insbesondere zur Anzeige von Datensätzen in Listenform, jetzt über den 4D Befehl **DIALOG** Projektformulare aufzurufen, welche Datensatzauswahlen in Unterformularen anzeigen können. Mit wenig zusätzlichem Programmieraufwand erzielen Sie so eine Verbesserung der 4D Befehle **MODIFY SELECTION** und **DISPLAY SELECTION**

Tabellenformulare und Projektformulare erscheinen im Explorer auf der **Seite Formulare** jeweils als eigene Gruppe.

Merkmale von Tabellenformularen

Jedes Formular ist mit einer Tabelle verknüpft. Jede Tabelle muss zumindest ein Formular haben, so dass sich Informationen in Datenfelder eingeben und auf dem Bildschirm anzeigen lassen. Normalerweise hat eine Tabelle getrennte Eingabe- und Ausgabeformulare. Das Eingabeformular dient zur Dateneingabe. Es zeigt jeweils einen Datensatz auf dem Bildschirm an und enthält Schaltflächen zum Sichern und Löschen von Änderungen am Datensatz und zum Navigieren zwischen den Datensätzen (z.B. Erster Datensatz, Letzter Datensatz, Voriger Datensatz, Nächster Datensatz). Das Ausgabeformular zeigt eine Liste von Datensätzen mit einem Datensatz pro Zeile. Die Ergebnisse von Suchläufen erscheinen im Ausgabeformular. Durch Doppelklick auf eine Zeile kann der Anwender das Eingabeformular für diesen Datensatz anzeigen.

Folgende Abbildung zeigt ein typisches Eingabeformular und Ausgabeformular:

Versuchen Sie, die Datensätze einer Tabelle vor dem Erzeugen eines Formulars für diese Tabelle anzuzeigen, erscheint die Meldung, ob 4D ein Standardeingabe- und Standardausgabeformular anlegen soll:

Hinweis: Sie können diese Standardeinstellung in den Datenbankeigenschaften ändern, also z.B. festlegen, dass die Standardformulare automatisch angelegt werden und der obige Dialog nicht erscheint. Weitere Informationen dazu finden Sie in den 4D Einstellungen auf der **Seite Allgemein**.

Klicken Sie auf die Schaltfläche **Ja**, um Standardformulare zu erzeugen. Sie können jederzeit in die Designumgebung zurückkehren und diese ändern oder durch besser ausgearbeitete Formulare ersetzen. Sie können diese Formulare aber auch ohne Änderungen zur Eingabe und Anzeige von Daten in Ihrer Datenbank verwenden. Klicken Sie auf **Nein**, wenn Sie der Tabelle kein Standardformular zuweisen wollen. Sie können die Dateneingabe bzw. -anzeige dann mit Projektformularen ausführen.

Ihre Datenbank kann pro Tabelle bis zu 32.000 Formulare verwenden, die spezielle Aufgaben erfüllen.

Merkmale von Projektformularen

Projektformulare unterscheiden sich von Tabellenformularen folgendermaßen:

- Projektformulare können nur vom Typ Eingabe (Seite) sein. Die Funktionalitäten für Ausgabeformulare (Liste) sind nicht kompatibel mit Projektformularen.
- Projektformulare erscheinen nicht in der Tabellenliste und lassen sich nicht als aktuelles Eingabe- bzw. Ausgabeformular definieren. Sie sind weder im **Etiketteneditor**, noch im Import/Exporteditor (siehe **Daten importieren und exportieren**) von 4D verwendbar.
- Projektformulare lassen sich nur mit dem Befehl **DIALOG** oder als vererbte Formulare anzeigen (siehe Abschnitt **Vererbte Formulare verwenden**).
- Projektformulare können dieselben Objekttypen wie Tabellenformulare enthalten, inkl. Felder. Bei Feldern speichert das Projektformular die Nummer der Tabelle und des Feldes. Wird ein Formular von einer Datenbank in eine andere oder innerhalb einer Komponente kopiert, werden die Referenzen mitkopiert. Die verwendete Tabelle und Felder sind die der Zieldatenbank. Bei Unstimmigkeiten wie nicht-vorhandene Tabelle, inkorrekt Feldtyp, etc. arbeitet das Formular nicht korrekt. Da Projektformulare hauptsächlich über den Befehl **DIALOG** verwendet werden, stehen die Schaltflächen mit den Standardaktionen wie Nächster Datensatz, Datensatz löschen weder im Editor noch im Formularassistenten zur Verfügung. Sie müssen die Anzeige und Änderung von Datensätzen über Befehle der Programmiersprache steuern. Verwenden Sie dagegen Projektmethoden als vererbte Formulare für Tabellenformulare, können Sie auch die automatischen Operationen zur Datensatzverwaltung verwenden.
- Projektformulare können genauso wie Tabellenformulare eine Formularmethode haben, die Sie im Explorer auf der **Seite Methoden** aufrufen können.

Tabellenformular in Projektformular umwandeln

Sie können jederzeit ein Tabellenformular in ein Projektformular umwandeln und umgekehrt. Beachten Sie jedoch, dass bei der Umwandlung in Projektformulare die automatischen Operationen zur Datenverwaltung dann nicht mehr funktionieren. Außerdem wird der Formulartyp "Listenformular" oder "Listenformular für Druck" in ein Projektformular vom Typ "Seite" umgewandelt.

Sie können den Formulartyp auf der **Seite Formulare** des Explorers per Drag&Drop oder Kopieren/Einfügen ändern. Das ist in derselben Anwendung und zwischen zwei verschiedenen Anwendungen möglich.

Um ein Projektformular in ein Tabellenformular umzuwandeln und umgekehrt, gehen Sie folgendermaßen vor:

1. Klicken Sie auf der Seite Formulare des Explorers in das gewünschte Formular und ziehen es auf den Zieleintrag. Beim Umwandeln eines Projektformulars in ein Tabellenformular müssen Sie das Formular auf den Namen der Tabelle ziehen, der Sie es zuordnen wollen. Das Formular wird bewegt, wenn die Drag&Drop Operation innerhalb derselben Anwendung ausgeführt wird. Wollen Sie das Formular kopieren, halten Sie während der Drag&Drop Operation unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** gedrückt. Bei Drag&Drop zwischen zwei verschiedenen Anwendungen lässt sich das Formular nur kopieren. Sie können auch die Standardeinträge **Ausschneiden/Kopieren/Einsetzen** im Kontextmenü des Explorers verwenden.

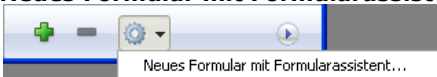
Formular über Formularassistent erstellen

Sie können neue Formulare rasch mit dem Formularassistenten erzeugen. Sie können ein neues Formular unmittelbar nach dem Erstellen verwenden oder es mit dem Formulareditor weiterbearbeiten.
Der Formularassistent hat zwei Bildschirmdarstellungen. Mit der Basisanzeige erstellen Sie neue Formulare mit ein paar einfachen Operationen. Die erweiterte Darstellung lässt Sie das Formular vor der Erstellung weiter anpassen.

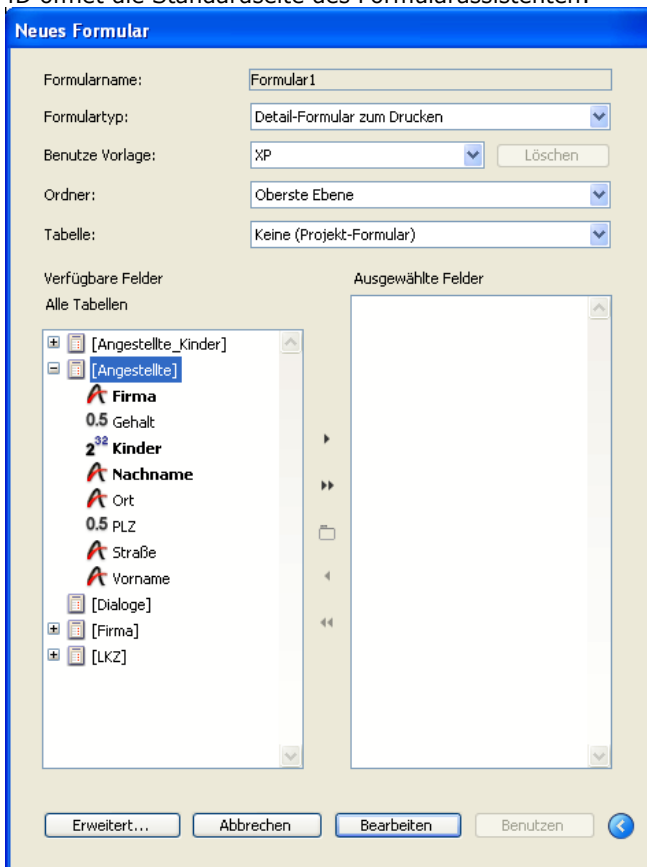
Schritte zum Anlegen eines Basisformular

Um ein neues Formular über den Formularassistenten anzulegen:

1. Wählen Sie im Menü **Datei/Ablage** den Menübefehl **Neu>Formular** oder in der Werkzeugleiste die Schaltfläche **Neu** ODER
Klicken Sie auf der **Seite Formulare** des Explorer, am unteren Rand auf das Icon Optionen und wählen den Eintrag **Neues Formular mit Formularassistent**.



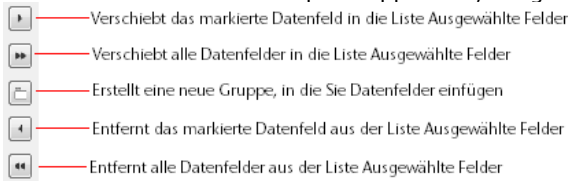
4D öffnet die Standardseite des Formularassistenten:



2. Um ein Tabellenformular anzulegen, wählen Sie im Dropdown-Menü **Tabelle** die gewünschte Tabelle.
Die Datenfelder erscheinen im Bereich *Verfügbare Felder* in alphabetischer Reihenfolge, wenn in der DropDown-Liste die Option Haupttabelle ausgewählt ist.
ODER
Um ein Projektformular zu erstellen, wählen Sie im Menü **Tabelle** den Eintrag **Keine** (Projekt-Formular).
Im Bereich *Verfügbare Felder* erscheinen dann die Datenfelder sämtlicher Tabellen. Im PopUp-Menü *Formulartyp* erscheinen dann nicht die Einträge *Listen-Formular* und *Listen-Formular zum Drucken*
3. Tragen Sie im Bereich *Formularname* einen Namen für das Formular ein.
Sie können mit der Programmiersprache über diesen Namen auf das Formular verweisen. Stellen Sie sicher, dass der Name die für 4D geltenden Namensregeln berücksichtigt. Weitere Informationen dazu finden Sie im Abschnitt **Namenskonventionen** im Handbuch *4D Programmiersprache*.
4. Wählen Sie in der Dropdown-Liste *Formulartyp* einen Formulartyp.
Sie können wählen zwischen:
 - **Detail-Formular**: Ein Formular zur Dateneingabe und -änderung
 - **Listen-Formular** (nur Tabellenformulare): Ein Formular zum Auflisten der Datensätze auf dem Bildschirm
 - **Detail-Formular zum Drucken**: Ein gedruckter Bericht mit einem Datensatz pro Seite, z.B. eine Rechnung
 - **Listen-Formular zum Drucken** (nur Tabellenformulare): Ein gedruckter Bericht, der Datensätze auflistet
5. Wählen Sie eine Vorlage für das Formular.
Die Vorlage bestimmt verschiedene Aspekte der Formulardarstellung, wie Schriftigenschaften, Position des

Datenfeldnamen, Layout der Schmuckrahmen für die Datenfelder und Plattformoberfläche. 4D bietet mehrere Vorlagen an. Sie können aber auch den Formularassistenten verwenden, um der Liste eigene Vorlagen hinzuzufügen. Weitere Informationen dazu finden Sie im Abschnitt **Formularvorlage erstellen**.

6. (Optional) Wählen Sie einen Ordner zum Speichern des Formulars.
Wählen Sie einen Ordnernamen in der DropDown-Liste, wird das Formular dort abgelegt. Über Ordner können Sie die Objekte Ihrer Anwendungen organisieren. Sie werden auf der **Seite Home** des Explorers verwaltet. Standardmäßig wird das Formular auf der obersten Ebene angelegt, d.h. in keinem Ordner
7. Wählen Sie im Bereich **Verfügbare Felder** die Datenfelder, die in Ihrem Formular erscheinen sollen.
Sie können die Datenfelder per Doppelklick, Drag&Drop oder über die Pfeiltasten im Mittelteil auswählen:



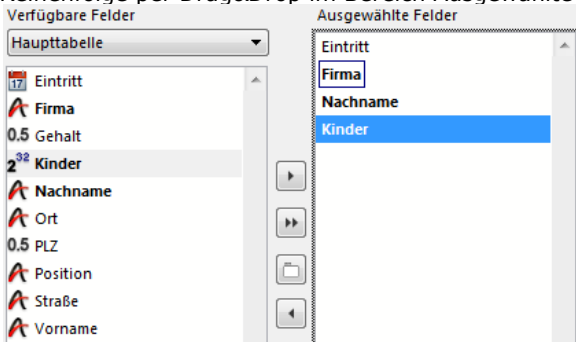
Sie können Datenfelder jeden Typs auswählen mit Ausnahme von BLOB-Datenfeldern. Sie können Formulare erzeugen, die Datenfelder enthalten von:

- Der Haupttabelle (bei Tabellenformularen)
- Einer verknüpften Eine-Tabelle
- Einer beliebigen Tabelle

Die Seite Unterformulare in den erweiterten Optionen ermöglicht Unterformulare einzurichten, die Datenfelder aus der Viele-Tabelle oder nicht verknüpfte Tabellen anzeigen.

Wählen Sie Datenfelder aus einer Tabelle, die nicht die Haupttabelle oder eine automatisch verknüpfte Eine-Tabelle, müssen Sie die Dateneingabe und -anzeige in den gewählten Datenfeldern per Programmiersprache verwalten.

Fügen Sie Datenfelder hinzu oder ändern den Typ bzw. die Vorlage des Formulars, erscheinen die Änderungen in der Vorschau im rechten Bereich des Formularassistenten. Haben Sie Datenfelder im Formular gesetzt, können Sie die Reihenfolge per Drag&Drop im Bereich **Ausgewählte Datenfelder** verändern:



Hinweis: Reicht der Bildschirm nicht aus, um in großen Formularen alle ausgewählten Datenfelder anzuzeigen, erstellt der Formularassistent ein mehrseitiges Formular und setzt alle Schaltflächen und statischen Objekte auf Seite 0. Wollen Sie diese Art von Formular bearbeiten, müssen Sie zuerst die Seite 0 anzeigen, um auf die Objekte zugreifen zu können.

8. (Optional) Bei Bedarf können Sie in der Liste *Ausgewählte Felder* eine oder mehrere Gruppenboxen erstellen.

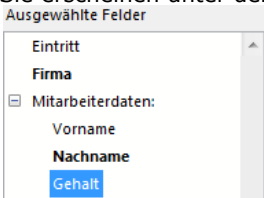
Hinweis: Diese Funktion ist für Listenformulare nicht verfügbar.

Eine Gruppenbox hat einen eigenen Namen und einen eigenen Satz Datensätze und sieht folgendermaßen aus:

Um eine Gruppenbox anzulegen, klicken Sie auf die Schaltfläche **Gruppe**. Der Eintrag mit dem Standardnamen "Gruppe" wird in der Liste *Ausgewählte Felder* gesetzt. Sie können ihn bei Bedarf umbenennen.

Hinweis: Der Gruppenname ist statischer Text. Sie können dafür, wie für alle 4D Bezeichnungen, auch eine "lokalisierbare" Referenz verwenden (siehe **Referenzen in statischem Text verwenden**).

Sie können Datenfelder einfach per Drag&Drop aus der Liste *Verfügbare Felder* in den neuen Eintrag setzen. Sie erscheinen unter dem Gruppennamen als Untereinträge.



Hinweis: Ist die Gruppe aufgeklappt, können Sie Felder, die bereits in der Liste *Ausgewählte Felder* liegen, per Drag&Drop hinzufügen.

9. Wollen Sie das neue Formular im Formulareditor bearbeiten, klicken Sie auf die Schaltfläche **Bearbeiten**.

ODER

Wollen Sie das Formular testen, klicken auf die Schaltfläche **Benutzen** (nur Tabellenformulare).

ODER

Wollen Sie das neue Formular mit den erweiterten Optionen des Formularassistenten anpassen, klicken Sie auf die Schaltfläche **Erweitert**.

Erweiterte Optionen

Um die erweiterten Optionen anzuzeigen, klicken Sie auf der Standardseite des Formularassistenten am unteren Rand auf die Schaltfläche **Erweitert**. Wollen Sie wieder zur Standardseite zurückgehen, klicken Sie auf die Schaltfläche **Zurück**. Über die erweiterten Optionen können Sie neue Formulare mit wenigen Mausklicks erstellen und eine breite Vielfalt von Optionen für eigene Formulare nutzen. Die Anpassungsoptionen sind abhängig vom Formulartyp, den Sie in der ersten Anzeige des Formularassistenten ausgewählt haben. Der Formularassistent unterstützt folgende Formulartypen:

- Detailformular
- Listenformular (nur Tabellenformulare)
- Detailformular zum Drucken
- Listenformular zum Drucken (nur Tabellenformulare)

Außerdem können Sie im Formularassistenten Ihre eigenen Optionen als Vorlage abspeichern. Der Vorlagenname wird der Dropdown-Liste Vorlage, die in der Basisanzeige des Formularassistenten erscheint, hinzugefügt. Mit benutzerdefinierten Vorlagen können Sie schnell angepasste Formulare von der Basisanzeige des Formularassistenten durch einfaches Auswählen der gewünschten Datenfelder und Ihrer eigenen Vorlage erzeugen.

Seite Felder

Diese Seite ähnelt der Basisanzeige des Formularassistenten. Hier wählen Sie die Datenfelder für das Formular aus. Diese Funktionalität wurde für Benutzer dupliziert, die direkt zu den erweiterten Optionen gehen. Die ausführliche Beschreibung zum Hinzufügen von Datenfeldern im Formular finden Sie im vorigen Abschnitt.

Eingebbare verknüpfte Felder

Die Seite Felder bietet die zusätzliche Option **Eingebbare verknüpfte Felder**. Damit können Sie wählen, ob Sie den Datenfeldern verknüpfter Tabellen die Eigenschaft „Eingebbar“ zuweisen. Die Option ist standardmäßig markiert, d.h. die Felder sind eingebbar. Wenn Sie diese Option deaktivieren, können Benutzer die Werte in verknüpften Feldern nicht verändern, wenn für eine Verknüpfung die Option „Automatische Wertübertragung“ aktiviert ist. Weitere Informationen dazu finden Sie im Abschnitt **Verknüpfungseigenschaften**.

Weitere Informationen zur Eigenschaft "Eingebbar" finden Sie im Abschnitt **Attribute Eingebbar und Zwingend und Feldeigenschaften**.

Seite Optionen

Hier bestimmen Sie verschiedene Einstellungen für die Größe, die Bezeichnungen und die Darstellung des Formulars

Formulargröße

Hier bestimmen Sie die Größe des Formulars. Sie können die Formulargröße an den Inhalt anpassen oder eine feste Formulargröße über die maximalen Breite und Höhe oder durch Auswählen einer Bildschirmgröße definieren. Sie können die beiden Arten auch kombinieren. Die Dropdown-Liste *Bildschirm* bietet folgende Möglichkeiten:

- Automatik (Größe passt sich an den aktuellen Bildschirm an)
- 640x480 (niedrige Auflösung)
- 800x600 (SVGA)
- 1024x768 (Standard)
- 1280x1024 (hohe Auflösung)
- 1600x1200 (Großer Bildschirm)
- 2048x1536 (QXGA)

Die Werte entsprechen der Bildschirmauflösung für Breite x Höhe, angezeigt in Pixel.

Wenn Sie eine Bildschirmgröße eingeben oder eine Größe aus der Dropdown-Liste wählen, wird die Änderung in der Vorschau angezeigt. Der Formularassistent versucht, Datenfeld- und Objektposition im Formular anzupassen, so dass alle Formularobjekte auf dem ausgewählten Bildschirm Platz haben. Ist die Option **Erzeuge wenn notwendig mehrere Seiten** markiert (siehe unten), generiert er mehrere Seiten, um alle Datenfelder im Formular unterzubringen. In diesem Fall werden Schaltflächen, Formulartitel und Schmuckrahmen auf die Hintergrundseite (Seite 0) gesetzt.

- **Größe anpassen**: Wenn Sie auf dieses Kontrollkästchen klicken, entfernt der Formularassistent evtl. vorhandene Leerräume um die Felder, um die Größe des Formulars an den Inhalt anzupassen.
- Für **Formulare zum Drucken** (Listenformular zum Drucken und Detailformular zum Drucken):
 - Der Bereich "Formulargröße" enthält die Schaltfläche **Papierformat**. Damit öffnen Sie das aktuelle Dialogfenster mit den Druckereinstellungen. Hier können Sie die Papiergröße zum Drucken des Formulars wählen. 4D passt die Größe des Formulars und die Vorschau an das hier gewählte Papierformat an.
 - Im Bereich "Optionale Elemente" können Sie Variablen für Ihre Formulare auswählen, um Formulartitel, Seitennummer bzw. Druckdatum und Druckzeit anzuzeigen.
- Für **Listenformulare**: Der Bereich "Formulargröße" enthält die Option **Zielbreite**. Ist diese Option nicht markiert, wird die Größe des Formulars automatisch anhand der Größe der Felder im Formular berechnet. Markieren Sie diese Option und geben eine Breite in Pixel vor, versucht der Formularassistent, alle Felder an die vorgegebene Breite anzupassen, d.h. die Größe der Felder wird reduziert. Markieren Sie auch **Bei Bedarf abschneiden**, entfernt der Formularassistent ein oder mehrere Felder, so dass die Formularbreite geringer als die vorgegebene Breite ist. Markieren Sie diese Option nicht, kann das Formular etwas breiter als der für **Zielbreite** vorgegebene Wert sein.

Beschriftung

Hier legen Sie fest, wo ein Datenfeldname in Bezug zum Datenfeld gesetzt wird. Sie können die Namen entweder vor oder über die Datenfelder legen.

Optionale Elemente

Hier können Sie im Formular optionale Elemente mit bestimmten Eigenschaften hinzufügen. Sie können wählen zwischen:

- **Formulartitel**: Fügt den Namen der Tabelle als Formulartitel oberhalb der Datenfelder hinzu. Diese Option ist für Projektformulare nicht verfügbar.
- **Ein Feld pro Zeile**: Markieren Sie diese Option, um die Datenfelder vertikal anzuordnen. Ist diese Option inaktiv, versucht der Formularassistent, die Datenfelder in Zeilen anzuordnen.

- **Erzeuge wenn notwendig mehrere Seiten:** Markieren Sie diese Option, wenn der Formularassistent automatisch weitere Seiten erzeugen soll, sobald die Datenfelder nicht auf eine Seite passen. Wenn Sie diese Option anwenden, setzt der Formularassistent die entsprechenden Objekte auf die Hintergrundseite.
- **Benutze dynamische Feldnamen:** Markieren Sie diese Option, werden die Tabellen- und Feldnamen als dynamische Referenzen in das Formular eingefügt. 4D aktualisiert sie dann automatisch, wenn Sie diese Namen im Strukturfenster oder mit den 4D Befehlen **SET FIELD TITLES** oder **SET TABLE TITLES** verändern. Weitere Informationen dazu finden Sie im Abschnitt **Referenzen in statischem Text verwenden**.
- **Menüleiste:** Markieren Sie diese Option und wählen Sie den Namen der Menüleiste aus, die Sie mit dem Formular verknüpfen wollen. Weitere Informationen dazu finden Sie im Abschnitt **Menüleiste einem Formular zuordnen**.
- **Datensatznummer/-Anzahl:** Fügt eine 4D Variable in das Formular ein, welche die aktuelle Datensatznummer und die gesamte Datensatzanzahl ausgeben. Sie heißt standardmäßig *vRecNum*. Diese Option ist für Projektformulare nicht verfügbar.

Seite Schaltflächen

Hier richten Sie die im Formular verwendeten Schaltflächen nach Ihren Wünschen ein:

Hinweis: Diese Seite ist für die Formulartypen "Listenformular für Drucken" und "Detailformular für Drucken" nicht verfügbar.

Listen- und Detailformulare (nicht für Drucken) verwenden Schaltflächen, über die der Anwender Änderungen an einem Datensatz sichern bzw. löschen oder in einem mehrseitigen Formular von einer Seite zur nächsten springen kann. Tabellenformulare können auch Schaltflächen haben, um von einem Datensatz zum nächsten (erster Datensatz, letzter Datensatz, nächster Datensatz, voriger Datensatz) zu gelangen, Datensätze in einem Unterformular hinzuzufügen oder zu löschen bzw. den aktuellen Datensatz zu löschen.

Auf der Seite Schaltflächen können Sie ein Schaltflächenlayout auswählen, die gewünschten Schaltflächenaktionen bestimmen, die Position der Schaltflächen im Formular festlegen und jede Schaltfläche benennen.

Hinweis: Im Formulareditor können Sie einzelne Schaltflächen hinzufügen, löschen oder verschieben und einer Schaltfläche eine Methode zuweisen, die ihre Aktion beim Anklicken festlegt.

Schaltflächenfamilie und Position

Hier legen Sie Stil und Position der Schaltflächen fest.

Wählen Sie in der Dropdown-Liste Schaltflächen-Familie eine Familie und klicken auf die Pfeiltaste [**<**] oder [**>**], um jede Schaltfläche in der Vorschau anzusehen.

Automatische Schaltflächenaktionen

4D bietet einen Satz integrierter Schaltflächenaktionen. Wenn Sie einer Schaltfläche eine vordefinierte Aktion zuweisen, müssen Sie keine Methode schreiben, die eine bestimmte Aktion auslöst, wenn der Benutzer auf die Schaltfläche klickt. Die Anzahl der integrierten Aktionen richtet sich nach der Art des Formulars (Tabelle oder Projekt). So sind für Projektformulare Aktionen zum Navigieren in den Datensätzen einer Tabelle nicht möglich.

Der Formularassistent stellt für Detailformulare folgende Schaltflächenaktionen zur Verfügung:

- **OK:** Sichert einen neuen Datensatz bzw. Änderungen an einem bestehenden Datensatz
- **Abbrechen:** Annulliert den neuen Datensatz bzw. Änderungen an einem bestehenden Datensatz
- **Nächste Seite, Vorige Seite, Erste Seite, Letzte Seite:** Zeigt in einem mehrseitigen Formular die entsprechende Seite an

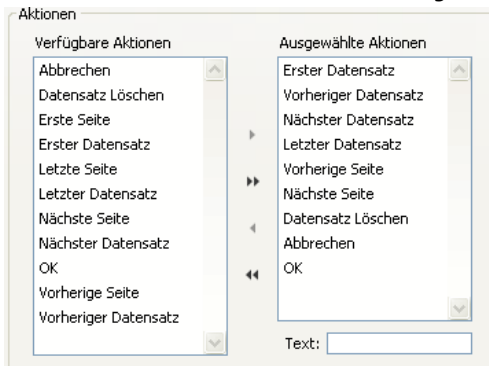
Nachfolgende Aktionen sind nur für Tabellenformulare verfügbar:

- **Next Record, Previous Record, First Record, Last Record:** Sichert den aktuellen Datensatz und zeigt den gewünschten Datensatz an.
- **Datensatz löschen:** Löscht den aktuellen Datensatz aus der Datenbank. Es erscheint ein Dialogfenster zum Bestätigen.

Hinweis: 4D fügt automatisch zwei zusätzliche Schaltflächen ein, wenn Sie ein Unterformular hinzufügen und auf der Seite „Unterformular“ die Option „Hinzufügen und Löschen Schaltflächen“ aktiv ist. Diesen Schaltflächen sind folgende Aktionen zugeordnet: **Hinzufügen** (fügt der Viele-Tabelle oder der Untertabelle einen neuen Datensatz hinzu) und **Löschen** (löscht den aktuellen Datensatz des Unterformulars).

4D bietet weitere vordefinierte Aktionen. Diese sind verfügbar, wenn Sie Ausgabeformulare über den Formularassistenten erstellen oder ein Formular, egal welchen Typs, im Formulareditor ändern. Weitere Informationen finden Sie im Abschnitt **Standardaktionen**.

Die Schaltflächen sind im Bereich *Verfügbare Aktionen* aufgelistet:



Sie können automatische Schaltflächen in der gleichen Weise aus- und abwählen, wie Sie Datenfelder im Formular hinzufügen oder löschen. Die Schaltflächen, die Sie im Bereich „Ausgewählte Aktionen“ hinzufügen, erscheinen im Formular.

Hinweis: Auch wenn die Schaltflächen zur Seitenverwaltung (Nächste Seite, Vorige Seite, etc.) ausgewählt sind, erscheinen sie im Formular nur, wenn ein mehrseitiges Formular erstellt werden muss.

Um eine vorgegebene Bezeichnung zu ändern (die Vorlage muss Bezeichnungen enthalten), markieren Sie die Schaltfläche in der Liste „Ausgewählte Aktionen“ und geben im Eingabebereich eine neue Bezeichnung ein. Drücken Sie anschließend die

Tabulatortaste oder klicken Sie auf eine andere Schaltfläche in der Liste. Die Vorschau zeigt dann die neue Bezeichnung an.
Hinweise:

- Schaltflächennamen lassen sich auch als Referenz eingeben. Das erleichtert die Übersetzung einer Anwendung in andere Sprachen (siehe **PICTURE TO BLOB**).
- Hilfetexte sind von den Schaltflächennamen unabhängig. Wollen Sie eine Schaltfläche mit einem Hilfetext versehen, verwenden Sie die Eigenschaftensliste des Formulareditors. Weitere Informationen dazu finden Sie im Abschnitt **Hilfemeldungen**.

Seite Unterformulare

Hier fügen Sie dem Formular ein Unterformular hinzu. Dieses Unterformular muss von einer verknüpften Viele-Tabelle kommen, damit Sie die automatischen Update Mechanismen für Unterformulare nutzen können.

Wollen Sie Datenfelder von einer verknüpften Viele-Tabelle oder von einer Untertabelle verwenden, fügen Sie im Formular ein Unterformular ein. Es listet mehrere Datensätze auf einmal auf. Mit einem Unterformular können Sie verknüpfte Datensätze oder Datensätze aus einer anderen Tabelle anzeigen. Sie können auch Informationen in Datensätze eingeben, die im Unterformular angezeigt werden.

Sie können Datenfelder von einer verknüpften Viele-Tabelle oder einer nicht verknüpften Tabelle in einem Unterformular anzeigen. Wenn Sie Datenfelder von einer verknüpften Viele-Tabelle einbeziehen, bestimmt die Verknüpfung, welche Datensätze angezeigt werden. Wenn Sie Datenfelder von einer nicht verknüpften Tabelle oder von einer Tabelle mit einer manuellen Verknüpfung einbeziehen, wird standardmäßig die aktuelle Auswahl von Datensätzen dieser Tabelle angezeigt. Sie können die Auswahl der Datensätze auch über eine Methode steuern.

Die Seite Unterformular ermöglicht, ein Formular als Unterformular zu verwenden, Optionen dafür zu definieren und Schaltflächen hinzuzufügen, über die Benutzer mit dem Unterformular arbeiten können.

Um dem Formular ein Unterformular hinzuzufügen, markieren Sie die Option **Unterformular hinzufügen**. Wählen Sie die Tabelle für das Unterformular aus der DropDown-Liste "Tabelle" aus und dann das gewünschte Unterformular aus der DropDown-Liste "Ausgabeformular". Sie können auch die DropDown-Liste "Eingabeformular" verwenden, um die Formularseite zu setzen, die angezeigt werden soll, wenn der Benutzer auf das Unterformular klickt. Dieses Unterformular erscheint dann in der Vorschau als Teil des aktuellen Formulars.

Sie können für das Unterformular folgende Optionen definieren:

- **Mehrfach-Auswahl:** Damit kann der Benutzer mehrere Unterdatensätze gleichzeitig auswählen, unter Windows mit der Tastenkombination **Umschalttaste+Strg**, auf Mac OS **Umschalt- +Befehlstaste**.
- **Eingebbar in Liste:** Damit kann der Benutzer die Werte von Unterdatensätzen direkt im Unterformular ändern.
- **Hinzufügen und Löschen Schaltflächen:** Damit fügt der Formularassistent im Formular automatisch zwei Schaltflächen mit den Standardaktionen Unterdatensatz hinzufügen und Unterdatensatz löschen hinzu.



Weitere Informationen dazu finden Sie im Abschnitt **Unterformulare als Liste**.

Erweitertes Formular erstellen

Wenn Sie mit dem Festlegen aller Eigenschaften des neuen Formulars fertig sind, klicken Sie auf einer beliebigen Seite auf die Schaltfläche **OK**, um das neue Formular zu erzeugen. Sie erhalten folgendes Dialogfenster:

Um das neue Formular zu erzeugen, klicken Sie entweder auf die Schaltfläche **Benutzen**, um das Formular zu testen (nur Tabellenformulare) oder auf die Schaltfläche **Bearbeiten**, um das neue Formular im Formulareditor zu öffnen.

Formularvorlage erstellen


Das Dialogfenster zum Bestätigen des Formulars bietet Ihnen die Möglichkeit, eine neue Formularvorlage mit den aktuellen Einstellungen der erweiterten Optionen (Schaltflächen, Optionen, etc.) zu erzeugen. Wenn Sie eine Formularvorlage erzeugen, wird ihr Name auf der Hauptseite des Formularassistenten in die Dropdown-Liste **Benutze Vorlage** eingefügt. Die Formularvorlage wird zusätzlich zum Formular selbst gesichert.

Um eine eigene Vorlage zu erstellen, markieren Sie die Schaltfläche **Ja** und geben im Bereich *Vorlagename* einen Namen ein.

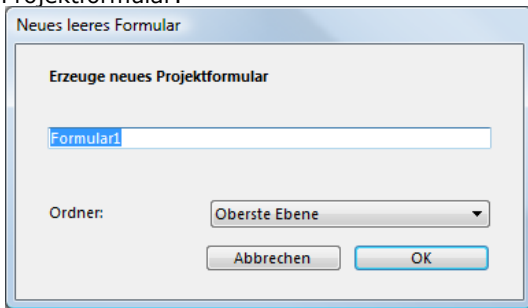
Leeres Formular erstellen

Sie können ein leeres Formular direkt aus dem **Explorer** ohne Einsatz des Formularassistenten erstellen. Es wird dann ohne Datenfeld, Schaltfläche oder Variable angelegt und im Formulareditor geöffnet. Sie können es vollständig in diesem Editor aufbauen. Leere Formulare sind hilfreich zum Erstellen von Dialogfenstern, die nur Variablen oder Plug-In Bereiche enthalten..

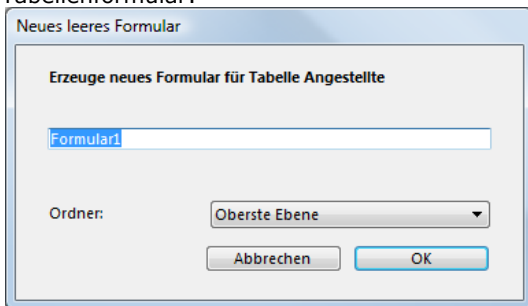
Um ein leeres Formular zu erstellen:

1. Gehen Sie auf die **Seite Formulare** des Explorers.
2. Wählen Sie den passenden Eintrag:
 - Für ein Projektformular entweder den Eintrag "Projektformulare" oder ein vorhandenes Projektformular,
 - Für ein Tabellenformular den Namen einer Tabelle.
3. Klicken Sie am unteren Rand des Explorers auf das Icon . Auf dem Bildschirm erscheint jeweils folgendes Dialogfenster

- **Projektformular:**



- **Tabellenformular:**



4. Tragen Sie im Eingabebereich einen Namen für das Formular ein.
Diesen Name verwenden Sie insbesondere beim Programmieren als Referenz.
5. (Optional) Wählen Sie einen Ordner zum Abspeichern des Formulars.
Wählen Sie einen Ordnersnamen aus der DropDown-Liste aus, wird das Formular dort abgelegt. Ordner sind nützlich, um die Objekte Ihrer Anwendung nach Funktionalität oder eigenen Kriterien zu organisieren. Sie werden über die **Seite Home** des Explorers verwaltet. Standardmäßig wird das Formular auf der obersten Ebene angelegt, d.h. außerhalb der Ordner.
6. Klicken Sie auf die Schaltfläche **OK**.
Das leere Formular öffnet sich in einem neuen Fenster des **Formulareditor**.

Hinweis zur Kompatibilität: Seit mehreren Versionen fügt 4D nicht mehr automatisch Schaltflächen zum Navigieren in Formularen hinzu, wenn Sie keine eigenen angelegt haben. Sie müssen bei Bedarf Ihren eigenen Satz Schaltflächen zum Navigieren erstellen.

Formular bearbeiten

Sie können ein Formular im Formulareditor folgendermaßen öffnen:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Öffnen>Formular**.
ODER
Doppelklicken Sie im Strukturfenster unter Windows bei gedrückter Tastenkombination **Strg-Taste+Umschalttaste**, auf Mac OS mit gedrückter **Befehlstaste+Umschalttaste** auf den Tabellentitel, dessen Formulare Sie bearbeiten wollen
ODER
Klicken Sie im Strukturfenster mit der rechten Maustaste auf den Tabellentitel, und wählen im Kontextmenü den Eintrag **Zeige Tabellen-Formulare**.
4D zeigt die **Seite Formulare** des Explorers an.
2. Sie können den Eintrag "Projektformulare" oder jeden Tabellennamen erweitern, um die damit verknüpften Formulare anzuzeigen. Bei Doppelklick auf einen Tabellennamen oder Öffnen des Kontextmenüs im Struktureditor ist die Tabelle bereits ausgewählt.
3. Bei Bedarf können Sie den Tabellennamen, der das zu ändernde Formular enthält, erweitern.
4. Klicken Sie mit der rechten Maustaste auf den Namen des gewünschten Formulars und wählen im Kontextmenü den Eintrag **Formular bearbeiten**.
ODER
Wählen Sie das Formular und dann im Explorer am unteren Rand im Menü **Optionen** den Eintrag **Formular bearbeiten**.
ODER
Doppelklicken Sie auf den Namen des Formulars oder in die Vorschau.
4D zeigt das Formular in einem Formulareditorfenster.
Hinweis: Sie können diese Aktionen auch auf der **Seite Home** des Explorers ausführen.
4D Server: Gesperrte Objekte treten auf, wenn zwei oder mehr Benutzer gleichzeitig versuchen, dasselbe Formular zu ändern. Wenn ein Benutzer ein Formular in der Designumgebung ändert, wird das Formular gesperrt. Andere Benutzer können das gleiche Formular solange nicht ändern, bis der erste Benutzer das Formular durch Schließen wieder freigibt.

Formular umbenennen

Sie können das Formular in der Eigenschaftensliste oder über die **Seite Formulare** im Explorer umbenennen.

Sie verwenden die Formularnamen, wenn Sie Standardeingabe- und Standardausgabeformulare für eine Tabelle erzeugen und in Befehlen, die einen Formularnamen als Parameter verwenden, z.B. die 4D Befehle **FORM SET INPUT** und **FORM SET OUTPUT**.

Sie können denselben Namen nur für ein Projektformular bzw. ein Formular pro Tabelle verwenden. Ansonsten entsteht bei 4D Verwirrung, wenn Sie versuchen, sich auf ein Formular per Namen zu beziehen. Sie können aber ohne weiteres denselben Namen in verschiedenen Tabellen verwenden. Sie können zum Beispiel alle Ihre Eingabeformulare "Eingabe" und alle Ihre Ausgabeformulare "Ausgabe" nennen.

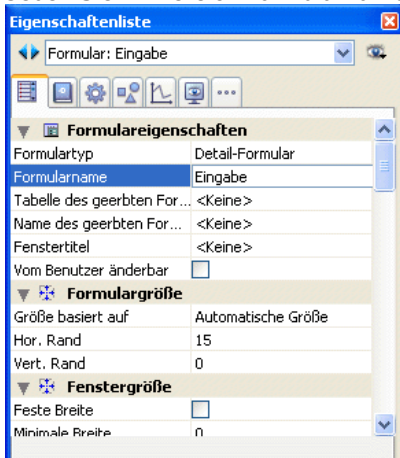
Warnung: Wenn Sie ein Formular umbenennen, auf das an anderen Stellen in der Datenbank Bezug genommen wird (wie z.B. in Methoden), müssen Sie die Bezüge auf das Formular aktualisieren. Dazu können Sie die Funktion **Suchen und Ersetzen** in der Designumgebung verwenden (siehe **Umbenennen**).

Um ein Formular im Explorer umzubenennen:

1. Zeigen Sie die **Seite Formulare** des Explorer an.
Eine hierarchische Liste mit Tabellen und Formularen erscheint.
2. Drücken Sie unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** und klicken auf den Formularnamen.
ODER
Klicken Sie zweimal auf den Namen des entsprechenden Formulars.
Der Formularname wird editierbar.
3. Geben Sie den neuen Namen ein.
4. Drücken Sie die Tabulatortaste oder klicken Sie außerhalb des Ebenenbereichs, um den neuen Namen zu speichern.

Um ein Formular über die Eigenschaftensliste umzubenennen (siehe **Eigenschaftensliste verwenden**):

1. Zeigen Sie die Formulareigenschaften in der Eigenschaftensliste an
2. Geben Sie im Bereich Formularname einen Namen ein.



Sie können ein Formular auch über das Fenster Formulareigenschaften des Explorer umbenennen (siehe **Formulareigenschaften (Explorer)**).

Eingabe- und Ausgabeformulare festlegen

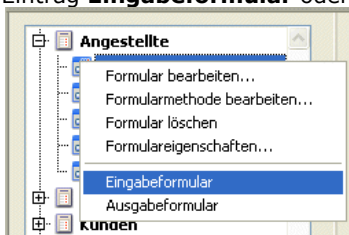
Jede Tabelle hat ein aktuelles Eingabeformular und ein aktuelles Ausgabeformular. Im allgemeinen verwenden Sie ein Detailformular zum Eingeben und Ändern von Datensätzen, ein Listenformular zum Auflisten der Datensätze.

Sie können jederzeit bestimmen, welches Formular Sie für die Eingabe bzw. Ausgabe verwenden. Sie können die Änderung im **Fenster Liste der Tabellen** als auch mit den 4D Befehlen **FORM SET INPUT** und **FORM SET OUTPUT** ausführen. In diesem Fall gilt die Festlegung nur für die aktuelle Arbeitssitzung.

Sie können das Standardeingabe- bzw. -ausgabeformular auch in der Designumgebung festlegen. In diesem Fall wird die Festlegung mit der Datenbank gesichert.

Um das Eingabe- und das Ausgabeformular für eine Tabelle zu ändern:

1. Zeigen Sie die **Seite Formulare** des Explorers an.
2. Erweitern Sie die Tabelle, für welche Sie das standardmäßige Eingabe- bzw. Ausgabeformular ändern wollen.
Der Buchstabe **E** erscheint neben dem Namen des aktuellen Eingabeformulars, der Buchstabe **A** erscheint neben dem Namen des aktuellen Ausgabeformulars.
3. Klicken Sie mit der rechten Maustaste auf den entsprechenden Formularnamen und wählen Sie im Kontextmenü den Eintrag **Eingabeformular** oder **Ausgabeformular**:



ODER

Wählen Sie am unteren Rand des Explorers im Menü Optionen den Eintrag **Eingabeformular** oder **Ausgabeformular**.


Hinweis: Nur Tabellenformulare lassen sich als Eingabe- oder Ausgabeformular bestimmen.

Sie können ein Formular auch gleichzeitig als aktuelles Eingabe- und Ausgabeformular definieren. Neben dem Formularnamen erscheint dann der Buchstabe **B**.

Formular löschen

Sie können jedes Projekt- oder Tabellenformular löschen, das nicht als aktuelles Eingabe- oder Ausgabeformular (oder beides) festgelegt ist. Für diese Formulare ist die Schaltfläche **Löschen** deaktiviert.

Um ein Formular zu löschen:

1. Zeigen Sie die **Seite Formulare** des Explorers an.
2. Erweitern Sie den Eintrag Projektformulare oder die Tabelle, die das zu löschende Formular enthält.
3. Wählen Sie das gewünschte Formular aus und klicken am unteren Rand des Explorers auf das Icon  des Explorers.
ODER
Klicken Sie mit der rechten Maustaste auf den Formularnamen und wählen im Kontextmenü den Eintrag **Formular löschen**.
4D fordert Sie auf, das Löschen zu bestätigen.
4. Klicken Sie auf die Schaltfläche **OK**.
4D löscht das Formular. Das Formular wird in den Papierkorb gelegt und kann wieder daraus entnommen werden, solange der Papierkorb nicht gelöscht wird (siehe **Seite Papierkorb**).

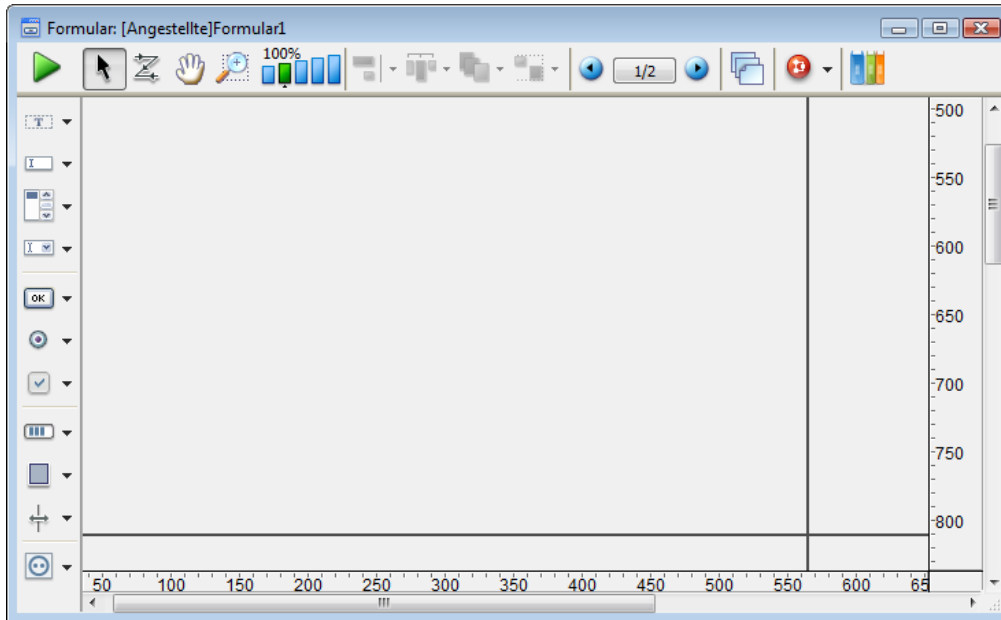
Hinweis: Auf dieselbe Weise können Sie ein Formular auch auf der **Seite Home** des Explorers löschen.

Formular drucken

Jedes Formular kann bis zu 3 qm groß sein. Über die Rollbalken können Sie nicht angezeigte Teile des Formulars in Sicht bringen. Zur Anzeige auf dem Bildschirm kann Ihr Formulardesign den gesamten Bereich verwenden. Durch Scrollen können Sie jedes Element ansehen, das Sie ins Formular setzen.

Zum Drucken müssen Formularelemente auf eine Seitenbreite passen, in der Länge können sie mehrere Seiten umfassen. Die tatsächliche Seitengröße hängt von Ihrem Drucker, dem verwendeten Papier und den Einstellungen, die Sie im Dialogfenster Seiteneinstellung festlegen, ab. 4D zeigt Seitenrandlinien im Formulareditor. Sie ändern sich je nach Seiteneinstellung. Die Seiteneinstellungen werden beim Schließen mit dem Formular gespeichert. Die Begrenzung des Formulars lässt sich über den Befehl Papier im Untermenü Anzeige des Menüs Formular ein- bzw. ausblenden. Weitere Informationen dazu finden Sie im Abschnitt **Elemente im Formulareditor ein-/ausblenden**.

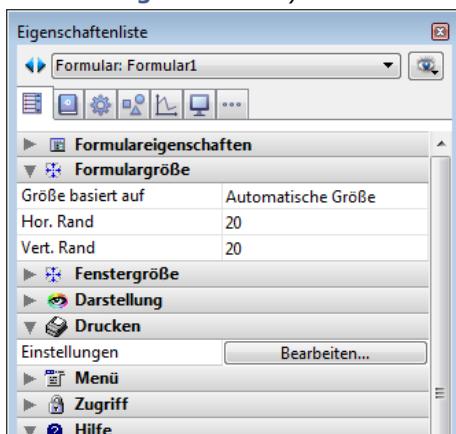
Ihr Formular mit Seitenrändern könnte folgendermaßen aussehen:



Druckeinstellungen für Formular

Sie können für jedes Formular spezifische Einstellungen zum Drucken definieren. Sie werden berücksichtigt, wenn das Formular im Anwendungsmodus gedruckt wird.

Dazu klicken Sie in der Eigenschaftensliste des Formulars unter der Gruppe "Drucken" auf die Schaltfläche **Bearbeiten** (siehe **Formulareigenschaften**).



Auf dem Bildschirm erscheint der Druckdialog, wo Sie die spezifischen Druckeinstellungen des Formulars ändern können, wie Papierformat, Ausrichtung, etc. Die verfügbaren Optionen in diesem Standarddialog richten sich nach Ihrer Systemkonfiguration.

Weitere Informationen dazu finden Sie im Abschnitt **Datensätze drucken**.

Formulare speichern






















Es empfiehlt sich, alle Änderungen, die Sie an einem Formular vornehmen, zu speichern, insbesondere bei mehreren Benutzern mit 4D Server. Sie können ein Formular durch Schließen oder Speichern sichern. Sie können ein Formular schließen, wenn Sie auf die Schließbox klicken oder im Menü **Datei/Ablage** den Befehl **Schließen** wählen.

Wollen Sie ein Formular ohne Schließen sichern, wählen Sie im Menü **Datei/Ablage** den Befehl **Speichern**.

Wenn ein Formular gespeichert wurde, können Sie weiter daran arbeiten. Wenn Sie einen Fehler machen oder die ausgeführten Änderungen verwerfen wollen, können Sie auf die zuletzt gespeicherte Version des Formulars zurückgreifen. Dies öffnet das Formular mit der Darstellung der letzten Sicherung. Wählen Sie hierzu im Menü **Datei/Ablage** den Befehl **Vorherige Version**.

4D Server: Wurde ein Formular in der Designumgebung gespeichert, können Benutzer Ihre Änderungen beim nächsten Öffnen des Formulars sehen.

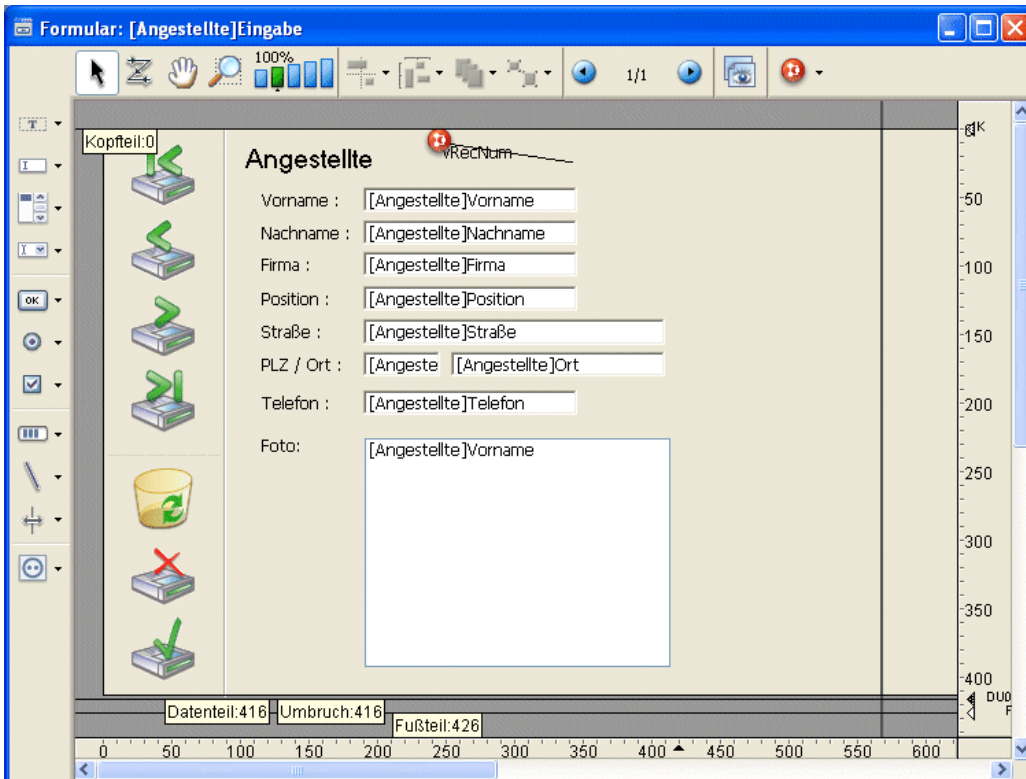
Formulare erstellen

-  Formulareditor
-  Formulareigenschaften
-  Formulareigenschaften (Explorer)
-  Mehrseitiges Formular erstellen
-  Vererbte Formulare verwenden
-  Eingabereihenfolge der Daten ändern
-  Formularobjekte einfügen und organisieren
-  Objekteigenschaften setzen
-  Textrotation
-  Statische Bilder verwenden
-  Referenzen in statischem Text verwenden
-  Auf Matrix duplizieren
-  Variablensatz erhöhen
-  Formular skalieren
-  Objektmethoden verwenden
-  Formularebenen verwenden
-  Erkennungszeichen verwenden
-  Vorkonfigurierte Objektbibliothek verwenden
-  Eigene Objektbibliothek anlegen und einsetzen
-  Information über Objekte in ausgeführten Formularen
-  Dynamische Formulare

Wenn Sie ein neues Formular mit dem Formularassistenten erzeugen, können Sie unter vielen Anpassungsoptionen wählen. Mit Vorlagen können Sie das Aussehen von Datenfeldern und Datenfeldbezeichnungen festlegen und einen Satz automatischer Schaltflächen hinzufügen.

Das ist aber nur ein Teil der Möglichkeiten, denn 4D bietet einen ausgeklügelten Formulareditor, mit dem Sie Ihr Formular so lange verändern können, bis es alle gewünschten Kriterien erfüllt. Mit dem Formulareditor können Sie Objekte erzeugen und löschen, Objekte direkt bearbeiten und Formular- und Objekteigenschaften zuweisen.

Der Formulareditor zeigt jedes Formular in einem eigenen Fenster mit Objekt- und Werkzeugleiste an. Es können mehrere Formulare gleichzeitig geöffnet sein. Die Lineale am rechten und unteren Rand erleichtern das Setzen von Objekten im Formular. Sie können die Skala in Zoll, Zentimetern oder Pixel anzeigen.



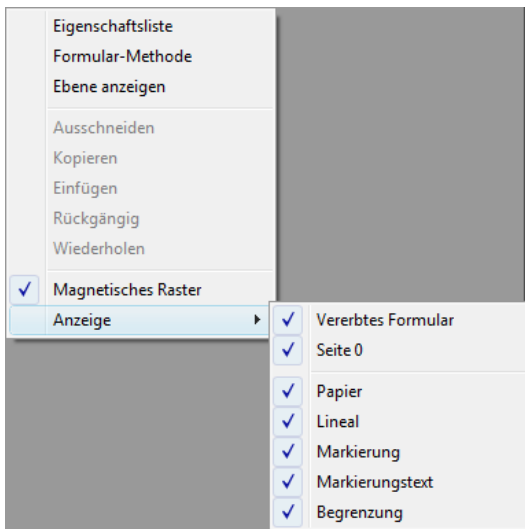
Dieser Abschnitt beschreibt die Grundelemente des Formulareditors. Der Abschnitt **Mit aktiven Objekten arbeiten** beschreibt ausführlich Felder und andere Objekte.

Elemente im Formulareditor ein-/ausblenden

Sie können die Oberflächenelemente im Formulareditor ein- bzw. ausblenden. So können Sie nur die Elemente anzeigen, die Sie zum Erstellen oder Ansehen im Formular benötigen oder nur die Tools, mit denen Sie arbeiten. Ein- und Ausblenden gilt immer für das aktuelle Fenster des Formulareditors. Sie können z.B. im Ausgabeformular die Steuerlinien ausblenden, da sie hier nicht benötigt werden.

Um ein Element im Formulareditor ein- oder auszublenden:

1. Wählen Sie im Menü **Formular** den Eintrag **Anzeige**.
ODER
Klicken Sie mit der rechten Maustaste in das Formular (jeweils ohne ein Objekt anzuklicken) und wählen im Kontextmenü den Eintrag **Anzeige**.
Auf dem Bildschirm erscheint ein hierarchisches Untermenü, das alle ein-/ausblendbaren Elemente auflistet:



Das Häkchen vor dem Element gibt an, dass es eingeblendet ist. Wollen Sie ein Element ausblenden, klicken Sie es an. Das Häkchen wird dann entfernt.



2. Wählen Sie ein Element, das Sie ein- bzw. ausblenden wollen.

Die Befehle führen folgendes aus:

- **Vererbtes Formular:** Blendet – sofern ein geerbtes Formular vorhanden ist – geerbte Formularobjekte in der aktuellen Seite des Formulars ein oder aus. Weitere Informationen dazu finden Sie im Abschnitt **Vererbte Formulare verwenden**.
- **Seite 0:** Blendet die Objekte von Seite 0 in der aktuellen Seite des Formulars ein/aus. Damit können Sie im Formular unterscheiden zwischen den Objekten der aktuellen Seite und den Objekten der Seite 0. Weitere Informationen dazu finden Sie im Abschnitt **Mehrseitiges Formular erstellen**.
- **Papier:** Blendet die Ränder der Druckseite ein/aus. Sie erscheinen als graue Linien. Diese Option hat keine sichtbare Auswirkung, wenn die Option **Begrenzung** ausgewählt ist. Ist das Formular kleiner als die Druckseite, liegen die Seitenränder außerhalb des Formularbereichs und erscheinen deshalb nicht. Weitere Informationen dazu finden Sie im Abschnitt **Formular drucken**.
- **Lineal:** Blendet im Formulareditor die Lineale ein - oder aus.
- **Markierung:** Blendet die Markierungen, die die Begrenzung der verschiedenen Formularebenen anzeigen, ein bzw. aus.
- **Markierungstext:** Blendet den Markierungstext ein bzw. aus, er ist nur verfügbar, wenn die Markierungslinien angezeigt werden. Weitere Informationen dazu finden Sie im Abschnitt **Steuerlinien für Ausgabe bewegen**.
- **Begrenzung:** Blendet die Formularbegrenzungen ein oder aus. Ist diese Option gewählt, wird das Formular im Formulareditor angezeigt, so wie es im Anwendungsmodus erscheint. Auf diese Weise können Sie Ihr Formular anpassen, und sehen sofort das Ergebnis, ohne in den Anwendungsmodus zu wechseln.

Hinweis: Die Einstellungen **Fenstergröße**, **Hor. Rand** und **Vert. Rand** in den Formulareigenschaften beeinflussen die Begrenzungen des Formulars. Verschieben oder vergrößern Sie ein Objekt in Randnähe, wird die Fenstergröße entsprechend angepasst. Weitere Informationen dazu finden Sie im Abschnitt **Formulareigenschaften**.

Zoom

Sie können das aktuelle Formular vergrößern oder verkleinern. Dazu klicken Sie in der Werkzeugleiste am oberen Rand entweder auf das Vergrößerungsglas , oder direkt daneben auf eins der Kästchen . Es gibt die Prozentsätze 50%, 100%, 200%, 400% und 800%.

- Klicken Sie auf das Vergrößerungsglas, um es zu aktivieren. Zum Vergrößern klicken Sie einfach auf das Formular, zum Verkleinern klicken Sie mit gedrückter **Umschalttaste** auf das Formular.
- Klicken Sie auf eins der Kästchen, ändert sich die Anzeige unmittelbar.

Beim Zoomen bleiben alle Funktionen des Formulareditors erhalten. *



* Aus technischen Gründen lassen sich keine Listbox Elemente (Kopfteil, Spalten, Fußteil) auswählen, wenn der Formulareditor im Zoom-Modus ist.

Werkzeugleiste verwenden













Die Werkzeugleiste am oberen Rand bietet einen Satz Funktionen zum Bearbeiten und Ändern von Formularobjekten. Jedes Fenster hat seine eigene Werkzeugleiste



Die Werkzeugleiste enthält folgende Elemente:












-  **Formular ausführen:** Damit testen Sie die Ausführung des Formulars. Klicken Sie auf das Icon, öffnet 4D ein neues Fenster und zeigt das Formular im jeweiligen Kontext an (Liste der Datensätze für das Listenformular, aktueller Datensatz für das Detailformular). Das Formular wird im Hauptprozess ausgeführt.
-  **Auswahl:** Damit wählen, bewegen und passen Sie Formularobjekte in der Größe an. Weitere Informationen dazu finden Sie im Abschnitt **Objekte auswählen**.

Hinweis: Ist ein Objekt vom Typ Text oder Gruppenrahmen ausgewählt, gelangen Sie durch Drücken der **Eingabetaste** in den Bearbeitungsmodus.

-  **Eingabefolge:** Wechselt in den Modus "Eingabefolge", wo Sie die aktuelle Eingabefolge im Formular ansehen und verändern können. Weitere Informationen dazu finden Sie im Abschnitt **Eingabereihenfolge**. Beachten Sie, dass Sie über Ebenen die aktuelle Eingabefolge ansehen können, während Sie im Formular arbeiten (siehe Abschnitt **Erkennungszeichen verwenden**).
-  **Bewegen:** Wechselt in den Modus "Bewegen", wodurch Sie per Drag&Drop im Fenster rasch zu jedem Teil des Fensters gelangen. Der Cursor nimmt die Form einer Hand an. Das ist besonders hilfreich bei vergrößertem Formular.
-  **Zoom:** Damit verändern Sie die Anzeige des Formulars in Prozent. Standardmäßig ist 100% vorgegeben. Klicken Sie entweder auf das Vergrößerungsglas oder eins der danebenliegenden Kästchen. Weitere Informationen dazu finden Sie im vorigen Abschnitt.
-  **Ausrichten:** Damit öffnen Sie ein DropDown-Menü, wo Sie Objekte auf verschiedene Weise aneinander ausrichten können. Ob es aktiviert ist, richtet sich nach den ausgewählten Objekten.
-  **Verteilen:** Damit öffnen Sie ein DropDown-Menü, um die Objekte im Formular horizontal oder vertikal anzuordnen. Ob es aktiviert ist, richtet sich nach den ausgewählten Objekten. Weitere Informationen dazu finden Sie im Abschnitt **Objekte ausrichten**.
-  **Position wechseln:** Damit öffnen Sie ein DropDown-Menü, um die Lage der Objekte im Formular zu verändern. Ob es aktiviert ist, richtet sich nach den ausgewählten Objekten. Weitere Informationen dazu finden Sie im Abschnitt **Objekte übereinanderlegen**.
-  **Gruppieren/Gruppierung aufheben:** Damit öffnen Sie ein DropDown-Menü, um Objektauswahlen im Formular zu gruppieren oder die Gruppierung aufzuheben. Ob es aktiviert ist, richtet sich nach den ausgewählten Objekten. Weitere Informationen dazu finden Sie im Abschnitt **Objekte gruppieren**.
-  **Seite anzeigen/vor- und zurückblättern:** Hier blättern Sie in den Seiten und löschen bzw. fügen Seiten hinzu. Sie können über die Pfeile in den Seiten blättern oder in die Mitte klicken und die entsprechende Seite im DropDown-Menü wählen. Um eine Seite anzufügen, klicken Sie auf den rechten Pfeil, wenn die letzte Seite des Formulars angezeigt wird. Weitere Informationen dazu finden Sie im Abschnitt **Mehrseitiges Formular erstellen**.
-  **Ansichten:** Damit blenden Sie die Palette Ansichten ein/aus. Weitere Informationen dazu finden Sie im Abschnitt **Formularebenen verwenden**.
-  **Erkennungszeichen:** Mit jedem Klick auf dieses Icon zeigen Sie nacheinander die Erkennungszeichen für das Formular an. Über den Pfeil öffnen Sie ein DropDown-Menü, um das entsprechende Erkennungszeichen direkt zu wählen. Weitere Informationen dazu finden Sie im Abschnitt **Erkennungszeichen verwenden**.
-  **Vorkonfigurierte Objektbibliothek:** Damit öffnen Sie die vorkonfigurierte Objektbibliothek. Sie enthält unzählige Objekte mit vordefinierten Eigenschaften. Weitere Informationen dazu finden Sie im Abschnitt **Vorkonfigurierte Objektbibliothek verwenden**.
-  **Benutzersperre:** Dieses Icon gibt an, ob das Formular "Durch Benutzer editierbar ist". Es lässt sich nicht direkt bearbeiten. Sie müssen zuerst auf dieses Icon klicken, um die Sperre zu lösen. Weitere Informationen dazu finden Sie im Abschnitt **Vom Benutzer editierbar**.

Objektliste verwenden

Die Objektliste am linken Rand enthält alle aktiven und inaktiven Objekte, die Sie in 4D Formularen verwenden können. Manche Objekte sind nach Themen gruppiert, z.B. Schaltfläche. Jedes Thema enthält mehrere Varianten, zwischen denen Sie wählen können. Hat die Objektliste Fokus, können Sie die Schaltflächen über Tastenkürzel auswählen. Die folgende Tabelle beschreibt die verfügbaren Objektgruppen und die zugeordneten Tastenkürzel:

Schaltfläche	Gruppe	Taste
	Text / Gruppenrahmen	T
	Feld / Variable	F
	Rollbarer Bereich / Hierarchische Liste / Listbox	L
	Combobox / PopUp/DropDown-Liste / Hierarchisches PopUp-Menü / Bild PopUp- Menü	P
	Schaltfläche / 3D Schaltfläche / Highlight Schaltfläche / Unsichtbare Schaltfläche / Bildschaltfläche / Schaltflächengitter	B
	Optionsfeld / 3D Optionsfeld / Optionsfeld mit Bild	R
	Kontrollkästchen / 3D Kontrollkästchen	C
	Ablaufbalken / Halbkreissskala / Lineal	I
	Rechteck / Linie / abgerundetes Rechteck / Oval / Matrix	S
	Splitter / Registerkarte	D
	Plug-In Bereich / Unterformular /Web Area	X

Um einen Objekttyp zu zeichnen, wählen Sie die entsprechende Schaltfläche und zeichnen das Objekt im Formular. Ist das Objekt angelegt, können Sie seine Eigenschaften über die Eigenschaftenliste verändern. Erstellen Sie ein Objekt mit gedrückter **Umschalttaste**, um es auf bestimmte Formen einzuschränken, d.h. Linien lassen sich dann nur horizontal, vertikal oder im 45° Winkel zeichnen, Rechtecke erscheinen quadratisch, Ovale als Kreis.

Die aktuelle Variante des Themas ist das Objekt, welches in das Formular eingefügt wird. Klicken Sie rechts neben eine Schaltfläche, öffnet sich das DropDown-Menü mit weiteren Varianten:



Ist eine Schaltfläche ausgewählt, können Sie die dazugehörigen Varianten über die Tastenkombination **Shift+Auswahl** aufklappen. Hilfetipps zeigen die aktuell ausgewählte Variante an und die zugewiesene Taste zum Auswählen des Objekts. Klicken Sie zweimal auf die Schaltfläche, bleibt sie nach Zeichnen eines Objekts im Formular ausgewählt (kontinuierliche Auswahl). Auf diese Weise können Sie nacheinander rasch mehrere Objekte vom selben Typ erstellen. Um die kontinuierliche Auswahl aufzuheben, klicken Sie auf ein anderes Objekt oder Werkzeug.

Eigenschaftenliste verwenden

Formulare und Formularobjekte haben Eigenschaften, die Zugriff, Aussehen und Verhalten des Formulars bei Gebrauch festlegen. Formulareigenschaften sind z.B. Formularname, Zuordnung zu einer Menüleiste; Objekteigenschaften sind z.B. Objektname, Hintergrundfarbe und Schriftstil.

Sie können die Formular- und Objekteigenschaften über die Eigenschaftenliste anzeigen und ändern. Sie zeigt entweder die Formular- oder die Objekteigenschaften, je nachdem, was Sie im Editorfenster ausgewählt haben.

Um die Eigenschaftenliste ein- bzw. ausblenden:

1. Wählen Sie im Menü **Formular** den Befehl **Eigenschaftenliste** oder im Kontextmenü des Formulareditors den Eintrag **Eigenschaftenliste**. Es erscheint, wenn Sie bei gedrückter rechter Maustaste in einen leeren Bereich des Formulars klicken.
ODER
Verwenden Sie unter Windows die Tastenkombination **Strg- + Umschalttaste + Leerschritt**, auf Macintosh **Befehlstaste + Umschalttaste + Leerschritt**.

Die Eigenschaftenliste sieht folgendermaßen aus:



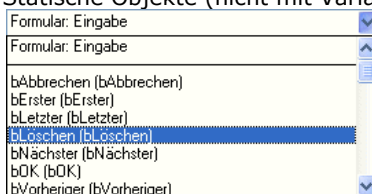
Auswahl des Inhalts

Die Eigenschaftenliste zeigt entweder die Eigenschaften des Formulars oder des bzw. der ausgewählten Objekte:

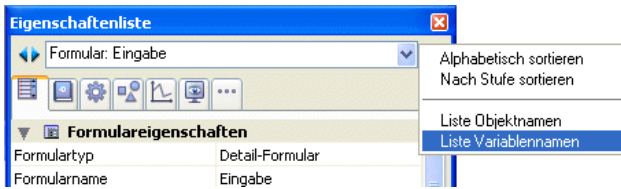
- Ist kein Objekt ausgewählt, erscheinen die Eigenschaften des Formulars,
- Ist ein Objekt ausgewählt, erscheinen die Eigenschaften des Objekts, bei mehreren Objekten die gemeinsamen Eigenschaften.

Die Eigenschaften werden angezeigt, wenn Sie das Objekt im Formular anklicken oder es in der Auswahlliste oben in der Eigenschaftenliste wählen. Die Liste ist in drei Bereiche gegliedert:


- Formularname,
- Dynamische Objekte (verknüpft mit Variable),
- Statische Objekte (nicht mit Variable verknüpft)



Die Liste zeigt für jedes dynamische Objekt den Variablennamen, gefolgt vom Objektname zwischen Klammern oder nur den Objektname. Über das zugeordnete DropDown-Menü können Sie definieren, welcher Name zuerst erscheint und wie sortiert wird:




Sie können die Sortierung jedes Listenteils ändern. Die Liste wird standardmäßig in alphabethischer Reihenfolge sortiert. Sie können alternativ auch nach Ebenen sortieren, d.h. von der obersten Ebene bis zur hintersten Ebene.

Über die Pfeile vor der Liste  können Sie jedes Objekt der Reihe nach aufrufen. Sie können vor- und zurückblättern.

Die Seiten der Eigenschaftenliste

Über die Registerkarten wählen Sie, wie die Information in der Eigenschaftenliste angezeigt wird:

- Die erste Seite  (Basis) zeigt alle Informationen zum gewählten Objekt, gegliedert nach den verschiedenen Themen, z.B. Datenquelle, Koordinaten&Größe, Ereignisse. Über den Pfeil vor dem Namen können Sie jedes Thema auf- oder zuklappen, so dass nur die Eigenschaften erscheinen, mit denen Sie arbeiten wollen. Die Anzeige richtet sich nach dem gewählten Objekttyp. Manche Eigenschaften ändern sich dynamisch, da sie von Werten anderer Eigenschaften abhängen. Setzen Sie z.B. für ein Feld die Eigenschaft **Eingebbar**, erscheint auch die Eigenschaft **Tabfähig**.
- Die weiteren Seiten enthalten spezifische Informationen zum gewählten Objekt. Jedes Thema zeigt alle Eigenschaften an, unabhängig vom gewählten Objekttyp. Nicht zutreffende Eigenschaften sind in Grauschrift. Diese Alternative bietet eine globalere Übersicht über die Einstellungen, die einem bestimmten Objekt zugeordnet sind. Es folgt die Beschreibung der sechs Themenseiten:
 - **Einstellung**: Enthält die allgemeinen Angaben zum Objekt wie Typ, Name, Datenquelle, etc.
 - **Verhalten**: Enthält die Angaben zum dynamischen Verhalten der Objekte, wie zugeordnete Methode, Drag&Drop, Formularereignisse, etc.
 - **Größe**: Enthält die Werte für Koordinaten, Größe und Anpassung der Objekte.
 - **Wert**: Enthält Angaben zum Bearbeiten und Anzeigen von Objekten, wie eingebbar, Tastenkürzel, Anzeigeformat, etc.
 - **Darstellung**: Enthält Angaben zur grafischen Darstellung der Objekte.
 - **Spezifisch**: Enthält Angaben zum gewählten Objekttyp, z.B. Tipps für Felder, animierte Bildschaltfläche, Druckeinstellungen für Formulare, etc. Im Gegensatz zu anderen Themenseiten erscheinen nur Eigenschaften, die für das Objekt relevant sind.

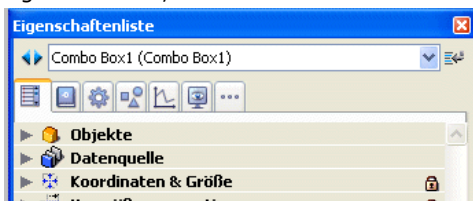
Tastenkürzel zum Navigieren

In der Eigenschaftenliste können Sie mit folgenden Tastaturkürzeln navigieren:

- **Pfeiltasten** ↑ ↓ : Damit springen Sie von Zelle zu Zelle.
- **Pfeiltasten** ← → : Damit klappen Sie Themen oder Menüs auf und zu.
- **Bild** ↑ und **Bild** ↓ : Damit wählen Sie die erste bzw. die letzte sichtbare Zelle der Eigenschaftenliste.
- **Pos1** und **Ende**: Damit wählen Sie die erste bzw. die letzte Zelle der Eigenschaftenliste.
- **Strg+Klick** (Windows) oder **Befehlstaste+Klick** (Mac OS) auf ein Ereignis: Damit aktivieren/deaktivieren Sie alle Ereignisse gemäß dem Anfangsstatus des angeklickten Ereignisses.
- **Strg+click** (Windows) oder **Befehlstaste+Klick** (Mac OS) auf ein Thema: Damit klappen Sie alle Themen auf/zu.

Icon Schloss

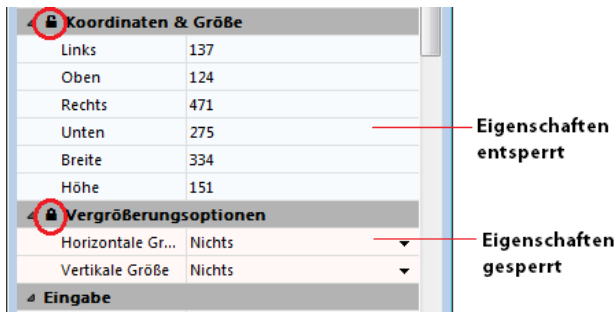
Ist die Eigenschaft **Vom Benutzer editierbar** für das Formular aktiv, geben Icons in Form eines Vorhängeschlosses bei einer Eigenschaft an, ob der Benutzer diese ändern kann oder nicht:



Weitere Informationen dazu finden Sie im Abschnitt **Vom Benutzer editierbar**.

Zusätzlich zum Schloss gibt die Hintergrundfarbe der Eigenschaften den jeweiligen Status an (nur 64-bit Versionen von 4D):

- blassrot: gesperrt
- blassblau: entsperrt
- weiss: nicht sperrbar



Lineale verwenden

Die Lineale des Formulareditors erstrecken sich über die Höhe und die Breite des Formulars.

Sie können die Lineale ausblenden, um Ihren Arbeitsbereich im Fenster des Formulareditors zu vergrößern. Sie können die Lineale bei Bedarf wieder einblenden. um die Lineale aus- oder einzublenden, wählen Sie im Menü **Formular** bzw. im Kontextmenü des Formulareditors das Untermenü **Anzeige** und dann den Eintrag **Lineale**.

Die Lineale enthalten Markierungen, die die Position des Mauszeigers beim Erzeugen oder Größe ändern eines Objektes anzeigen. Wenn Sie ein Objekt bewegen, ändern sich die Markierungen und zeigen die oberen und unteren bzw. linken und rechten Seiten des Objekts. Über die Objektmarkierungen können Sie andere Objekte an den gleichen Positionen auf den Linealen ausrichten.

Sie können die Einheiten der Lineale nach Ihren eigenen Wünschen ändern.

Um die Maßeinheit für Lineal zu definieren:

1. Wählen Sie im Menü **Formular** den Befehl **Lineal festlegen**.
Auf dem Bildschirm erscheint das Dialogfenster zum Festlegen der Einheiten.
2. Markieren Sie die gewünschte Einheit.
 - Wählen Sie **Punkte**, um Lineale mit der Maßeinheit Punkt anzuzeigen. Ein Punkt entspricht der Breite eines Pixel. Ein Zoll umfaßt 72 Punkt.
 - Markieren Sie **Zentimeter**, um metrische Lineale anzuzeigen.
 - Markieren Sie **Zoll**, um Lineale anzuzeigen, die Fuß und Zoll verwenden.
3. Klicken Sie auf die Schaltfläche **OK**.
4D passt die Maßeinheit an die gewählte Einheit an. Koordinaten und Größe der Objekte werden in der neuen Einheit angegeben.

Formulareigenschaften

Formulareigenschaften legen Sie in der Eigenschaftenliste des **Formulareditor** fest. Bestimmte Formulareigenschaften lassen sich im Fenster Formulareigenschaften des Explorer setzen. Weitere Informationen dazu finden Sie in den **Formulareigenschaften (Explorer)**.

Allgemeine Eigenschaften

Formulartyp

Sie können den Formulartyp ändern, z.B. sein Ziel. Für Projektformulare gibt es zwei Typen: **Detailformular** und **Detailformular für Drucken**. Für Tabellenformulare sind zwei weitere Typen verfügbar: **Listenformular** und **Listenformular für Drucken**.

Diese Eigenschaft bestimmt die Optionen, die in der Eigenschaftenliste für das Formular erscheinen.

Damit können Sie auch die Anzahl der Formulare, die in der aktuellen Auswahlliste für Eingabe- und Ausgabeformulare erscheinen, begrenzen (siehe **Zwischen Tabellen und Formularen wechseln**): Nur Formulare, deren Typ zur Liste passt, werden angezeigt.

Die Eigenschaft **Formulartyp** erscheint in der Eigenschaftenliste als erster Eintrag. Ist hier **Ohne** gewählt, erscheint der Formulartyp in beiden Menüs der Tabellenliste.

Hinweis: Sie können den Formulartyp auch im Fenster Formulareigenschaften des Explorer festlegen (siehe **Formulareigenschaften (Explorer)**).

Geerbte Formulare

Diese Eigenschaften werden im Abschnitt beschrieben.

Standardfenstertitel

Der Standardfenstertitel wird verwendet, wenn das Formular mit den 4D Funktionen **Open window** und **Open form window** in der Anwendungsumgebung geöffnet wird. Der Standardfenstertitel erscheint in der Titelleiste des Fensters. Um den Standardnamen zu definieren, tragen Sie im Eingabebereich **Fenstertitel** der Eigenschaftenliste einen Namen ein.

Sie können dynamische Referenzen in den Namen des Formularfensters einfügen, z.B.:

- Eine standardmäßige XLIFF Referenz (siehe **PICTURE TO BLOB**).
- Dynamischer Tabellen- oder Feldname mit der Syntax **<?[TabellenNr]FeldNr>** oder **<?[TabellenName]FeldName>**. Die Referenz wird aufgelöst, wenn der 4D Befehl **FORM SET INPUT** aufgerufen wird (wenn der Parameter * übergeben ist und der Aufruf von **Open window** folgt) und wenn der 4D Befehl **Open form window** aufgerufen wird.
- Variable oder Feld mit der Syntax **<VariableName>** oder **<[TabelleName]FeldName>**. Der aktuelle Wert der Variablen bzw. des Feldes erscheint im Fenstertitel.

Hinweise:

- Ein Fenstertitel kann max. 31 Zeichen lang sein.
- Sie können den Formularnamen auch im Fenster Formulareigenschaften im Explorer festlegen (siehe **Formulareigenschaften (Explorer)**).

Vom Benutzer editierbar

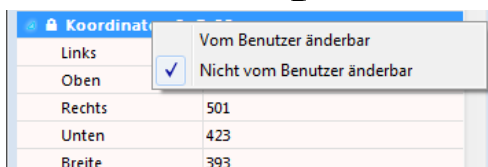
Mit der Eigenschaft **Vom Benutzer editierbar** legen Sie fest, dass der Benutzer das Formular in der Anwendungsumgebung im Benutzer-Formulareditor bearbeiten kann (siehe **Benutzerformulare**). Dafür sind vom Entwickler weitere Einstellungen notwendig, die im Handbuch *4D Programmiersprache* im Kapitel **Benutzerformulare** beschrieben werden.

Hinweis: Sie können die Eigenschaft **Vom Benutzer editierbar** auch im Fenster Formulareigenschaften im Explorer festlegen (siehe **Formulareigenschaften (Explorer)**).

Ist diese Eigenschaft aktiv, wird das Formular beim Sichern automatisch gesperrt. Öffnen Sie ein Formular mit dieser Eigenschaft, erscheint in der Werkzeugleiste das Icon für Gesperrt und es sind keine Änderungen möglich. Sie müssen auf dieses Icon klicken, um das Formular zu entsperren. In diesem Fall werden alle anderen benutzerdefinierten Formulare aufgehoben und müssen neu generiert werden.

Ist die Option **Vom Benutzer editierbar** markiert, enthält die Eigenschaftenliste auch Vorhängeschlösser, die die gesperrten Eigenschaften kennzeichnen.

Sie können auf diese Icons klicken, um sie zu entsperren. Dann werden die entsprechenden Eigenschaften im Benutzer-Formulareditor editierbar . Sie können auch das Kontextmenü in der Themenleiste verwenden (nur 64-bit Versionen):



Geometrie sichern

Ist die Option **Geometrie sichern** markiert und wird das Fenster über die Funktion **Open form window** mit dem Parameter * geöffnet, sichert 4D beim Schließen des Fensters automatisch verschiedene Parameter des Formulars, unabhängig, wie sie während der Arbeitssitzung verändert wurden:

- Die aktuelle Seite
- Position, Größe und Sichtbarkeit jedes Formularobjekts (inkl. Größe und Sichtbarkeit der Spalten von Listboxen)

Hinweis: Diese Option berücksichtigt keine Objekte, die über den Befehl **OBJECT DUPLICATE** erstellt wurden. Damit ein Benutzer die Umgebung solcher Objekte wiederherstellen kann, muss der Entwickler die Abfolge Erstellen, Definition und Positionierung dieser Objekte wiederholen.

Ist diese Option markiert, ist für einige Objekte auch die Option **Werte sichern** verfügbar. Weitere Informationen und Anwendungsbeispiele finden Sie im Abschnitt **Geometrie des Fensters beibehalten**.

Objekte nicht umkehren (Windows)

Die Option **Objekte nicht umkehren** deaktiviert den voreingestellten „rechts-nach-links“ Modus für das Formular. Ist dieser Modus aktiviert (nur Windows), werden Formularobjekte und Bezeichnungen umgekehrt. Das ist für Sprachen von Bedeutung, die von rechts nach links gelesen werden, z.B. Arabisch. Weitere Informationen dazu finden Sie im Abschnitt „Rechts-nach-links Sprachen“ auf der **Seite Oberfläche**.

Ist die Option **Objekte nicht umkehren** für ein Formular markiert, wird der „rechts-nach-links“ Modus nie für dieses Formular im Anwendungsmodus angewandt, unabhängig von der Konfiguration der Datenbank. Sie können diese Möglichkeit z.B. wählen, um „internationale“ Formulare innerhalb einer Datenbank einzurichten, die im Modus rechts-nach-links konfiguriert wurde.

Die Option **Objekte nicht umkehren** finden Sie in der Eigenschaftsliste und im Dialogfenster „Formulareigenschaften“ (siehe **Formulareigenschaften (Explorer)**).

Hinweis: Diese Eigenschaft lässt sich über den Befehl **FORM GET PARAMETER** auch per Programmierung steuern.

Größe von Formular und Fenster

Ein Formular wird immer in einem Fenster angezeigt. Mit 4D können Sie die Größe des Formulars und des Fensters definieren sowie ihr Verhalten beim Verändern der Größe.

Die Darstellung von Fenster und Formular richtet sich nach der Kombination dieser Eigenschaften. Beachten Sie, dass für das Ergebnis auch die Vergrößerungsoptionen, die jedem Formularobjekt zugewiesen wurden, eine Rolle spielen. Weitere Informationen dazu finden Sie im Abschnitt **Größe anpassen**).

Formulargröße

Die Eigenschaften für Formulargröße legen Sie in der Eigenschaftsliste unter der Gruppe „Formulargröße“ fest. Sie können wählen zwischen:

- **Größe basiert auf: Automatische Größe:** Die Größe des Formulars richtet sich nach der Anzeige aller Objekte plus der Werte, die Sie für **Hor. Rand** und **Vert. Rand** eingegeben haben (in Pixel).
- **Größe basiert auf: Setze Größe:** Größe des Formulars basiert auf den Werten in Pixel, die Sie in den Eingabebereichen **Breite** und **Höhe** festlegen (Wählen Sie **Setze Größe**, wechseln die Felder hor. und vert. Rand in Breite und Höhe).
- **Größe basiert auf: Formularobjekt:** Die Größe des Formulars richtet sich nach der Position des ausgewählten Formularobjekts. Wählen Sie z.B. ein Objekt an der rechten unteren Ecke, öffnet 4D ein Fenster, das gerade groß genug ist, um das Objekt einzuschließen.
Wählen Sie diese Option, wenn Sie aktive Objekte außerhalb des Bildschirms setzen wollen (z.B. außerhalb des Rechteckrahmens). Durch diese Option verändern solche Objekte nicht die Größe des Fensters.
- Wenn Sie die Option **Automatische Größe** oder **Größe basiert auf: Formularobjekt** wählen, gibt es die Bereiche **Hor. Rand** und **Vert. Rand**. Sie können dann zusätzlich zum Formullarrand in Pixel Werte für einen Rahmen eingeben. Diese Werte bestimmen auch den oberen und rechten Rand der Formulare im **Etiketteneditor**.
- **Hinweis:** Für Ausgabeformulare sind nur die Optionen **hor. Rand** oder **Breite** verfügbar.
- **Dynamische Anpassung:** Diese Eigenschaft ist unter der Gruppe „Formulargröße“ für Formulare verfügbar, die von einer früheren 4D Version konvertiert wurden, wenn die Eigenschaft **Auto. Größe** gewählt wurde. Damit können Sie den Modus Vergrößerungsoptionen verändern. In älteren 4D Versionen wurde die Größe eines Formulars mit der Eigenschaft **automatische Größe** nur beim Öffnen des Formulars berechnet. Gab es dann Änderungen über 4D Befehle wie **OBJECT MOVE**, wurde die Formulargröße nicht angepasst. Ab 4D 2004 wird die Formulargröße in solchen Fällen dynamisch angepasst. Dieses Prinzip gilt nur für die neuen Formulare und aus Kompatibilitätsgründen nicht standardmäßig für Formulare konvertierter Datenbanken. Damit auch diese standardmäßig angepasst werden, können Sie die Option **Dynamische Anpassung** wählen.

Fenstergröße

Ein Eingabeformular, das in einer eigenen Anwendung angezeigt wird, wird in der Regel über die 4D Funktion **Open window** oder **Open form window** geöffnet.

Über **Open window** können Sie die Werte der Koordinaten für oben, unten, links und rechts sowie den Fenstertyp festlegen. Dann hängt die Fenstergröße nicht von der Formulargröße ab. Die Vergrößerungsoptionen richten sich dagegen nach den Werten und nach dem Fenstertyp, die unter dem Thema „Formulargröße“ angegeben sind. **Open form window** erstellt ein neues Fenster gemäß den Parametern für Formulargröße und Vergrößerungsoptionen.

Sie können die Größe des Formularfensters in der Eigenschaftsliste verändern. Es gibt folgende Optionen:

- **Mit Einschränkung:** Diese Eigenschaft ist für Formulare verfügbar, die aus früheren Versionen von 4D konvertiert wurden (vor Version 2004). Sie reproduziert das Verhalten der bisherigen Eigenschaft „Anpassbar“. Ist die Option inaktiv, sind die Möglichkeiten zum Anpassen von Objekten und Beschränken der Fenstergröße außer Kraft gesetzt. Der Benutzer kann das Formularfenster frei anpassen. Die darin enthaltenen Objekte werden jedoch weder angepasst noch bewegt. Die Eigenschaften Minimum/Maximum und feste Größe sowie zum Anpassen der Objekte werden ignoriert. Das entspricht dem Verhalten früherer 4D Versionen und darf nur aus Kompatibilitätsgründen in spezifischen Oberflächen verwendet werden. Das Verhalten aktueller 4D Datenbanken entspricht der markierten Option (Standardmodus).
- **Feste Breite:** Damit sperren Sie die Fensterbreite, d.h. der Benutzer kann sie nicht anpassen. Ist die Option nicht markiert, lässt sich die Breite des Formularfensters verändern. Dann können Sie über die Bereiche **Minimale Breite** und **Maximale Breite** Werte für die Anpassung festlegen.

- **Feste Höhe:** Damit sperren Sie die Fensterhöhe, d.h. der Benutzer kann sie nicht anpassen. Ist die Option nicht markiert, lässt sich die Höhe des Formularfensters verändern. Dann können Sie über die Bereiche **Minimale Höhe** und **Maximale Höhe** Werte für die Anpassung festlegen.

Über feste Werte können Sie verhindern, dass der Benutzer das Fenster zu stark verkleinert bzw. vergrößert, so dass eingebare Bereiche und Steuerschaltflächen außerhalb des sichtbaren Bereichs liegen.

Darstellung

Plattform

Die Eigenschaft **Plattform** setzt das Aussehen eines Formular gemäß dem Kontext, in dem es ausgeführt wird. Diese Eigenschaft lässt sich auch individuell für jedes Formularobjekt setzen (siehe Abschnitt "Plattform" unter **Objekteigenschaften setzen**).

Es gibt zwei Eigenschaften für die Oberfläche: **System** und **Drucken**. Sie werden folgendermaßen verwendet: Beim Anzeigen auf dem Bildschirm muss das Formular die Oberfläche des aktuellen Betriebssystems berücksichtigen (**System**). Beim Drucken muss das Aussehen der Objekte angepasst werden (**Drucken**), unabhängig von der Plattform.

- **System:** Damit können Sie das Aussehen des Formulars oder Objekts automatisch an die aktuelle Plattform anpassen, auf der 4D läuft:
 - Wird das Formular auf OS X angezeigt, haben Formular bzw. Objekte Aussehen und Verhalten von OS X,
 - Wird das Formular unter Windows, angezeigt, haben Formular bzw. Objekte Aussehen und Verhalten der aktuellen Einstellung für "Aussehen" in der Steuerpalette.
Dieser Modus wird standardmäßig für alle Formulare und Objekte in Datenbanken ab 2004 und höher verwendet.
- **Drucken:** Damit können Sie das Aussehen des Formulars oder Objekts zum Drucken anpassen: Jedes Objekt oder grafische Element (Schaltfläche, Optionsfeld, Registerkarte, Linie, etc.) wird als Vektor gezeichnet, um ein zufriedenstellendes Ergebnis zum Drucken zu liefern. Es erscheint immer dasselbe Ergebnis, unabhängig auf welcher Plattform das Formular angezeigt oder gedruckt wird.

Metallisch (Mac OS)

Diese Formulareigenschaft aktiviert die metallische Darstellung des Formulars, wenn es auf Mac OS über die Funktion **Open form window** angezeigt wird. Diese Darstellungsart gilt für die OS X Oberfläche:



Erscheint das Formular in einem Fenster, das nicht mit **Open form window** erstellt wurde, z.B. im Designmodus, wird die Eigenschaft nicht berücksichtigt.

Auf Mac OS erscheint die metallische Darstellung in der Vorschau des Formulareditors, wenn in der Eigenschaftsliste die Option **Metallisch** markiert ist und die Formularbegrenzungen angezeigt werden. Wird das Formular unter Windows ausgeführt, hat diese Option keine Auswirkung.

Vergrößerungskästchen ausblenden

Markieren Sie diese Option, wird das Vergrößerungskästchen ausgeblendet. Diese Option wird berücksichtigt, wenn Sie das Formular z.B. über den 4D Befehl **DIALOG** aufrufen.

Drucken

Die Beschreibung der Druckeigenschaften finden Sie im Abschnitt **Formular drucken**.

Menüleiste einem Formular zuordnen

Wenn Sie eine Anwendung anlegen, erzeugen Sie auch eigene Menüs. In eigenen Menüs können Sie Menübefehle hinzufügen, um Aktionen zu automatisieren, die mit einem einzelnen Formular verknüpft sind, wie z.B. einen Bericht aus Daten generieren.

Eigene Menüs erstellen Sie im Menüleisteneditor. Jede Menüleiste enthält mindestens ein Menü und hat eine einmalige Nummer und einen Namen. Weitere Informationen dazu finden Sie im Kapitel **Menüs und Menüleisten**.

Um einem Formular eine Menüleiste zuzuordnen, wählen Sie in der Eigenschaftsliste eine Menüleiste in der DropDown Liste "Zugeordnete Menüleiste". Über die Schaltfläche [...] können Sie direkt auf den Menüleisteneditor zugreifen (siehe **Menüs und Menüleisten**). In der Anwendungsumgebung wird eine Menüleiste, die einem Formular zugewiesen ist, rechts an die aktuelle Menüleiste hinzugefügt.

Ist die Menüleiste identisch mit der aktuellen Menüleiste, wird sie nicht angehängt. Eine Formularmenüleiste funktioniert in Eingabe- und Ausgabeformularen.

- Die Option **Menüleiste aktivieren** erscheint, wenn Sie eine Menüleiste auswählen. Wollen Sie dieses Formular in einer eigenen Anwendung einsetzen, markieren Sie diese Option. Damit weisen Sie 4D an, die aktuelle Menüleiste aktiv zu lassen. Ist diese Option nicht markiert, deaktiviert 4D die aktuelle Menüleiste und erlaubt nur den Zugang auf die dem Formular zugewiesene Menüleiste.

Hinweis: Sie können eine Menüleiste einem Formular auch über das Fenster **Formulareigenschaften (Explorer)** zuweisen.

Formularzugriff steuern

Sie können den Zugriff auf ein Formular steuern, indem Sie Zugriffs- und Eigentümerrechte für Benutzergruppen festlegen. Für jede Gruppe lassen sich unter dem Thema "Zugriff" Zugriff und Eigentümer festlegen. Weitere Informationen zum Erzeugen eines Kennwortsystems mit Benutzern und Gruppen finden Sie im Kapitel **Benutzer und Gruppen**.

Die Dropdown-Liste *Zugriff* legt fest, welche Gruppe das Formular im Anzeigefenster für Datensätze oder in eigenen Anwendungen verwenden kann. Wenn ein Benutzer, der nicht zu dieser Gruppe gehört, versucht, das Formular zu verwenden, zeigt 4D eine Meldung, dass er dazu nicht berechtigt ist.

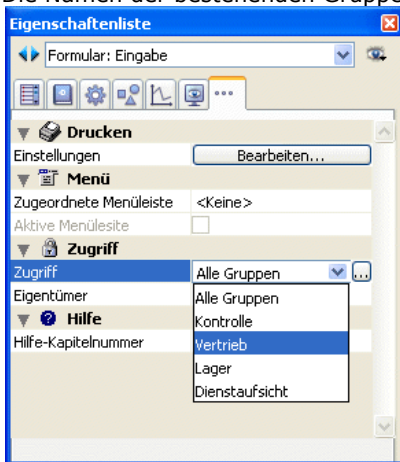
Die Dropdown-Liste *Eigentümer* legt fest, welche Gruppe das Formular in der Designumgebung bearbeiten kann. Wenn ein Benutzer, der nicht zu dieser Gruppe gehört, versucht, das Formular in der Designumgebung zu öffnen, zeigt 4D eine Meldung, dass der Benutzer nicht berechtigt ist, das Formular zu bearbeiten.

Benutzer, die beiden Gruppen angehören, können das Formular in der Designumgebung und in eigenen Anwendungen verwenden.

Um Zugriffsrechte für ein Formular zu setzen:

1. Zeigen Sie die **Formulareigenschaften** in der Eigenschaftenliste an
2. Verwenden Sie zum Zuweisen der gewünschten Zugriffsrechte die Dropdown-Listen *Zugriff* und *Eigentümer*.

Die Namen der bestehenden Gruppen erscheinen in der jeweiligen Dropdown-Liste.



Über die Schaltfläche [...] gelangen Sie direkt in den Kennworteditor in der Toolbox von 4D.

Sie können den Zugriff auf ein Formular auch im Fenster **Formulareigenschaften (Explorer)** definieren.

Hilfe

4D ermöglicht, jeder Anwendung eine eigene Online Hilfe zuzuordnen. Wie Sie Hilfedateien erstellen, erfahren Sie unter **Anhang A: Eigene Online-Hilfe zuordnen**.

Hilfedateien können kontextuell sein, d.h. sie können Informationen passend zum Kontext zeigen, in dem sie aufgerufen werden. Dazu können Sie einem Formular eine **Hilfe-Kapitelnummer** zuweisen und dieselbe Nummer dem Abschnitt oder der entsprechenden Seite der Hilfedatei. Das führen Sie in der Eigenschaftenliste unter dem Thema „Hilfe“ aus.

Aktion

In diesem Bereich können Sie über die Schaltfläche **Bearbeiten** auf die Formalmethode zugreifen. Weitere Informationen dazu finden Sie unter **Methoden bearbeiten**.

Markierungen

Dieser Bereich bestimmt die genaue Position der Markierungen im vertikalen Lineal des Formulars. Markierungen werden hauptsächlich in Ausgabeformularen verwendet. Sie steuern die Information, die in den Bereichen Kopfteil, Umbruch, Detail und Fußteil gelistet und gesetzt werden. Weitere Informationen dazu finden Sie im Abschnitt **Steuerlinien für Ausgabe verwenden**.

Hinweis: Die Markierung für Breite auf dem horizontalen Lineal steuert die Breite einer Bezeichnung, wenn Sie ein Formular zum Drucken von Adresstiketten über den 4D Befehl **PRINT LABEL** erstellen.

Ereignisse

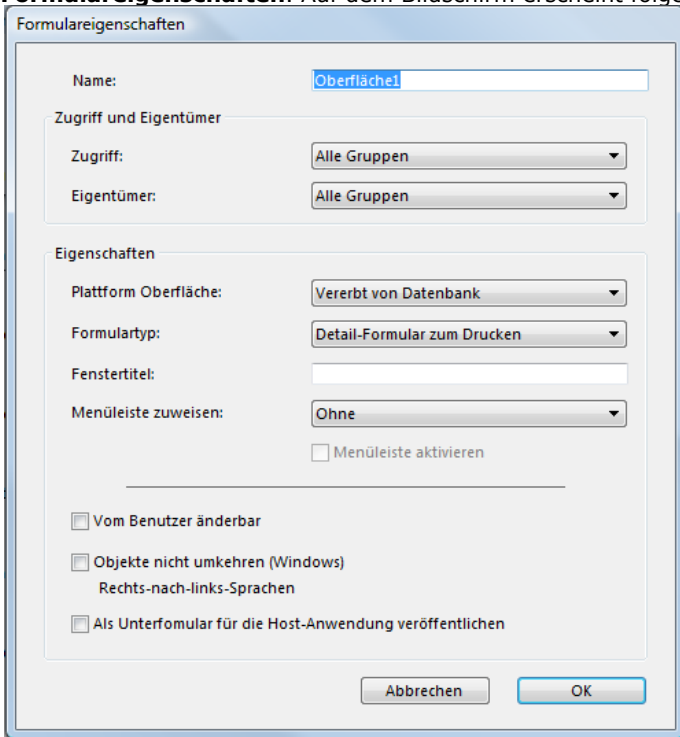
Dieser Bereich setzt die Ereignisse, die zur Ausführung der Formalmethode führen können. Wenn das Formular verwendet wird, treten nur die Ereignisse ein, die Sie ausgewählt haben. Wählen Sie keine Ereignisse aus, wird die Formalmethode nie aufgerufen.

Ihre Anwendung läuft schneller, wenn Sie unnötige Ereignisse deaktivieren.
Weitere Informationen zu Formularereignissen finden Sie unter **Form event**.

Wollen Sie alle Ereignisse aktivieren bzw. deaktivieren, klicken Sie unter Windows mit gedrückter **Strg-Taste**, auf Macintosh mit der **Befehlstaste** auf ein Ereignis.

Formulareigenschaften (Explorer)

Sie können auf die wichtigsten Formulareigenschaften im Explorer auf der [Seite Formulare](#) zugreifen. Klicken Sie dazu auf den Namen des gewünschten Formulars und wählen im Kontextmenü oder im Menü **Optionen** am unteren Rand den Eintrag **Formulareigenschaften**. Auf dem Bildschirm erscheint folgendes Dialogfenster:



Formulareigenschaften

Name:

Zugriff und Eigentümer

Zugriff:

Eigentümer:

Eigenschaften

Plattform Oberfläche:

Formulartyp:

Fenstertitel:

Menüleiste zuweisen:

Menüleiste aktivieren

Vom Benutzer änderbar

Objekte nicht umkehren (Windows)
Rechts-nach-links-Sprachen

Als Unterformular für die Host-Anwendung veröffentlichen

Die meisten der Eigenschaften in diesem Dialogfenster, wie Name, Zugriff, Plattform, lassen sich auch in der Eigenschaftenliste des Formulareditors setzen. Weitere Informationen dazu finden Sie im Abschnitt [Formulareigenschaften](#). Die Eigenschaft "Als Unterformular für die Host-Anwendung veröffentlichen" ist dagegen nur in diesem Dialogfenster verfügbar.

Als Unterformular für die Host-Anwendung veröffentlichen

Diese Option ist sinnvoll beim Entwickeln von Komponenten, die Formulare enthalten (siehe [Formulare gemeinsam nutzen](#)).

Damit ein Komponentenformular als Unterformular in einer Host-Anwendung ausgewählt wird, muss es explizit als "veröffentlichtes Formular" deklariert sein. Markieren Sie dazu im Dialogfenster Formulareigenschaften die Option **Als Unterformular für die Host-Anwendung veröffentlichen**. Nur Projekt Unterformulare lassen sich als veröffentlichte Unterformulare definieren.

Weitere Informationen zum Anlegen und Einsetzen von Komponenten Unterformularen finden Sie im Abschnitt [Komponente entwickeln](#).

Mehrseitiges Formular erstellen

Sie können für ein Eingabeformular mehrere Seiten erzeugen. Wenn Sie mehr Datenfelder haben, als auf einen Bildschirm passen, können Sie weitere Seiten erzeugen, um die Datenfelder vollständig anzuzeigen. Mit mehreren Seiten können Sie Folgendes ausführen:

- Die wichtigsten Informationen auf der ersten Seite und weniger wichtige Informationen auf die anderen Seiten setzen
- Jedes Thema auf eine eigene Seite setzen
- Blättern bei der Dateneingabe einschränken oder verhindern
- Platz um Formularelemente herum schaffen, um eine ansprechende Darstellung auf dem Bildschirm zu erreichen

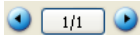
Mehrere Seiten lassen sich nur für Eingabeformulare verwenden. Sie sind nicht zur Druckausgabe gedacht. Beim Drucken eines mehrseitigen Formulars wird nur die erste Seite ausgedruckt.

Die Seitenzahl ist bei Formularen nicht beschränkt. Das gleiche Datenfeld kann beliebig oft in einem Formular und auf beliebig vielen Seiten vorkommen. Aber je mehr Seiten ein Formular umfasst, desto länger dauert das Öffnen des Formulars. Ein mehrseitiges Formular hat eine Hintergrundseite (Seite 0) und mehrere Anzeigeseiten. Objekte, die auf der Hintergrundseite liegen, sind auf allen Anzeigeseiten sichtbar, können aber nur auf der Hintergrundseite ausgewählt und bearbeitet werden. Sie sollten in mehrseitigen Formularen Ihre Schaltflächenpalette auf die Hintergrundseite legen. Hier müssen Sie auch ein oder mehrere Objekte setzen, damit der Benutzer von Seite zu Seite wechseln kann. Weitere Informationen dazu finden Sie im Abschnitt [Optionen für Seitenwechsel hinzufügen](#).


Hinweis: Die Seite Optionen im Formularassistenten bietet eine Option, um automatisch ein mehrseitiges Formular zu erzeugen, wenn die gewählten Datenfelder nicht mehr auf eine Seite passen. Damit kann das Formular schon von Anfang an mehr als eine Anzeigeseite haben. In diesem Fall setzt der Formularassistent Schaltflächen, Variablen, den Formulartitel und Schmuckrahmen auf die Hintergrundseite 0.

Anzeigeseite im Formular hinzufügen

Jedes Formular hat mindestens eine Anzeigeseite 1 und eine Hintergrundseite 0. Die aktuelle Seitennummer erscheint in der Werkzeugleiste des Formularfensters. Über die Pfeilsymbole können Sie zwischen den Seiten des Formulars navigieren.



Um eine Anzeigeseite hinzuzufügen:

1. Wählen Sie im Menü **Formular** den Befehl **Seite hinzufügen**. 4D erstellt eine neue Seite.
ODER
Wechseln Sie auf die letzte Seite im Formular. Klicken Sie dann in der Werkzeugleiste auf das Symbol Nächste Seite .
4D öffnet ein Dialogfenster mit der Frage, ob Sie eine Seite hinzufügen wollen. Klicken Sie auf die Schaltfläche **OK**, um eine neue Seite anzulegen.
Eine neue leere Seite erscheint im Fenster des Formulareditors. Der Seitenbereich in der Werkzeugleiste zeigt die Nummer der gerade angezeigten Seite. Sie können jetzt Datenfelder und andere Formularelemente in die neue Seite einfügen.


Um eine Anzeigeseite vor der aktuellen Seite einzufügen:

1. Wählen Sie im Menü **Formular** den Befehl **Seite einfügen**. 4D fügt vor der aktuellen Seite eine neue Seite ein und zeigt sie auf dem Bildschirm an.

Von Seite zu Seite wechseln


Wenn Sie die Hintergrundseite anzeigen oder zu einer anderen Anzeigeseite wechseln wollen, können Sie das DropDown-Menü für Seite oder die Pfeilsymbole in der Werkzeugleiste, bzw. die Befehle zum Navigieren im Menü **Formular** verwenden.

Um die Hintergrundseite (Seite 0) anzuzeigen:

1. Wechseln Sie zur ersten Seite im Formular und klicken in der Werkzeugleiste auf das Symbol Vorherige Seite .
ODER
Blättern Sie im DropDown-Menü der Werkzeugleiste zur Seite 0.
ODER
Wählen Sie im Menü **Formular** > **Gehe zu Seite** und dann die Seite 0
ODER
Klicken Sie unter Windows mit gedrückter **Alt-Taste**, auf Mac OS mit gedrückter **Wahltaste** auf ein Objekt, das zu Seite 0 gehört oder auf der aktuellen Seite komplett außerhalb eines Objekts liegt. (die Tastenkombination **Alt+Klick** oder **Wahltaste+Klick** auf ein Objekt der aktuellen Seite erstellt bzw. öffnet die dazugehörige Objektmethode.)
4D zeigt die Hintergrundseite. Die Seitenzahl der Hintergrundseite ist Null (0). Objekte auf der Hintergrundseite erscheinen auf allen Seiten. Sie können jeden Objekttyp auf die Hintergrundseite legen.

Hinweis: Verwechseln Sie nicht die Anzeige der Seite 0 als aktuelle Seite mit der Option **Seite 0** des Untermenüs **Anzeige**. Diese blendet in der aktuellen Seite die Objekte auf der Seite 0 ein bzw. aus. Weitere Informationen dazu finden Sie im Abschnitt "Elemente im Formulareditor ein/ausblenden" unter [Formulareditor](#).

Um in den Seiten zu navigieren:

1. Um die nächste Seite anzuzeigen, klicken Sie in der Werkzeugleiste auf den Pfeil für Nächste Seite . 4D zeigt die sofort die entsprechende Seite.

Klicken Sie auf Vorige Seite, wenn die erste Seite des Formulars angezeigt wird, erscheint die Hintergrundseite. Klicken Sie auf Nächste Seite, wenn die letzte Seite des Formulars angezeigt wird, fragt 4D, ob Sie eine weitere Seite im Formular anlegen wollen.

Um eine beliebige Seite anzuzeigen:

1. Klicken Sie in der Werkzeugleiste des Formulars mit gedrückter Maustaste auf die Seitenzahl, um das zugeordnete PopUp-Menü anzuzeigen.
ODER
Öffnen Sie im Menü **Formular>Gehe zu Seite**.
2. Wählen Sie die gewünschte Seitenzahl

Seite löschen

Sie können unerwünschte Anzeigeseiten aus einem mehrseitigen Formular löschen. Alle Datenfelder oder andere Objekte auf den gelöschten Seiten werden ebenfalls gelöscht und die verbleibenden Seiten werden neu nummeriert. Die erste Seite und die Hintergrundseite können Sie nicht löschen.

Um eine Seite im Formular zu löschen:

1. Verwenden Sie die Werkzeuge zum Navigieren oder das DropDown-Menü für Seiten in der Werkzeugleiste, um die gewünschte Seite anzuzeigen.
2. Wählen Sie im Menü **Formular** den Befehl **Seite löschen**.
Auf dem Bildschirm erscheint ein Dialogfenster zum Bestätigen.
3. Klicken Sie auf die Schaltfläche **OK**.
4D entfernt die Seite und alle Objekte auf der Seite aus dem Formular.

Optionen für Seitenwechsel hinzufügen

Wenn Sie ein mehrseitiges Formular erzeugen, müssen Sie dafür sorgen, dass der Benutzer von Seite zu Seite wechseln kann. 4D bietet dafür folgende Möglichkeiten:

- **Registerkarte:** Damit können Sie direkt auf einzelne Seiten zugreifen. Sie setzen die Registerkarte auf die Hintergrundseite des Formulars und verwenden deren Eigenschaften zum Steuern des Seitenwechsels.
- **Automatische Schaltflächen:** Sie können dem Formular automatische Schaltflächen zum Seitenwechsel hinzufügen - Erste Seite, Letzte Seite, Vorige Seite und Nächste Seite. Diese Schaltflächen sollten auf der Hintergrundseite liegen.
- **Objektmethoden:** Darüberhinaus bietet die Programmiersprache den 4D Befehl **FORM GOTO PAGE**. Sie können ihn in einer Objektmethode verwenden, um mit beliebigen Objekttypen eigene Schaltflächen zum Seitenwechsel zu erzeugen. Sie können dafür zum Beispiel eine Bildschaltfläche oder ein PopUp-Menü verwenden.

Schaltflächen zum Navigieren hinzufügen

Sie können Schaltflächen zum Seitenwechsel hinzufügen, wenn Sie beim Erstellen des Formulars die Seite Schaltflächen im Formularassistenten verwenden. Ist das Formular angelegt, öffnen Sie es im Formulareditor und fügen die erforderlichen Seiten hinzu. Wollen Sie die Schaltflächen zum Seitenwechsel erst nach Erzeugen des Formulars hinzufügen, wählen Sie dazu das Symbol Aktives Objekt in der Objektleiste oder das Symbol Schaltfläche in der Werkzeugleiste. Weitere Informationen finden Sie im Abschnitt **Schaltflächen**.

Registerkarte hinzufügen

Die Registerkarte bietet eine visuelle Anzeige der aktuellen Seite und weiterer Seiten. Weitere Informationen zum Erstellen und Aktivieren einer Registerkarte finden Sie im Abschnitt **Registerkarten**.

Vererbte Formulare verwenden

4D ermöglicht, „vererbte Formulare“ zu verwenden. Das bedeutet, dass Sie alle Objekte von Formular A in einem ein Formular B verwenden können. Formular B „erbt“ sozusagen die Objekte von Formular A.

Nehmen wir beispielsweise an, dass alle Eingabeformulare der Datenbank das Firmenlogo und die Schaltflächen OK, Abbrechen, Nächstes und Voriges enthalten sollen. Hierfür erstellen Sie ein Formular mit nur diesen Elementen und rufen es in allen Eingabeformularen als vererbtes Formular auf. Jedes Eingabeformular enthält nun diese gemeinsamen Elemente und zusätzlich eigene Felder und Objekte.

Im Unterschied zu Formularvorlagen, die über den Formularassistenten definiert werden (Näheres im Abschnitt **Formularvorlage erstellen**), ist die Referenz auf ein geerbtes Formular immer aktiv: Ändern Sie z.B. den Schriftstil einer Schaltfläche im geerbten Formular, wirkt sich diese Änderung automatisch auf alle Formulare aus, die das geerbte Formular verwenden.

Sowohl Tabellen- als auch Projektformulare können vererbte Formulare verwenden bzw. als solche verwendet werden.

Funktionsweise

Arbeiten Sie mit der Datenbank, werden die Objekte des geerbten Formulars dynamisch mit denen des geöffneten Formulars kombiniert. Die Funktionsweise ist ähnlich wie für die „Seite Null“ der Formulare, mit dem Unterschied, dass sich ein geerbtes Formular als Ganzes auf alle Formulare der Datenbank anwenden lässt.

Wird ein Formular ausgeführt, werden die Objekte folgendermaßen geladen und miteinander kombiniert:

1. Seite Null des geerbten Formulars
2. Seite 1 des geerbten Formulars
3. Seite Null des geöffneten Formulars
4. Aktuelle Seite des geöffneten Formulars

Diese Reihenfolge bestimmt auch die Eingabereihenfolge der Objekte im Formular.

Hinweis: Nur die Seiten 0 und 1 des geerbten Formulars können in den anderen Formularen erscheinen.

Die Eigenschaften (Name des Fensters, Anpassung der Größe, Ereignisse, etc.) sowie die Methode eines geerbten Formulars werden beim Vererben nicht berücksichtigt. Die den Objekten zugeordneten Methoden werden dagegen aufgerufen.

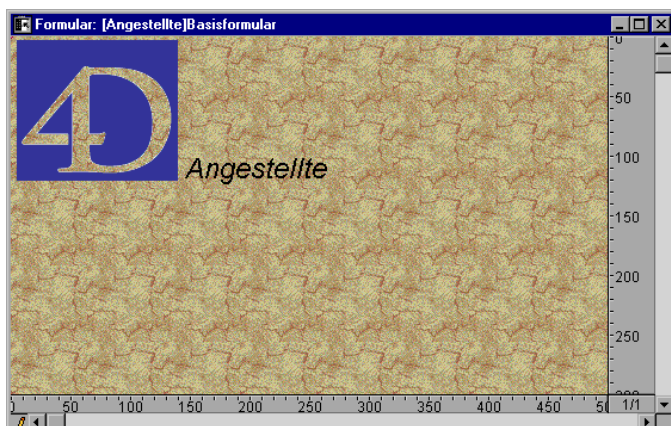
Geerbtes Formular definieren

Die Vererbung eines Formulars wird im 4D Formulareditor eingerichtet.

Um ein geerbtes Formular zu definieren:

1. Öffnen Sie im Formulareditor das Formular, welches von einem anderen Formular erben soll.
2. Öffnen Sie die Eigenschaftenliste und klicken Sie außerhalb eines Formularobjekts, damit die Formulareigenschaften angezeigt werden.
Hier gibt es die Eigenschaften „Tabelle des geerbten Formulars“ und „Name des geerbten Formulars“. Alle Tabellen und die dazugehörigen Formulare werden angezeigt.
3. Wählen Sie die Tabelle und dann den Namen des Formulars, von dem geerbt werden soll. Um von einem Projektformular zu vererben, wählen Sie unter „Tabelle des geerbten Formulars“ <Keine>.

Sie können jedes Formular als geerbtes Formular definieren. Sie müssen aber darauf achten, dass die enthaltenen Elemente in den verschiedenen Tabellen der Datenbank verwendbar ist.



Wählen Sie ein geerbtes Formular aus, erscheint sein Inhalt im aktuellen Fenster. Dies ist nur eine Vorschau, Sie können hier ein Formularobjekt weder auswählen noch ändern. Dazu müssen Sie das Formular in seinem eigenen Fenster öffnen.

Wollen Sie die Objekte eines geerbten Formulars ausblenden, wählen Sie im Menü **Formular** im Untermenü **Anzeige** bzw. im Kontextmenü des Editors erneut die Option **Vererbtes Formular**.

Wollen Sie die Vererbung des Formulars wieder aufheben, wählen Sie in der Eigenschaftenliste die Option **<Keine>**.

Hinweis: Sie können auch ein geerbtes Formular in einem Formular definieren, das selbst als geerbtes Formular für ein drittes Formular dient. Die Kombination der Objekte erfolgt rekursiv. 4D findet in jedem Fall die Rekursionen und unterbricht die Verkettung der Formulare. Das ist z.B. der Fall, wenn das Formular [Tabelle1]Formular1 als geerbtes Formular von [Tabelle1]Formular1, d.h. in sich selbst definiert ist.

Eingabereihenfolge der Daten ändern

Die Eingabefolge ist die Reihenfolge, in der Datenfelder, Unterformulare und andere aktive Objekte ausgewählt werden, wenn Sie mit der **Tabulatortaste** oder der **Zeilenschaltung** durch ein Eingabeformular springen. Sie können sich auch in der Gegenrichtung durch das Formular bewegen, wenn Sie die Tastenkombination **Shift+Tab** oder **Shift+Zeilenschaltung** verwenden.

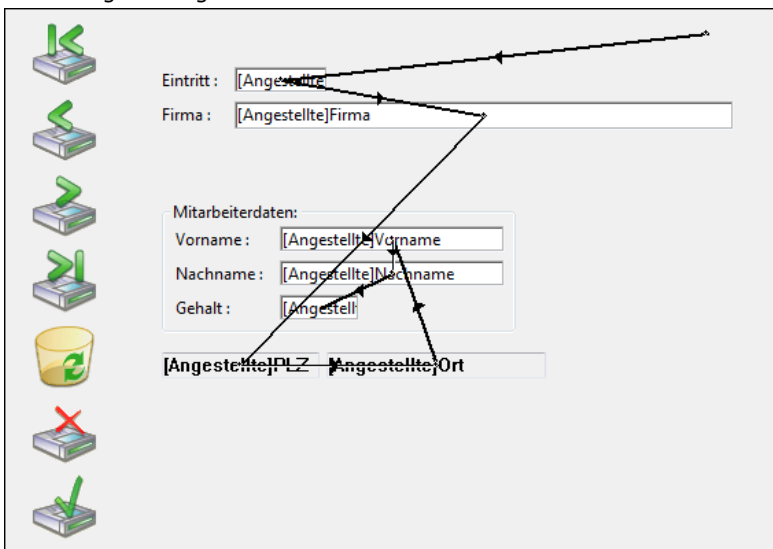
Jeder Bereich, in den Sie Werte eingeben können, ist in die Eingabefolge einbezogen. Boolesche Datenfelder (angezeigt als Optionsfelder oder Kontrollkästchen), Unterformulare, Combo Boxen und Datenfelder für Bilder sind ebenfalls in der Eingabefolge enthalten.

Auch in Thermometer, Lineale und Halbkreissskalen lassen sich Daten eingeben, sie werden jedoch nicht von der Eingabefolge berücksichtigt. Diese Objekte müssen Sie durch Anklicken auswählen.

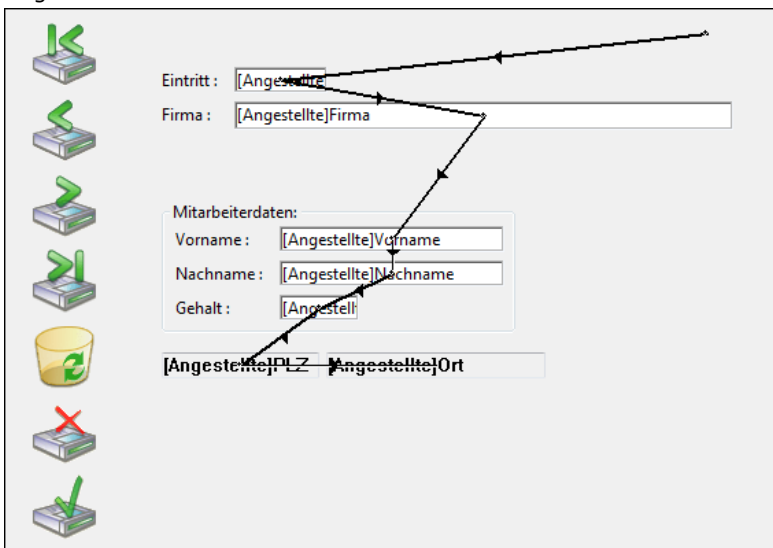
Geben Sie keine eigene Eingabefolge an, verwendet 4D standardmäßig die Anordnung der Objekte ausgehend vom Hintergrund zum Vordergrund. Die Standardeingabefolge entspricht also der Reihenfolge, in der die Objekte im Formular erstellt wurden.

In einigen Formularen wird eine eigene Eingabefolge benötigt. Zum Beispiel zeigt die folgende Darstellung Datenfelder, die nach dem Erstellen des Formulars hinzugefügt wurden.

Das beeinträchtigt die Eingabereihenfolge und zwingt den Benutzer dazu, die Informationen in einer eher unlogischen Reihenfolge einzugeben:



In solchen Fällen können Sie die Eingabefolge anpassen, so dass sich die Daten wieder in einer logischeren Reihenfolge eingeben lassen.



Eingabefolge anzeigen und ändern

Sie können die aktuelle Eingabefolge über das Erkennungszeichen oder den Modus Eingabefolge aufrufen. Veränderungen sind jedoch nur im Modus Eingabefolge möglich.

Dieser Abschnitt beschreibt den Modus Eingabefolge. Weitere Informationen zu Erkennungszeichen finden Sie unter [Erkennungszeichen verwenden](#).

Um die Eingabefolge anzusehen oder zu verändern:

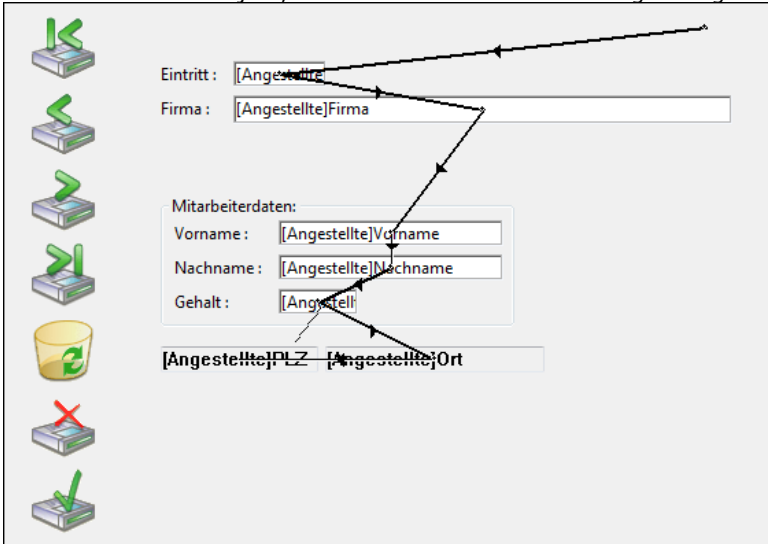
1. Wählen Sie im Menü **Formular** den Befehl **Eingabefolge** oder in der Werkzeugleiste des Formulareditors das Symbol



Der Mauszeiger wird zu einem Eingabefolgezeiger. 4D zeichnet im Formular eine Linie mit Pfeilen, die die aktuelle Reihenfolge zur Eingabe der Daten anzeigt.

Die Eingabefolge bleibt aktiv und ist änderbar, bis Sie ein anderes Tool in der Werkzeugleiste wählen.

2. Um die Eingabefolge zu ändern, klicken Sie mit dem Zeiger auf ein Objekt im Formular und ziehen bei gedrückter Maustaste zu dem Objekt, das Sie als nächstes in der Eingabefolge haben wollen.



4D passt die Eingabefolge entsprechend an.


3. Wiederholen Sie die Aktion, bis die neue Eingabefolge Ihren Wünschen entspricht.
4. Ist die Eingabefolge abgeschlossen, klicken Sie in der Werkzeugleiste auf ein anderes Symbol oder wählen Sie im Menü **Formular** den Eintrag **Eingabefolge** aus.
4D kehrt zur normalen Anzeige des Formulareditors zurück.

Hinweis: Nur die Eingabereihenfolge der aktuellen Seite des Formulars wird angezeigt. Enthält das Formular eingebbare Objekte von Seite 0 oder von einem geerbten Formular, gilt standardmäßig folgende Reihenfolge: Objekte von Seite 0 des geerbten Formulars -> Objekte von Seite 1 des geerbten Formulars -> Objekte von Seite 0 des offenen Formulars -> Objekte von der aktuellen Seite des offenen Formulars.

Erstes Objekt in der Eingabefolge festlegen

Alle eingebbaren Objekte sind Teil der Eingabefolge. Um das erste Objekt zu setzen, müssen Sie seine Position innerhalb der Formularebenen verändern. Dabei muss der Modus Eingabefolge inaktiv sein.


Um ein Objekt an den Anfang der Eingabefolge zu stellen:

1. Markieren Sie das Objekt, welches das erste in der Eingabefolge sein soll.
2. Wählen Sie im Menü **Objekt** den Befehl **Nach hinten bringen**
ODER
Wählen Sie im Kontextmenü des Objekts **Position>Nach hinten bringen**.
3. Wählen Sie im Menü **Formular** den Eintrag **Eingabefolge** oder klicken Sie in der Werkzeugleiste auf das Symbol .
Das ausgewählte Objekt wird erstes Objekt in der Eingabefolge, das bisherige erste Objekt wird zweites. Sie können jetzt bei gedrückter Maustaste von Objekt zu Objekt ziehen, um die gewünschte Eingabefolge herzustellen.
4. Ist die gewünschte Reihenfolge hergestellt, wählen Sie im Menü **Formular** erneut den Befehl **Eingabefolge** oder in der Werkzeugleiste ein anderes Symbol.
Der Formulareditor kehrt zur normalen Anzeige zurück.

Eingabegruppe verwenden

Beim Ändern der Eingabefolge können Sie im Formular eine Objektgruppe auswählen, so dass sich die Standardeingabefolge auf die Objekte in der Gruppe auswirkt. So können Sie problemlos die Eingabefolge in Formularen festlegen, in denen Datenfelder gruppen- oder spaltenweise angeordnet sind.

Um eine Gruppe zur Dateneingabe festzulegen:

1. Wählen Sie im Menü **Formular** den Eintrag **Eingabefolge** oder klicken Sie in der Werkzeugleiste auf das Symbol .
2. Ziehen Sie einen gestrichelten Rahmen um die Objekte, die Sie für die Eingabe gruppieren wollen.

Wenn Sie die Maustaste loslassen, folgen die Objekte, die innerhalb des Rahmens liegen oder von ihm berührt werden, der Standardeingabefolge. Die Eingabefolge für die verbleibenden Objekte wird entsprechend angepasst.

Datenfeld ausschließen

Standardmäßig sind in der Eingabefolge alle Datenfelder enthalten. Wollen Sie ein Datenfeld unter Beibehaltung der Eigenschaft „Eingebbar“ davon ausschließen, müssen Sie nur die Eigenschaft „Tabfähig“ deaktivieren. Weitere Informationen dazu finden Sie im Abschnitt **Tabfähig**.

Formularobjekte einfügen und organisieren

Objekte hinzufügen

Sie können Objekte auf verschiedene Arten in Formularen hinzufügen:

- Sie zeichnen das Objekt direkt im Formular, nachdem Sie den gewünschten Typ in der Objektleiste ausgewählt haben (siehe **Objektleiste verwenden**)
- Sie setzen das Objekt per Drag&Drop aus der Objektleiste ein
- Sie wählen ein Objekt per Drag&Drop oder Kopieren/Einsetzen aus der vorkonfigurierten Objektbibliothek (siehe **Vorkonfigurierte Objektbibliothek verwenden**) oder einer eigenen Bibliothek (siehe **Eigene Objektbibliothek anlegen und einsetzen**)
- Sie setzen das Objekt per Drag&Drop aus einem anderen Formular ein
- Sie setzen das Objekt per Drag&Drop aus dem Explorer (Datenfelder) oder anderen Editoren in der Designumgebung (Listen, Bilder, etc.) ein

Ist das Objekt im Formular gesetzt, können Sie seine Eigenschaften über den Formulareditor ändern.


Sie können in Ihren Formularen mit zwei Objekttypen arbeiten:

- **Statische Objekte** (Linien, Rahmen, Hintergrundbilder, etc.): Diese Objekte dienen in der Regel zum Gestalten des Formulars und seiner Bezeichnungen, sowie für die grafische Oberfläche. Sie sind in der Objektleiste des Formulareditors verfügbar. Über die Eigenschaftsliste können Sie auch Attribute wie Größe, Farbe, Schrift und Größenanpassungen festlegen. Im Gegensatz zu aktiven Objekten sind statischen Objekten keine Variablen zugeordnet. Sie können jedoch in statischen Objekten dynamische Objekte einfügen (siehe **Referenzen in statischem Text verwenden**).
- **Aktive Objekte**: Diese Objekte führen Aufgaben oder Funktionen auf der Oberfläche aus und können viele Formen haben: Felder, Schaltflächen, scrollbare Listen, etc. Jedes aktive Objekt ist entweder einem Feld oder einer Variable zugeordnet. Weitere Informationen dazu finden Sie im Kapitel **Mit aktiven Objekten arbeiten**.

Objekte auswählen

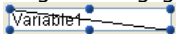
Bevor Sie eine Änderung an einem Objekt vornehmen (z.B. Linienstärke oder Schrift ändern), müssen Sie das betreffende Objekt auswählen.

Um ein Objekt mit der Werkzeugleiste auszuwählen:

1. Klicken Sie in der Werkzeugleiste auf das Symbol 

Ziehen Sie dieses Symbol mit der Maus in den Formularbereich, wird der Mauszeiger zu einem Pfeil auf weißem Hintergrund.

2. Klicken Sie auf das Objekt, das Sie auswählen wollen
Vergrößerungsgläser markieren das ausgewählte Objekt.




Um ein Objekt über die Eigenschaftsliste auszuwählen:

1. Wählen Sie in der Auswahlliste im oberen Bereich den Objektnamen aus.
Auf diese Weise können Sie ein Objekt auswählen, das durch andere Objekte verdeckt wird.

Wollen Sie ein Objekt wieder abwählen, klicken Sie außerhalb dieses Objekts oder mit gedrückter **Umschalttaste** auf das Objekt.

Hinweis: Sie können Objekte auch durch Doppelklick im Fenster auswählen, welches das Ergebnis einer Suche in der Datenbank angibt. Weitere Informationen dazu finden Sie im Abschnitt **Umbenennen**.

Wenn Sie Schwierigkeiten beim Auswählen eines Objekts haben, das durch den Formularassistenten erzeugt wurde, wechseln Sie über die Schaltfläche  zur Hintergrundseite oder wählen Sie im Seitenmenü die Seite 0 und versuchen es noch einmal. Sie können auch mit gedrückter **Alt-Taste** unter Windows bzw. **Wahltaste** auf Mac OS auf ein Objekt der Seite 0 klicken, um direkt darauf zuzugreifen.

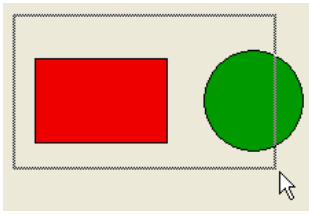
Wollen Sie alle Objekte im Hintergrund ausblenden, wählen Sie im Menü **Formular** bzw. im Kontextmenü des Formulareditors den Befehl **Anzeige** und deaktivieren Sie die Option Seite 0.

Mehrere Objekte auswählen

Oft ist es erforderlich, denselben Vorgang an mehreren Formularobjekten durchzuführen – zum Beispiel Objekte verschieben, ausrichten oder ihre Darstellung ändern. Mit 4D können Sie mehrere Objekte gleichzeitig auswählen. Dafür gibt es folgende Möglichkeiten:

- Wählen Sie im Menü **Bearbeiten** den Befehl **Alles auswählen**, um alle Objekte auszuwählen.
- Klicken Sie mit der rechten Maustaste auf das Objekt und wählen im Kontextmenü den Eintrag **Ähnliche Objekte auswählen**.
- Drücken Sie die **Umschalttaste** und klicken auf die Objekte, die Sie auswählen wollen.
- Wählen Sie eine Stelle außerhalb der gewünschten Objektgruppe und ziehen bei gedrückter Maustaste einen gestrichelten Rahmen um die Objekte (auch Auswahlrechteck genannt). Wenn Sie die Maustaste loslassen, ist jedes Objekt ausgewählt, das vollständig oder teilweise innerhalb dieses Auswahlrahmens liegt.
- Drücken Sie unter Windows die **Alt-Taste**, auf Macintosh die **Wahltaste** und ziehen bei gedrückter Maustaste einen gestrichelten Rahmen, sind nur die Objekte ausgewählt, die komplett innerhalb des Rahmens liegen.

Die folgende Abbildung zeigt die Auswahl von zwei Objekten:



Wollen Sie ein Objekt deaktivieren, das zu einer Objektgruppe gehört, klicken Sie mit gedrückter **Umschalttaste** auf das Objekt. Die anderen Objekte bleiben ausgewählt. Wollen Sie alle ausgewählten Objekte deaktivieren, klicken Sie außerhalb der Ränder dieser Objekte.

Objekte duplizieren

Sie können jedes Objekt im Formular duplizieren, einschließlich aktiver Objekte. Kopien aktiver Objekte erhalten alle Eigenschaften des Originals, einschließlich Name, Typ, automatische Aktion, Format und Objektmethode. Sie können ein bzw. mehrere ausgewählte Objekte direkt duplizieren oder mit dem Dialogfenster *Dupliziere* arbeiten. Hier können Sie in einem Durchgang von einem bzw. mehreren Objekten beliebig viele Kopien erstellen sowie den Versatz zwischen jedem duplizierten Objekt festlegen.

Beim Duplizieren einer Variable können Sie spezifische Funktionalitäten zum Duplizieren verwenden, um die Namen der Kopien automatisch durchnummerieren. Weitere Informationen dazu finden Sie im Abschnitt **Auf Matrix duplizieren**.

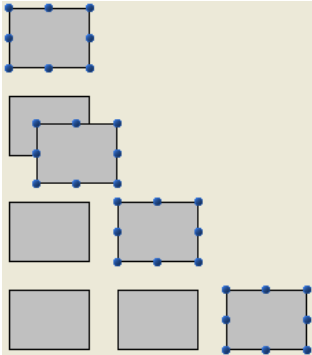
Um ein oder mehrere Objekte zu duplizieren:

1. Markieren Sie das bzw. die Objekte, die Sie duplizieren wollen.
2. Wählen Sie im Menü **Objekt** den Befehl **Duplizieren**.
3. Verschieben Sie die Kopie an die gewünschte Stelle.

Wählen Sie erneut den Befehl **Dupliziere**, erstellt 4D für jedes Objekt eine andere Kopie und setzt sie immer im selben Abstand und derselben Richtung zur ersten Kopie.

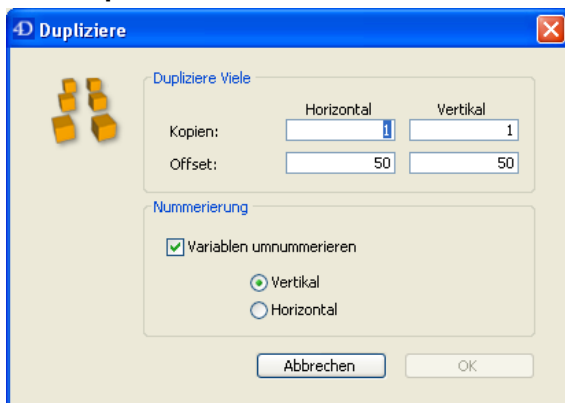
Wenn Sie Kopien des Objekts an einer Linie verteilen wollen, sollten Sie folgendes Verfahren anwenden: Duplizieren Sie das Originalobjekt, verschieben Sie die Kopie an eine andere Stelle im Formular und duplizieren Sie die Kopie. Die zweite Kopie wird automatisch im selben Abstand zur ersten Kopie wie die erste Kopie zum Originalobjekt gesetzt.

Darauffolgende Kopien werden ebenfalls im selben Abstand zu ihren Originalen gesetzt. Die folgende Abbildung zeigt, wie das relative Setzen von Kopien funktioniert:



Dupliziere Viele

Das Dialogfenster "Dupliziere" erscheint, wenn Sie ein oder mehrere Objekte auswählen und im Menü **Objekt** den Befehl **Viele Duplizieren** auswählen.



- Geben Sie im oberen Bereich die gewünschte Anzahl Objekte in horizontaler und vertikaler Richtung ein.
- Wollen Sie z.B. drei Objekte vertikal und zwei Objekte horizontal, geben Sie in Vertikal 3 und in Horizontal 2 ein. Wollen Sie in horizontaler Richtung drei weitere Kopien eines Objekts, geben Sie in Vertikal 3 ein und lassen Sie in Horizontal den Standardwert 1.
- Definieren Sie für Horizontal und Vertikal den Versatz zwischen den einzelnen Kopien. Der Wert wird in Punkt angegeben. Er gilt für die Kopie/n in Bezug auf das Originalobjekt. Soll beispielsweise der vertikale Versatz zwischen den Objekten 20 Punkte sein und die Höhe des Originalobjekts ist 50 Punkte, geben Sie in **Offset** für Vertikal 70 ein.
- Soll beispielsweise der vertikale Abstand zwischen den Objekten 20 Punkte sein und die Höhe des Originalobjekts ist 50 Punkte, geben Sie in **Offset** für Vertikal 70 ein.

- Wollen Sie eine Variablenvorlage erstellen, aktivieren Sie unter Nummerierung das Kontrollkästchen **Variablen unnummerieren** und markieren, ob die Variablen in horizontaler oder vertikaler Richtung nummeriert werden. Diese Option ist nur aktiv, wenn das ausgewählte Objekt eine Variable ist. Weitere Informationen dazu finden Sie im Abschnitt **Auf Matrix duplizieren**.

Objekte verschieben

Sie können beliebige grafische oder aktive Objekte im Formular einschließlich Datenfelder und Objekte verschieben, die mit einer Vorlage erzeugt wurden. Es gibt folgende Möglichkeiten:

- Objekt durch Ziehen bei gedrückter Maustaste verschieben
- Objekt um jeweils ein Pixel mit den Pfeiltasten verschieben
- Objekt mit gedrückter Umschalttaste und Pfeiltasten in Schritten von 20 Pixel verschieben
- In der Eigenschaftenliste unter dem Thema „Koordinaten&Größe“ die Werte manuell eingeben. Weitere Informationen finden Sie im unteren Abschnitt "Setting initial size of objects"XX.

Sobald Sie mit dem Ziehen beginnen, verschwinden die Griffe am Objekt. 4D zeigt Markierungen in den Linealen, die die Position der Objektränder andeuten, so dass Sie das Objekt ganz präzise verschieben können. Achten Sie darauf, dass Sie nicht an einem Griff ziehen, denn das verändert die Objektgröße. Drücken Sie die **Umschalttaste**, um Bewegungen mit einer Einschränkung auszuführen.

Ist das magnetische Raster aktiviert, werden die Objekte in Schritten verschoben, die durch Hilfslinien angezeigt werden. Weitere Informationen dazu finden Sie im Abschnitt **Magnetraaster verwenden**.

Um ein Objekt jeweils 1 Pixel zu verschieben:

1. Markieren Sie das bzw. die gewünschten Objekte.
2. Drücken Sie die entsprechende Pfeiltaste auf der Tastatur.
Das Objekt wird mit jedem Drücken um 1 Pixel in Pfeilrichtung verschoben.

Um ein Objekt in Pixelschritten zu verschieben:

1. Markieren Sie das bzw. die gewünschten Objekte.
2. Drücken Sie die **Umschalttaste** und eine der Pfeiltasten auf der Tastatur.
Das Objekt wird standardmäßig jeweils um 20 Pixel verschoben. In den 4D Einstellungen auf der **Seite Formulare** können Sie über die Option *Verschieben mit Tastatur* einen anderen Wert definieren.

Objekte gruppieren

4D ermöglicht das Gruppieren von Objekten. Sie können dann die Gruppe wie ein einzelnes Objekt auswählen, verschieben und ändern. Gruppierte Objekte behalten ihre relative Position zueinander. Sie können zum Beispiel ein Datenfeld und seine Bezeichnung gruppieren, eine unsichtbare Schaltfläche und ihr Symbol, usw. Gruppen können Teile von anderen Gruppen sein.

Wenn Sie die Größe einer Gruppe ändern, werden alle Objekte in der Gruppe proportional angepasst (mit Ausnahme von Textbereichen, deren Größen schrittweise gemäß deren Schriftgrößen geändert werden).

Für die koordinierte Funktionsweise von Optionsfeldern sind gruppierte Objekte erforderlich (siehe Abschnitt **Optionsfelder und Optionsbilder**). Gruppieren wird auch zum Synchronisieren rollbarer Bereiche verwendet (siehe Handbuch *4D Programmiersprache*, Abschnitt **Gruppierte rollbare Bereiche**).

Sie können die Gruppierung von Objekten auflösen, um sie wieder wie einzelne Objekte handzuhaben.

Die Gruppierung eines aktiven Objekts muss wieder aufgelöst werden, bevor Sie auf dessen Eigenschaften oder Methode zugreifen können. Sie können dagegen ein Objekt innerhalb einer Gruppe auswählen, ohne die Gruppierung aufzuheben, wenn Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** auf das Objekt klicken. Dazu muss die Gruppe ausgewählt sein.

Gruppieren hat für Objekte nur im Formulareditor Bedeutung. Beim Ausführen des Formulars verhalten sich alle gruppierten Objekte - mit Ausnahme von Optionsfeldern und rollbaren Bereichen - als wären sie ungruppiert.

Hinweis: Objekte, die auf verschiedenen Ebenen liegen, lassen sich nicht gruppieren. Nur die Objekte auf der aktuellen Ebene lassen sich gruppieren (siehe **Formularebenen verwenden**).

Um Objekte zu gruppieren:

1. Markieren Sie die Objekte, die Sie gruppieren wollen.
2. Wählen Sie im Menü **Objekt** den Befehl **Gruppieren**
ODER

Klicken Sie in der Werkzeugleiste des Formulareditors auf die Schaltfläche für Gruppieren  .

4D versieht die Ränder der neu gruppierten Objekte mit Griffen. Die Ränder der einzelnen Objekte innerhalb der Gruppe sind nicht durch Griffe markiert. Wenn Sie jetzt das gruppierte Objekt ändern, ändern Sie alle Objekte, die Teil der Gruppe sind.

Um eine Gruppierung aufzulösen:

1. Markieren Sie das Objekt, dessen Gruppierung Sie auflösen wollen.
2. Wählen Sie im Menü **Objekt** den Befehl **Gruppierung auflösen** (Variante der Schaltfläche Gruppieren)
ODER
Klicken Sie in der Werkzeugleiste des Formulareditors auf die Schaltfläche für Gruppierung auflösen.
Erscheint der Befehl **Gruppierung auflösen** in Grauschrift, ist das Objekt bereits in seiner einfachsten Form und es gibt keine Gruppierung zum Auflösen.

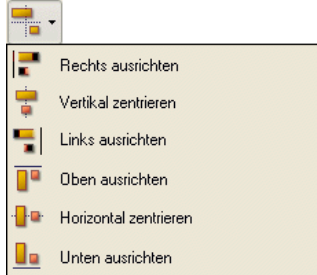
Objekte ausrichten

Im Formulareditor können Sie die Objekte untereinander oder über ein magnetisches Raster ausrichten.

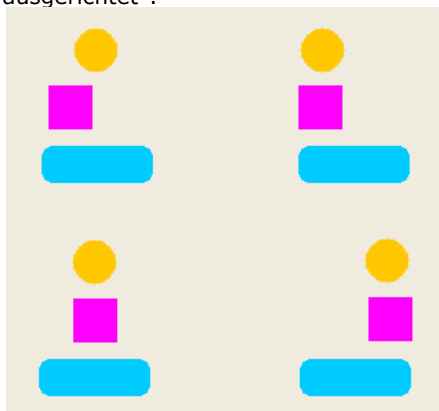
- Wenn Sie Objekte untereinander ausrichten, können Sie diese oben, unten, seitlich, horizontal mittig oder vertikal mittig ausrichten. Sie können eine Objektauswahl direkt ausrichten oder ein Dialogfenster verwenden, in dem Sie sowohl die Ausrichtung als auch die Verteilung der Objekte definieren können. Hier können Sie auch das Ausgangsobjekt bestimmen, nach dem die anderen Objekte ausgerichtet werden sollen. Eine Vorschau zeigt das Ergebnis Ihrer Einstellungen.
- Wenn Sie das magnetische Raster verwenden, lässt sich jedes Objekt manuell mit den anderen Objekten ausrichten, basierend auf der relativen Position untereinander, die durch gestrichelte Hilfslinien angezeigt wird.

Die Ausrichtungssymbole verwenden

Mit den Symbolen zum Ausrichten in der Werkzeugleiste (siehe **Formulareditor**) und den Befehlen im Untermenü **Ausrichten** des Menüs **Objekt** bzw. des Kontextmenüs können Sie Objekte rasch untereinander ausrichten.

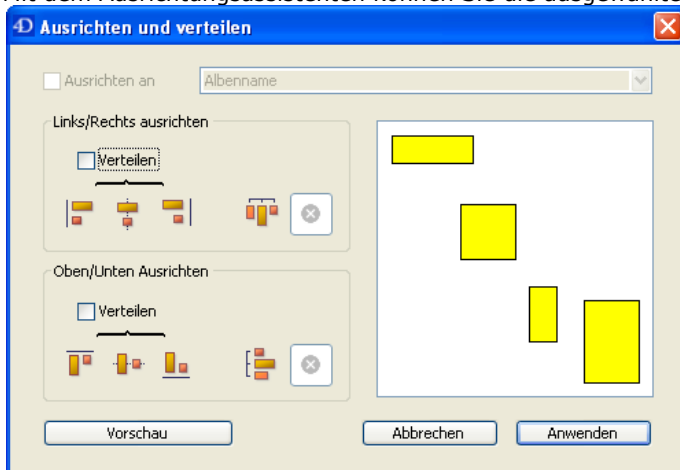


Beim Ausrichten verwendet 4D das der Ausrichtung am nächsten liegende Objekt als „Anker“ und richtet die anderen Objekte daran aus. So wird z.B. beim Links-Ausrichten das am weitesten links stehende Element als Ausgangspunkt gewählt. Die folgende Abbildung zeigt Objekte ohne Ausrichtung, "links-ausgerichtet", "horizontal mittig ausgerichtet" und "rechts ausgerichtet":



Ausrichtungsassistent verwenden

Mit dem Ausrichtungsassistenten können Sie die ausgewählten Objekte nach allen Richtungen ausrichten und/oder verteilen.



Um dieses Dialogfenster anzuzeigen, wählen Sie die gewünschten Objekte aus und dann im im Menü **Objekt** oder im Kontextmenü des Editors das Untermenü **Ausrichten**.

1. Klicken Sie im Bereich Links/Rechts ausrichten und/oder Oben/Unten ausrichten auf das passende Icon. Die Vorschau zeigt das Ergebnis Ihrer Wahl.
2. Um die ausgewählten Objekte standardmäßig auszurichten, klicken Sie auf die Schaltfläche **Vorschau** oder **Anwenden**. 4D nimmt dann beim Links-Ausrichten das am weitesten links stehende Element als Ausgangspunkt, beim Rechts-Ausrichten das am weitesten rechts stehende Element.
ODER:
Um Objekte nach einem bestimmten Objekt auszurichten, markieren Sie das Optionsfeld **Ausrichten an** und wählen den entsprechenden Objektnamen in der DropDown-Liste aus. In diesem Fall ändert sich die Position des Referenzobjekts nicht.

Um das Ergebnis der Ausrichtung zu betrachten, klicken Sie auf die Schaltfläche **Vorschau**. Die Objekte werden dann im Formulareditor ausgerichtet. Das Dialogfenster Ausrichten und verteilen bleibt weiterhin im Vordergrund, so dass Sie die

Ausrichtung anwenden oder annullieren können.

Hinweis: Über dieses Dialogfenster können Sie Objekte in einer Operation ausrichten und verteilen. Weitere Informationen zum Verteilen finden Sie im Abschnitt **Objekte verteilen**.

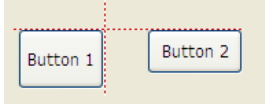
Magnetraaster verwenden

Der Formulareditor enthält ein visuelles magnetisches Raster, das zum Anordnen und Ausrichten von Objekten in einem Formular hilfreich ist. Die Magnetfunktion basiert auf der relativen Position der Objekte untereinander. Sie lässt sich nur verwenden, wenn im Formular mindestens zwei Objekte vorhanden sind.

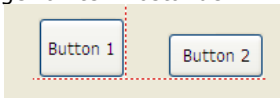
Das Magnetraaster arbeitet folgendermaßen: Legen Sie ein Objekt per Drag&Drop in das Formular, zeigt 4D seine mögliche Positionierung in Bezug auf andere Formularobjekte:

- Horizontal, wenn die Oberkante oder Mitte von zwei Objekten übereinstimmen
- Vertikal, wenn die Seitenkanten von zwei Objekten übereinstimmen

Ist das der Fall, legt 4D das Objekt an die jeweilige Position und zeigt mittels einer roten Linie die mögliche Ausrichtung an:



4D gibt bei der Verteilung von Objekten einen für Oberflächen üblichen Abstand an. Wieder kennzeichnen rote Linien die gewählten Abstände.



Diese Funktionsweise gilt für alle Objektarten im Formular. Der Magnetismus lässt sich jederzeit über den Befehl **Magnetisches Raster** im Menü **Formular** oder über das Kontextmenü des Editors aktivieren bzw. deaktivieren. Sie können das Magnetraaster auch standardmäßig ein- oder ausschalten, und zwar in den 4D Einstellungen auf der **Seite Formulare** mit der Option *Aktiviere autom. Ausrichtung als Standard*. Sie können das Magnetraaster bei einem ausgewählten Objekt zeitweise ein- oder ausschalten, wenn Sie unter Windows die **Strg-Taste**, auf Mac OS die **Befehlstaste** wählen.

Hinweis: Das Magnetraaster beeinflusst auch das manuelle Verteilen von Objekten.

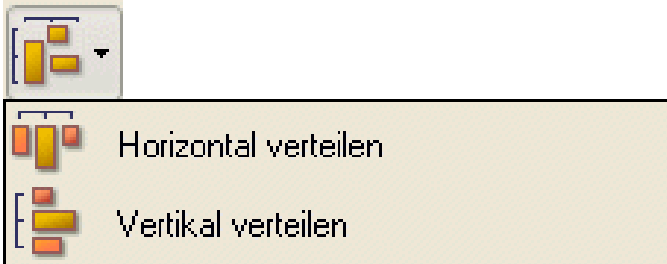
Objekte verteilen

Sie können drei oder mehr Objekte mit gleichmäßigem Abstand untereinander verteilen. Für einfaches Verteilen verwenden Sie die Werkzeugleiste, für die gesamte Bandbreite das Dialogfenster Ausrichten. Hier können Sie auch Ausrichten und Verteilen miteinander kombinieren.

Hinweis: Ist das magnetische Raster aktiviert, erscheinen beim manuellen Verschieben eines Objekts Hilfslinien, die das Verteilen vereinfachen. Weitere Informationen dazu finden Sie im vorigen Abschnitt.

Um Objekte mit gleichem Abstand zu verteilen:

1. Wählen Sie drei oder mehr Objekte aus.
2. Klicken Sie in der Werkzeugleiste auf das gewünschte Symbol zum Verteilen



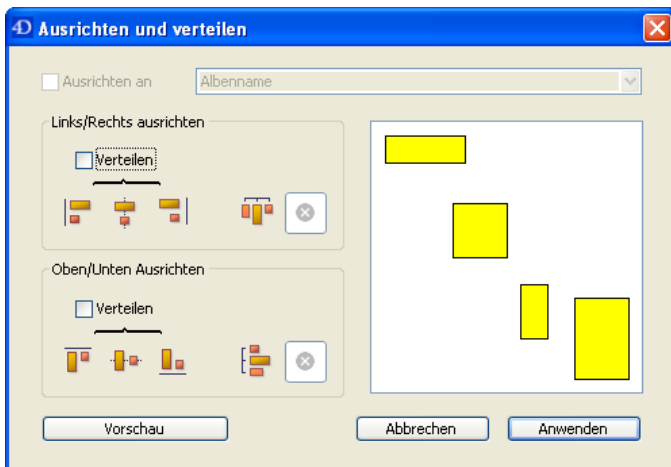
ODER

Wählen Sie im Menü **Formular** bzw. im Kontextmenü des Formulareditors das Untermenü **Anzeige** und dann den Eintrag horizontal bzw. vertikal verteilen.

4D verteilt von der Mitte aus alle Objekte mit gleichen Abständen. Als Referenz dient der größte Abstand zwischen zwei benachbarten Objekten.

Um Objekte mit dem Dialogfenster *Ausrichten und Verteilen* zu verteilen:

1. Wählen Sie die entsprechenden Objekte aus.
2. Wählen Sie im Menü **Objekt** oder im Kontextmenü das Untermenü **Ausrichten** und dann den Eintrag **Ausrichten**. Auf dem Bildschirm erscheint folgendes Fenster:



3. Klicken Sie zur Standardverteilung (horizontal oder vertikal) auf das Icon Ihrer Wahl.

 (Icon für standardmäßige horizontale Verteilung)

Die Vorschau zeigt das Ergebnis Ihrer Wahl.

4. Um die ausgewählten Objekte standardmäßig zu verteilen, klicken Sie auf die Schaltfläche **Vorschau** oder **Anwenden**. Die Objekte werden dann im gleichen Abstand voneinander verteilt.

ODER:

Um eine spezifische Verteilung auszuführen, markieren Sie das Kontrollkästchen **Verteilen** (z.B. um Objekte rechtsbündig anzuordnen). Diese Option funktioniert wechselseitig. Ist das Kontrollkästchen **Verteilen** aktiviert, arbeiten die darunterliegenden Icons anders:

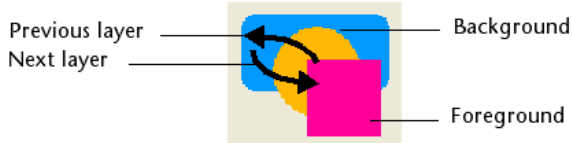
- Horizontal entsprechen die Icons folgender Verteilung: bündig in Bezug auf die linken Kanten, Mitten (hor.) und rechten Kanten der ausgewählten Objekte.
- Vertikal entsprechen die Icons folgender Verteilung: bündig in Bezug auf die oberen Kanten, Mitten (vert.) und unteren Kanten der gewählten Objekte.

Um das Ergebnis der Verteilung zu betrachten, klicken Sie auf die Schaltfläche **Vorschau**. Die Operation wird dann im Formulareditor ausgeführt. Das Dialogfenster *Ausrichten und Verteilen* bleibt weiterhin im Vordergrund, so dass Sie die Verteilung anwenden oder annullieren können.

Hinweis: Über dieses Dialogfenster können Sie Objekte in einer Operation ausrichten und verteilen. Weitere Informationen zum Ausrichten finden Sie im oberen Abschnitt "Objekte ausrichten".

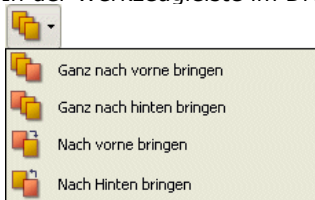
Objekte übereinanderlegen

Sie müssen manchmal Objekte neu anordnen, die die Sicht auf andere Objekte im Formular verdecken. Sie möchten zum Beispiel eine Grafik verwenden, die hinter den Datenfeldern in einem Formular erscheint. 4D stellt zum Übereinanderlegen von Objekten die Optionen **Ganz nach vorne bringen**, **Ganz nach hinten bringen** und **Nach vorne bringen**, **Nach hinten bringen** zur Verfügung. Diese Ebenen bestimmen auch die standardmäßige Eingabereihenfolge (siehe **Eingabereihenfolge der Daten ändern**). Die folgende Abbildung zeigt ein Beispiel:



Um ein oder mehrere Objekte auf eine andere Ebene zu legen, wählen Sie das bzw. die gewünschten Objekte aus und dann:

- Im Menü **Objekt** den Befehl **Ganz nach hinten bringen**, **Ganz nach vorne bringen** bzw. **Nach vorne bringen**, **Nach hinten bringen**,
- Im Kontextmenü des Editors im Untermenü **Position** einen der oben genannten Befehle,
- In der Werkzeugleiste im DropDown-Menü für Ebenen auf das entsprechende Icon.



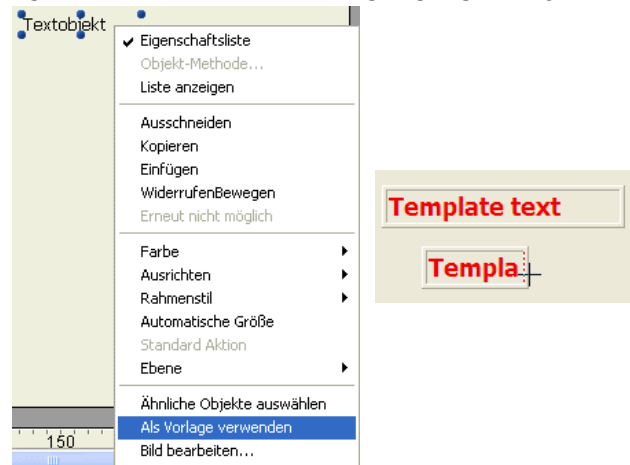
Hinweis: Liegen mehrere Objekte übereinander, können Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** der Reihe nach jedes Objekt auswählen. Jeder Klick geht eine Ebene tiefer.

Beim Anordnen verschiedener Ebenen durchläuft 4D die Ebenen immer von der letzten zur ersten Ebene. Folglich setzt die *vorige* Ebene die Objektauswahl im Formular eine Ebene nach hinten, die *nächste* Ebene eine Ebene nach vorne.

Objekt als Vorlage definieren

Klicken Sie auf ein Objekt, erscheint im Kontextmenü des Formulareditors der Eintrag **Als Vorlage verwenden**. Damit sichern Sie das angeklickte Objekt als Vorlage; erstellen Sie ein Objekt vom selben Typ, wird diese Vorlage mit den dazugehörigen Eigenschaften verwendet.

Im folgenden Beispiel wird der Befehl auf ein Textobjekt angewandt. Alle anschließend erstellten Textobjekte verwenden die Eigenschaften des standardmäßig angelegten Objekts



Sie können für jeden Objekttyp eine eigene Vorlage verwenden. In der Anwendung gibt es für jeden Objekttyp eine selbst angelegte Vorlage. Die Vorlage sichert alle festgelegten Objekteigenschaften, wenn der Befehl ausgeführt wird – mit Ausnahme der festen Koordinaten und der Objektmethode (falls zutreffend).

Objektvorlagen löschen

Um eine Objektvorlage zu ändern bzw. zu ersetzen, müssen Sie für das betreffende Objekt eine neue erstellen.

Sie können auch alle Objektvorlagen entfernen und zur Standardvorlage zurückkehren. Dazu wählen Sie im Menü **Objekt** des Formulareeditors den Befehl **Eigene Vorlagen löschen**. Auf dem Bildschirm erscheint dann ein Dialog zum Bestätigen. Klicken Sie auf die Schaltfläche **Bestätigen**, werden alle Objektvorlagen auf ihre Standardwerte zurückgesetzt.

Objekteigenschaften setzen

Jedes Objekt in einem Formular hat Eigenschaften, die seine Darstellung bestimmen und wie es in der Anwendungsumgebung angezeigt und in der Größe angepasst wird. Diese Eigenschaften sind über die Eigenschaftenliste verfügbar (siehe **Formulareditor**).

Eigenschaften speziell für dynamische Objekte finden Sie in den folgenden Kapiteln: **Mit aktiven Objekten arbeiten**, **Listboxen** und **Unterformulare und Widgets**.

Objektgröße

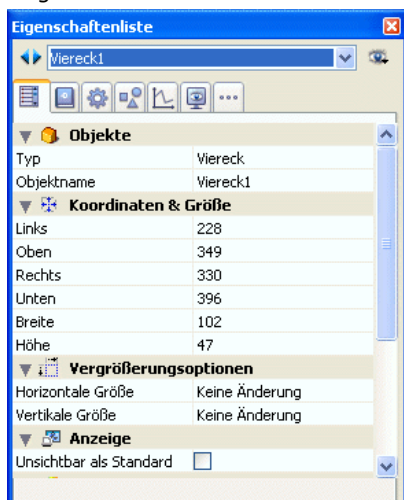
Um die Größe eines Objekts im Formulareditor zu setzen, können Sie:

- an einem seiner Vergrößerungsgriffe ziehen
- Das Objekt durch Drücken der **Strg-Taste** unter Windows, **Befehlstaste** auf Macintosh und der Pfeiltasten um je 1 Pixel vergrößern/verkleinern.
- Das Objekt schrittweise vergrößern/verkleinern (standardmäßig um jeweils 20 Pixel).
- Seine Ausmaße in der Eigenschaftenliste unter dem Thema „Koordinaten&Größe“ manuell eingeben.

Sie können die Größe eines Objekt über seine Handgriffe standardmäßig ändern. Ziehen Sie mit gedrückter **Umschalttaste** am Griff, ist die Bewegung eingeschränkt: Striche können dann nur vertikal, 45° oder horizontal, Rechtecke nur quadratisch und Ovale nur kreisrund sein. Ist das magnetische Raster eingeschaltet, erfolgt die manuelle Anpassung in Schritten von 5 Pixel. Weitere Informationen dazu finden Sie im Abschnitt **Magnetraaster verwenden**.

Um die Größe eines Objekts um jeweils 1 Pixel zu verändern, wählen Sie das Objekt aus und betätigen Sie die Pfeiltasten, unter Windows mit gedrückter **Strg-Taste**, auf Macintosh mit gedrückter **Befehlstaste**. Der Pfeil nach oben bzw. nach unten verändert die Objekthöhe, der Pfeil nach rechts bzw. nach links die Objektbreite. Betätigen Sie die Pfeiltasten, unter Windows mit gedrückter **Umschalt- und Strg-Taste**, auf Macintosh mit gedrückter **Umschalt- und Befehlstaste**, erfolgt die Anpassung in Schritten (standardmäßig in Schritten von 20 Pixel). Sie können diesen Wert in den 4D Einstellungen auf der **Seite Formulare** ändern.

Um ein Objekt über Koordinaten zu verschieben bzw. seine Größe zu verändern, wählen Sie das Objekt aus und erweitern Sie in der Eigenschaftenliste die Gruppe Koordinaten & Größe. Die Koordinaten erscheinen – je nach Einheit auf dem Lineal – in Pixel, Zentimeter oder Zoll. Der Ursprung der Koordinaten (0,0) liegt links oben. Geben Sie neue Werte in die Eingabebereiche der Koordinaten ein.



4D setzt die Objektränder auf die eingegebenen Koordinaten. Das Objekt wird je nach den eingetragenen Werten in der Größe geändert oder verschoben (oder beides).

Hinweis: In einer Multiplattform-Umgebung kann bei Verwendung von Stilvorlagen die Objekthöhe variieren, da diese Höhe automatisch angepasst wird, so dass sie ein Vielfaches der Höhe ist, die in der aktuellen Plattform für die Schrift gesetzt wurde. Weitere Informationen dazu finden Sie im Kapitel **Stilvorlagen**.

Automatische Größe

Sie können statischen Text und Bilder automatisch anpassen, damit sie mit ihren grafischen Merkmale, wie Schriftgröße, -stil, etc., optimal angezeigt werden:

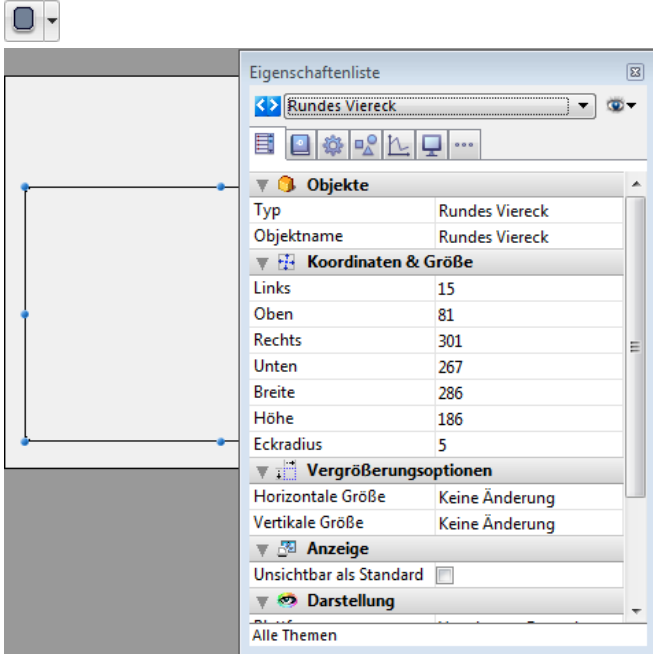
1. Klicken Sie mit der rechten Maustaste auf das Objekt und wählen Sie im Kontextmenü den Eintrag **Automatische Größe**
ODER
Drücken Sie unter Windows die **Strg-Taste**, auf Mac OS die **Befehlstaste** und klicken Sie in die untere rechte Ecke des Objekts.

Dieser Befehl führt folgendes aus:

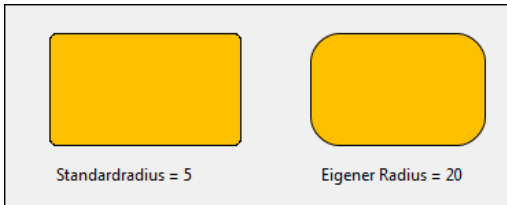
- Statische Textbereiche, Kontrollkästchen, Optionsfelder und Schaltflächen werden angepasst, so dass ihr Inhalt exakt ihren Kanten entspricht.
- Felder und Variablen werden angepasst, so dass ihre Höhe mindestens der Höhe der aktuellen Schrift und ihre Breite 100 Punkt entspricht — ist das Objekt größer als diese Mindestmaße, hat der Befehl keine Auswirkung.
- Bilder oder Bildschaltflächen erscheinen in der Originalgröße des Quellbildes.
- Rollbare Bereiche und hierarchische Listen erscheinen in einer Höhe, die der Höhe der aktuellen Schrift entspricht.
- Comboboxen, PopUp/DropDown-Listen und hierarchische PopUp-Menüs erscheinen mit ihrer standardmäßigen Mindesthöhe, wenn die aktuelle Höhe nicht ausreicht.

Eckradius (Vierecke)

Im Formulareditor ist für runde Vierecke unter der Gruppe "Koordinaten & Größe" die Eigenschaft **Eckradius** verfügbar.



Als Radius ist standardmäßig 5 Pixel vorgegeben. Diesen Wert können Sie an eigene Bedürfnisse anpassen. Hier ein Beispiel:



Der Mindestwert ist 0, in diesem Fall wird das Viereck ohne gerundete Ecken gezeichnet.

Der maximale Wert richtet sich nach der Größe des Vierecks und wird dynamisch berechnet. Er kann maximal halb so lang wie die kürzere Seite des Vierecks sein.

Sie können diese Eigenschaft auch über die Befehle **OBJECT Get corner radius** and **OBJECT SET CORNER RADIUS** setzen.

Größe anpassen

Verändert ein Benutzer in der Anwendungsumgebung die Größe eines Fensters, das ein Formular anzeigt, werden die darin enthaltenen Objekte ebenfalls angepasst oder bewegt.

Automatisches **Wachsen** bewirkt, dass ein Objekt mitwächst, wenn das Formular vergrößert wird, bzw. schrumpft, wenn das Fenster verkleinert wird. Wenn Sie zum Beispiel einen Rahmen verwenden, der die Datenfelder in einem Eingabeformular umschließt, wächst er beim Vergrößern des Fensters proportional mit.

Automatisches **Bewegen** verschiebt das Objekt horizontal oder vertikal, wenn das Formular in der Größe geändert wird. Ist die Option **Bewegen** aktiviert, versucht 4D das Objekt sichtbar zu halten, wenn der Benutzer das Fenster verkleinert. Wenn ein Benutzer zum Beispiel die Größe der Schaltflächenreihe ändert, so dass einige Schaltflächen verschwinden, versucht 4D die Schaltflächen horizontal oder vertikal so zu bewegen, dass sie sichtbar bleiben.

Es gibt die Zeilen horizontale Größe und vertikale Größe. Hier können Sie zwischen den Eigenschaften **Keine Änderung**, **Wachsen** und **Bewegen** wählen.

Option

Horizontale Größe:
Wachsen

Horizontale Größe:
Bewegen

Vertikale Größe:
Wachsen

Vertikale Größe:
Bewegen

Ergebnis

Vergrößert der Benutzer das Fenster in der Breite, vergrößert 4D das Objekt um denselben Prozentsatz nach rechts.

Vergrößert der Benutzer das Fenster in der Breite, bewegt 4D das Objekt um dieselbe Anzahl Pixel nach rechts.

Vergrößert der Benutzer das Fenster in der Höhe, vergrößert 4D das Objekt um denselben Prozentsatz nach unten.

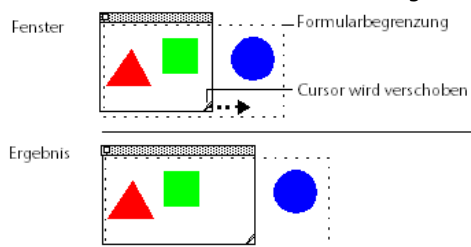
Vergrößert der Benutzer das Fenster in der Höhe, bewegt 4D das Objekt um dieselbe Anzahl Pixel nach unten.

Mit den Optionen zur Repositionierung können Sie das Objekt in der angegebenen Richtung bewegen, damit es sichtbar bleibt.

Mit der Option **Keine Änderung** bleibt das Objekt unverändert, wenn das Formular verkleinert/vergrößert wird.

Seitenschieber

In 4D sind die rechte und untere Seite von Fenstern standardmäßig mit "Splittern vom Typ Schieber" ausgestattet. Das heißt, Objekte rechts oder unterhalb der Begrenzung eines Fensters auf dem Bildschirm werden beim Vergrößern des Fensters automatisch nach rechts oder unten geschoben.



Hinweis: Das funktioniert nicht bei Fenstern mit Rollbalken.

Unsichtbar als Standard

Sie können den meisten Formularobjekten die Eigenschaft **Unsichtbar als Standard** zuweisen.

Diese Eigenschaft ist unter der Gruppe "Anzeige" der Eigenschaftsliste verfügbar. Sie vereinfacht das Entwickeln dynamischer Oberflächen. In diesem Zusammenhang ist es oft notwendig, während dem Ereignis On load im Formular Objekte per Programmierung auszublenden und anschließend bestimmte Objekte einzublenden.

Mit der Eigenschaft **Unsichtbar als Standard** können Sie diese Logik umkehren, indem Sie bestimmte Objekte standardmäßig ausblenden. Der Entwickler kann ihre Anzeige je nach Kontext dann über den 4D Befehl **OBJECT SET VISIBLE** programmieren.

Hinweis: Verwechseln Sie diese Eigenschaft nicht mit der Eigenschaft **Unsichtbar** in PopUp-Menüs, die das Zeichnen von Objekten nicht zulässt, solange sie aktiv sind.

Plattformoberfläche

Sie können die Plattformoberfläche für jedes Objekt getrennt festlegen. Gehen Sie dazu in der Eigenschaftsliste zur Gruppe "Darstellung". Es gibt folgende Möglichkeiten:

- **Vererbt von Formular:** Die Plattformoberfläche für das Objekt ist die gleiche wie für das Formular. Die Plattformoberfläche des Formulars wird im Dialogfenster **Formulareigenschaften** festgelegt.
- **System:** Das Objekt wird gemäß der Plattform des Betriebssystems erstellt – unabhängig, welche Plattform für das Formular definiert ist.
- **Drucken:** Das Objekt wird für den Druck angepasst erstellt – unabhängig, welche Plattform für das Formular definiert ist.

Hinweis: Für Datenbanken, die von einer früheren Version von 4D konvertiert wurden, sind weitere Einstellungen verfügbar. Weitere Informationen dazu finden Sie im Abschnitt **Formulareigenschaften**.

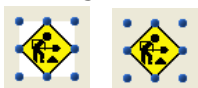
Hintergrund und Rahmen

Über die Eigenschaften Hintergrund und Rahmen können Sie die Ränder und die grafische Darstellung von Objekten verändern.

Transparent

Das Attribut **Transparent** ist unter der Gruppe "Hintergrund und Rahmen" für Objekte verfügbar, die gefüllt oder transparent sein können: Felder, Variablen, Listboxen, Schaltflächen, Bilder oder statischer Text, etc. Bilder werden standardmäßig mit weißem Hintergrund eingesetzt. Sie können den Hintergrund des Bildes transparent machen, so dass es die Hintergrundfarbe bzw. das Hintergrundmuster des Formulars annimmt.

Nachfolgendes Beispiel zeigt ein statisches Bild mit und ohne das Attribut **Transparent**:



Randstil

Sie können für die meisten Formularobjekte einen Randstil festlegen. Gehen Sie dazu in der Eigenschaftsliste zur Gruppe "Hintergrund und Rahmen" oder wählen Sie im Kontextmenü des Objekts den Eintrag **Rahmenstil**. Es gibt folgende Wahlmöglichkeiten:

- **Transparent:** Objekte erscheinen ohne Rahmen.
- **Normal:** Objekte erscheinen umrahmt mit Linie in 1 Punkt Stärke.
- **Gepunktet:** Objekte erscheinen umrahmt mit gepunkteter Linie in 1 Punkt Stärke.
- **Erhoben:** Objekte erscheinen mit erhabenem Rahmen in 3D Effekt.
- **Vertieft:** Objekte erscheinen mit vertieftem Rahmen in 3D Effekt.
- **Doppelt:** Objekte erscheinen mit doppelter Umrahmung, z.B. zwei Linien von 1 Punkt Stärke, mit 1 Pixel Abstand zueinander.
- **System:** Objekte erscheinen unter Windows umrahmt mit einer blauen Linie von 1 Punkt Stärke, auf Mac OS mit vertieftem Rahmen.

Das folgende Beispiel vergleicht diese Stilarten unter Windows mit der Plattformoberfläche "System":

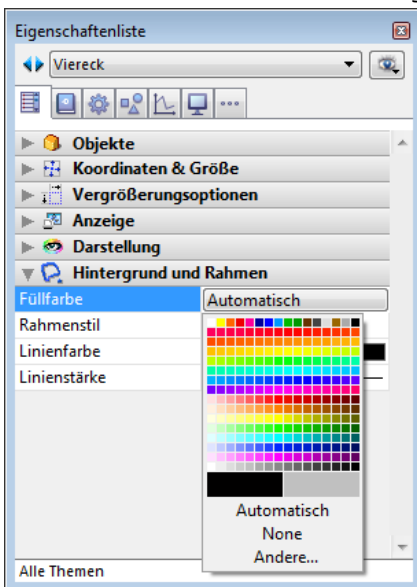


Weitere Informationen zu den Auswirkungen dieser Optionen auf verschiedene Objekttypen finden Sie in den Abschnitten **Schaltflächen** und **Kontrollkästchen**.

Farben für Hintergrund und Rahmen

4D ermöglicht, Objekten Farben zuzuordnen zur Anzeige auf einem Farbmonitor oder für farbige Ausdrücke, wenn Ihr Drucker Farbe unterstützt.

Sie können für den Hintergrund und Rahmen von Objekten verschiedene Farben festlegen, entweder über den Eintrag **Farbe** im Kontextmenü oder direkt in der Eigenschaftsliste:



Hinweis: In der Eigenschaftsliste lautet die Hintergrundfarbe **Füllfarbe** und die Rahmenfarbe **Linienfarbe**. Im Kontextmenü entspricht die Vordergrundfarbe der Rahmenfarbe.

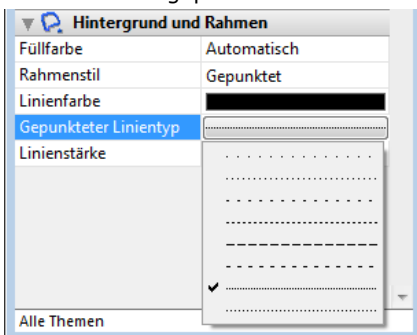
In der Eigenschaftsliste gibt es für die Hintergrundfarbe zwei Standardoptionen:

- **Automatisch:** das Objekt erscheint mit den im Betriebssystem festgelegten Farben
- **Keine:** Der Objekthintergrund ist transparent

In der Eigenschaftsliste können Sie mit der Option **Andere** die Farbpalette des Systems anzeigen und hier individuell eine Farbe festlegen.

Gepunkteter Linientyp

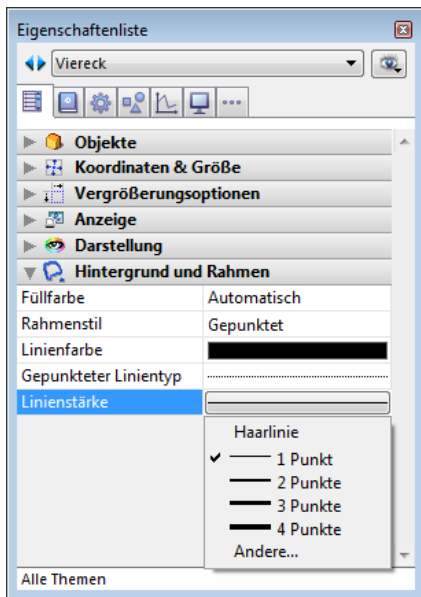
Diese Eigenschaft ist verfügbar, wenn Sie als **Rahmenstil** "Gepunktet" wählen. Sie können dann im PopUp-Menü zwischen verschiedenen gepunkteten Linien wählen:



Linienstärke

4D ermöglicht, verschiedene Stärken für Linien und Objekte mit Linien, wie Ovale, Raster und Rechtecke festzulegen.

Sie können die Linienstärken in der Eigenschaftsliste unter "Linienstärke" festlegen:



Sie können zwischen 5 Stärken wählen. Die erste Stärke ist Haarlinie, das ist die dünnste Linie, die der Drucker drucken kann. Wenn Sie den Eintrag **Andere** auswählen, zeigt 4D ein Dialogfenster, in dem Sie eine beliebige Linienstärke bis zu 20 Punkt festlegen können.

Textattribute

Sie können Textattribute für Textobjekte in der Eigenschaftensliste unter der Gruppe "Text" festlegen.

Sie können folgende Attribute setzen:

- **Stilvorlage:** Weist dem Objekt eine Stilvorlage zu. Weitere Informationen dazu finden Sie im Abschnitt **Stilvorlagen**
- **Schrift:** Name der Schrift, die im Objekt verwendet wird
- **Stil:** Stil, der im Objekt verwendet wird (fett, kursiv, unterstrichen)
- **Schriftgröße:** Verwendete Schriftgröße in Punkt
- **Schriftfarbe:** Verwendete Schriftfarbe. Im Kontextmenü entspricht die Schriftfarbe der Farbe für den **Vordergrund**
- **Zeilenstil Array / Hintergrundfarbe Array:** Name des Array zum Verwalten von Schriftstil und Farbe (nur verfügbar für Listboxen vom Typ Array)
- **Horizontale Ausrichtung:** Horizontale Position des Textes im Textbereich des Objekts
- **Vertikale Ausrichtung:** Vertikale Position des Textes im Textbereich des Objekts (nur für Listboxen)
- **Rotation:** Ändert die Rotation des Textbereichs. Weitere Informationen dazu finden Sie im Abschnitt **Textrotation**
- **Mehrfachstil:** Verwendet in der Anwendungsumgebung in dynamischen Textbereichen, wie Felder oder Variablen eigene Stilarten. Weitere Informationen dazu finden Sie im Abschnitt **Text mit Stil (Rich Text)**.
- **Schrift/Farbpalette erlauben:** Erlaubt dem Benutzer, Schrift oder Farbe in der aktuellen Sitzung direkt zu ändern, wenn die Palette des Systems über den Befehl **OPEN FONT PICKER** bzw. **OPEN COLOR PICKER** aufgerufen wird. Diese Option ist für Formularobjekte vom Typ Feld, Variable oder Combo Box verfügbar. Standardmäßig ist sie inaktiv. Sie müssen sie explizit für jedes Objekt markieren, für das die Schrift bzw. Farbe über die jeweilige Palette des Systems änderbar sein soll.

Hinweis zu Kompatibilität beim Rendern von Text (OS X)

Gemäß den Empfehlungen von Apple haben wir für v13 auf OS X das *Frameworks* zum Text Rendern in 4D aktualisiert und vereinheitlicht (das obsoleete QD/MLTE *Frameworks* wurde entfernt, jetzt gilt generell die Anwendung von CoreText).

Trotz unserer Bemühungen, die Veränderungen gering zu halten, können durch das neue CoreText *Framework* beim Rendern von Textbereichen für Anwendungen, die in 4D v13 auf OS X konvertiert wurden, Variationen auftreten, insbesondere bei Zeilenabständen, d.h. Sie müssen u.U. bestimmte Formularobjekte beim Konvertieren von v12 Anwendungen nachbearbeiten.

Textrotation

Sie können in 4D Textbereiche in Ihren Formularen drehen:

	New York	Chicago	Los Angeles	San Diego
Alpha	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bravo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Charlie	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Delta	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Echo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Die Textrotation lässt sich im Formulareditor (dauerhafte Eigenschaft) oder über den neuen Befehl **OBJECT SET TEXT ORIENTATION** (Eigenschaft für den aktuellen Prozess) verwenden.

Drehbare Objekte

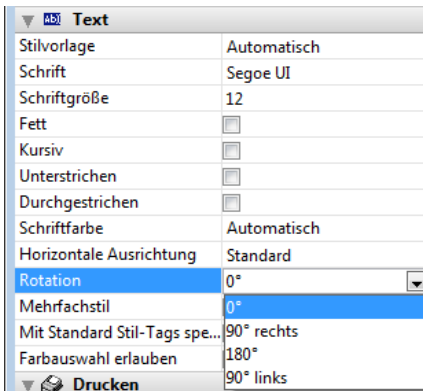
Text lässt sich drehen, wenn es sich um nicht-eingebbare Textbereiche in Formularen handelt. Das gilt für:

- Statische Texte
- Nicht-eingebbare Variablen und Felder, die als Text gewertet werden. Das sind Objekte mit textbasiertem Inhalt, wie Zeichenketten, aber auch Datum, Zeit oder Zahl sowie Objekte mit mehreren Zeilen oder Mehrfachstil.

Andere Formularobjekte, wie Schaltflächen, Eingabebereiche, Listen, Optionsfelder, etc. lassen sich nicht drehen.

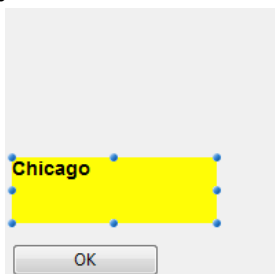
Rotation setzen

Die Eigenschaft "Rotation" liegt in der Eigenschaftsliste unter der Gruppe "Text". Sie erscheint nur, wenn das Objekt Rotation unterstützt. Textbereiche werden um 90° oder um ein Vielfaches davon gedreht:

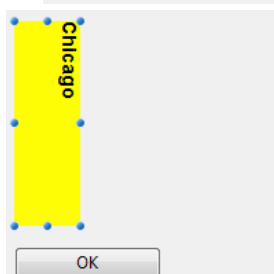


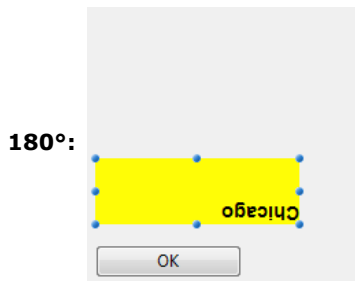
Das gedrehte Objekt wird immer an der linken unteren Ecke ausgerichtet. Hier ein paar Beispiele:

0° (Standard):



90° Rechts:





Rotation über Formulareditor oder 4D Befehl

Bei Rotation im Formulareditor werden Objekt und Text zusammen gedreht. Bei Rotation über den Befehl **OBJECT SET TEXT ORIENTATION** ändert sich nur die Position des Textes, und nicht das Objekt, in dem der Text liegt. Folgendes Beispiel zeigt den Unterschied für die Rotation 90° Links:

90° Links (Eigenschaftensliste) 90° Links (Programmiersprache)

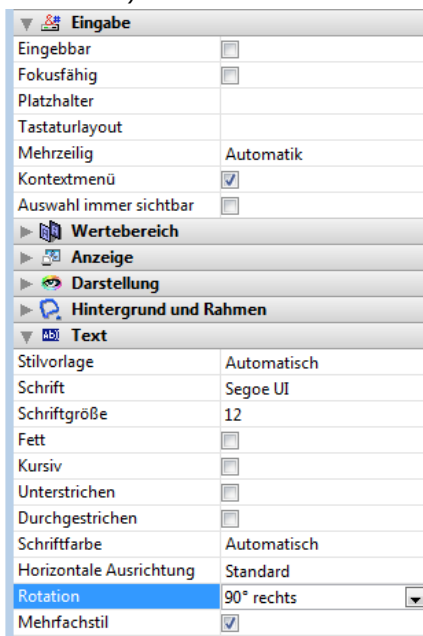


Weitere Informationen dazu finden Sie unter dem Befehl **OBJECT SET TEXT ORIENTATION**.

Rotation von Feldern und Variablen

Die Rotation ist nur für Textobjekte möglich, die weder eingebbar und noch fokusfähig sind

Wählen Sie für ein Objekt vom Typ Feld oder Variable in der Eigenschaftensliste im DropDown-Menü "Rotation" eine Option (außer 0°), werden die Eigenschaften **Eingebbar** und **Fokusfähig** für dieses Objekt automatisch deaktiviert (falls erforderlich):

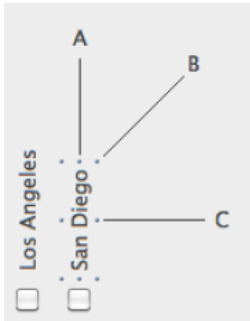


Das Objekt wird dann von der Eingabereihenfolge ausgeschlossen, sein Hintergrund wird standardmäßig transparent. Markieren Sie umgekehrt für ein gedrehtes Objekt die Eigenschaft **Eingebbar** bzw. **Fokusfähig**, wird die Rotation automatisch auf 0° zurückgestellt.

Ein gedrehtes Objekt ändern

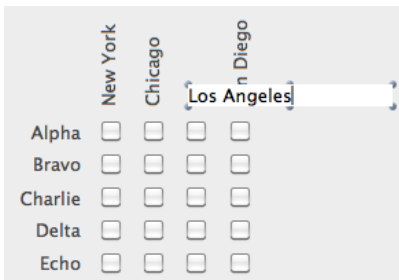
Bei gedrehten Texten lassen sich weiterhin die Eigenschaften verändern und über die Auswahlgriffe oder 4D Befehle wie **OBJECT SET COORDINATES** eine andere Größe bzw. Position setzen.

Beachten Sie, dass Höhe und Breite des Textbereichs nicht von der Rotation abhängen:



- Bei Anpassung in Richtung A ändert sich die Höhe
- Bei Anpassung in Richtung C ändert sich die Breite
- Bei Anpassung in Richtung B ändern sich Breite und Höhe

Sie können den Inhalt eines Bereichs auch im Formulareditor ändern. Dabei wird der Text vor dem Wechseln in den Bearbeitungsmodus in die standardmäßige Rotation zurückgesetzt:



Überblick

In Formularen können Sie zwei Arten von statischen Bildern verwenden: Unabhängige Bilder, die direkt mit dem Formular gespeichert werden oder Bildreferenzen, die außerhalb des Formulars gespeichert werden. Zur besseren Optimierung sollten Sie Bilder generell als Referenz einfügen, wenn diese an mehreren Stellen vorkommen, denn dann wird das Bild nur einmal gespeichert. Außerdem gibt es eine Verbindung zwischen Bild und Referenz, so dass beim Ändern des Ursprungsbildes alle Vorkommen in der Anwendung automatisch geändert werden.

In 4D sind Bilder, die per Referenz eingefügt werden vom Typ „Bild aus Bibliothek“ und unabhängige Bilder vom Typ „statisches Bild“ (sie erscheinen in der Eigenschaftsliste im Feld Typ). Um ein Bild per Referenz einzufügen, müssen Sie entweder die Bildbibliothek benutzen oder ein im Ordner **Resources** gespeichertes Bild per Drag and Drop einsetzen. Wenn Sie ein Bild auf der Seite 0 eines mehrseitigen Formulars einfügen, erscheint es automatisch als Hintergrundelement auf allen Seiten. Sie können das Bild auch in ein vererbtes Formular einfügen, das als Hintergrund für verschiedene andere Formulare dient. Das beschleunigt die Operationen in Ihrer Anwendung.

Sie können statischen Bildern für Formulare Eigenschaften zuweisen. Die Eigenschaft *Wiederholte Anzeige* dient z.B. dazu, um ein kleines Bild als Hintergrundbild des Formulars zu setzen. Es wird dann so oft wiederholt, bis die Seite des Formulars ausgefüllt ist.

Statisches Bild einfügen

Es gibt drei Möglichkeiten, um statische Bilder in Ihre Formulare einzufügen: Sie können ein Bild aus der Zwischenablage übernehmen, per Drag and Drop direkt oder aus der Bildbibliothek einsetzen.

Einfügen über Kopieren oder Drag and Drop

Um ein statisches Bild in ein Formular zu kopieren oder zu ziehen:

1. Kopieren Sie ein Bild in die Zwischenablage und übertragen es in das Formular.
ODER
Wählen Sie ein Bild in einer externen Anwendung (Bilddatei in einem Systemfenster, Web Browser, andere Anwendung, etc.) und fügen es per Drag and Drop in das Formular ein.

Das Originalbild muss in einem Format gespeichert sein, das 4D nativ verwaltet. 4D erkennt alle gängigen Bildformate wie JPEG, PNG, BMP, SVG, GIF, etc.. Es wird automatisch in seinem native Format in den Zielbereich übertragen.

Hinweise:

- Das funktioniert auch für Bilder der Bildbibliothek; für Bilder, die anderen Objekten zugeordnet sind, z.B. eine 3D Bildschaltfläche; und für Felder oder Variablen im Anwendungsmodus.
 - Sie können eine Bildreferenz einfügen, wenn Sie eine Datei aus dem Ordner **Resources** der Datenbank per Drag and Drop in das Formular setzen (siehe nachfolgenden Abschnitt "Automatisches Verweisen auf Bilder").
2. Positionieren Sie das Bild und legen bei Bedarf spezifische Eigenschaften fest.

Bild über Bildbibliothek einsetzen

Ausführliche Informationen zur Bildbibliothek finden Sie im Kapitel **Bildbibliothek**.

Um ein Bild aus der Bildbibliothek in ein Formular zu setzen:

1. Öffnen Sie die Bildbibliothek über die Toolbox und klicken auf den Namen des gewünschten Bildes.
2. Ziehen Sie es aus der Bildbibliothek in das Formular.
 - Ist das per Drag&Drop eingesetzte Bild in Spalten oder Zeilen gesplittet (thumbnails), wird es automatisch als Bildschaltfläche eingesetzt bzw. als Bild PopUp-Menü, wenn Sie es mit gedrückter **Umschalttaste** ziehen. Wollen Sie es als statisches Bild einsetzen, ziehen Sie es unter Windows bei gedrückter **Alt-Taste**, auf Macintosh bei gedrückter **Wahltaste**.
 - Das eingefügte Bild behält standardmäßig die Verbindung zum Originalbild in der Bildbibliothek (Einfügen per Referenz). Wollen Sie diese Verbindung nicht aufrechterhalten, müssen Sie über die Eigenschaftsliste den Typ „statisches Bild“ zuweisen (siehe nächster Absatz). Die Eigenschaften von statischen Bildern und Bibliotheksbildern sind in der Regel identisch.
3. Positionieren Sie das Bild und legen bei Bedarf spezifische Eigenschaften fest.

Das Bild hat wie andere Objekte im Formular bestimmte Standardeigenschaften, die Sie bei Bedarf ändern können.

Eingefügtes Bild von Ursprungsbild unabhängig machen

Fügen Sie in ein Formular ein Bild aus der Bildbibliothek ein, fügen Sie eine Referenz auf das Bild ein. Die Eigenschaftsliste gibt unter Objekte als Typ *Bild aus Bildbibliothek* an. Wird das Bild in der Bildbibliothek geändert, ändert es sich demnach auch im Formular. Sie können das eingefügte Bild aber auch vom Ursprungsbild in der Bildbibliothek unabhängig machen.

Um ein Bild von seiner Quelle in der Bildbibliothek zu lösen:

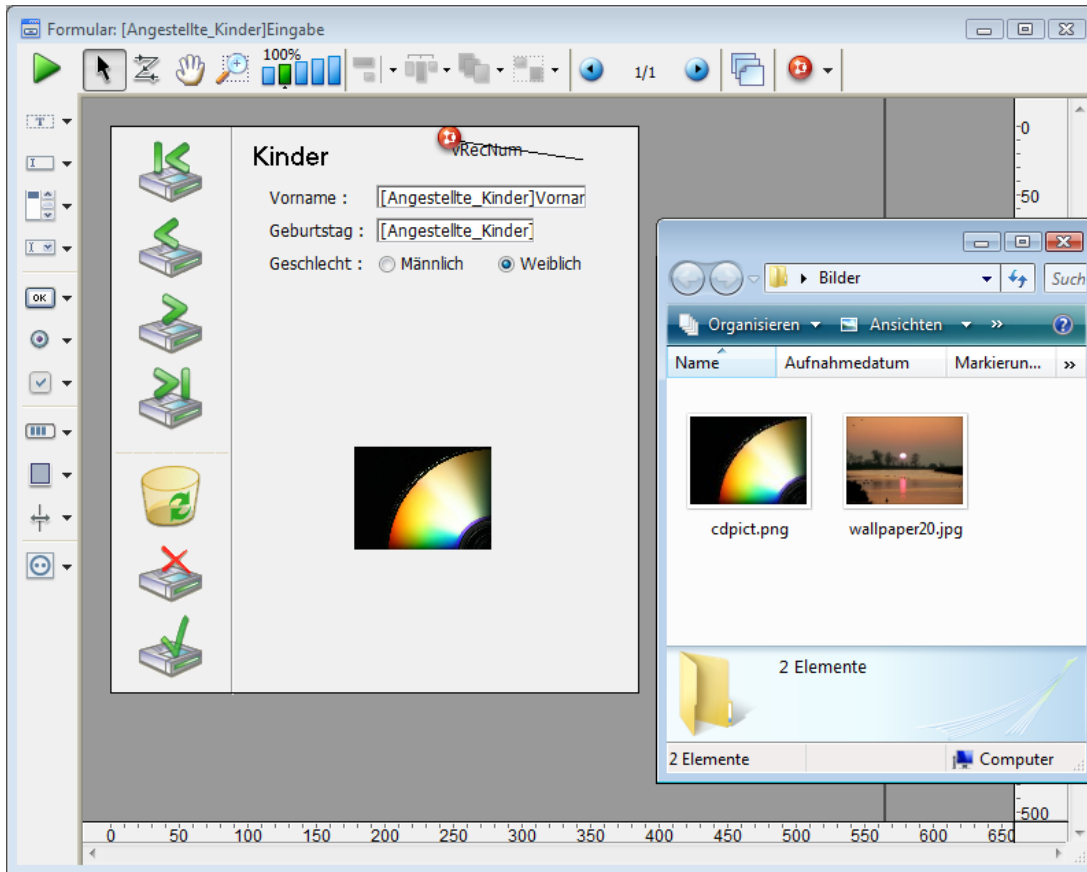
1. Zeigen Sie in der Eigenschaftsliste die Bildeigenschaften an
2. Klicken Sie in der Gruppe **Objekte** auf die Zeile **Typ**.
Es gibt zwei Typen: Von der Bildbibliothek (Standardeinstellung) und statisches Bild.
3. Wählen Sie die Option **statisches Bild**
Das Bild wird nun genauso behandelt wie ein Bild, das aus der Zwischenablage oder per Drag and Drop in das Formular eingefügt wurde.

Automatisches Verweisen auf Bilder

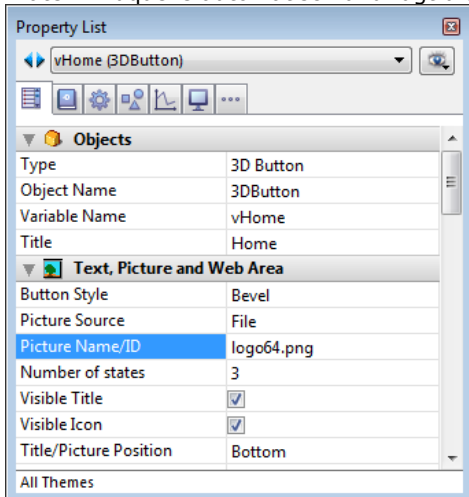
Sie können im Ordner **Resources** der Datenbank statische Bilder speichern, die in Ihren Formularen verwendet werden und mit diesen über Referenz arbeiten. Das optimiert die Anzeige von Bildern und vereinfacht entscheidend ihre Verwaltung.

Hinweis: Weitere Informationen zum Ordner **Resources** finden Sie unter **Beschreibung der 4D Dateien**.

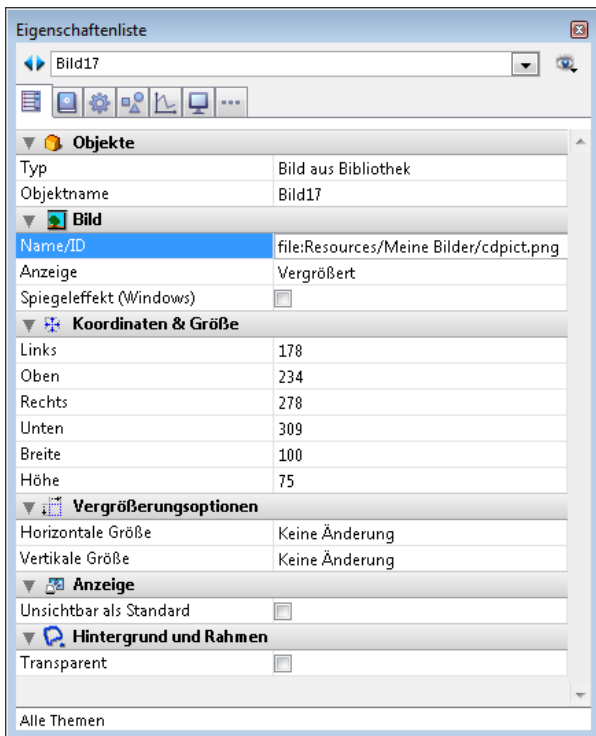
Sie können insbesondere auf Bilder vom Typ .png (Bitmap) oder .svg (Vektor) verweisen. Diese Bilder können Sie einfach per Drag&Drop aus dem Ordner **Resources** in das Formular ziehen:



- Ziehen Sie das Bild auf ein dynamisches Objekt (Bildschaltfläche, 3D Schaltfläche, Bild PopUp-Menü, etc.), setzt 4D die "Datei" Bildquelle automatisch und fügt die Bildreferenz in das Formular in Form von "**{Pfadname+}Dateiname**" ein.



- Ziehen Sie das Bild auf einen leeren Bereich des Formulars, wird ein Bibliotheksbild erstellt und seine Referenz hat die Form "**Datei:{Pfadname+}Dateiname**".



Sie können Bilder im Ordner **Resources** auch in Unterordner legen. Für Bilder in verschiedenen Sprachen können Sie die Funktionsweise des Ordners *lproj* verwenden. Weitere Informationen dazu finden Sie unter **Statische Bilder verwenden**.

Anzeigemodus für statisches Bild setzen

Sie können die Art der Anzeige für ein statisches Bild oder ein Bild aus der Bildbibliothek festlegen, das in ein Formular gesetzt wird. Die Anzeige bestimmt, wie ein Bild neu gezeichnet werden muss, wenn der dazugehörige Bereich in der Größe verändert wird.

Wählen Sie dazu das Bild aus und dann in der Eigenschaftenliste unter der Gruppe "Bild" im DropDown-Menü **Anzeige** eine der folgenden Optionen:

- **Vergrößert** (Standardeinstellung): Wird der Bereich verändert, wird das Bild entsprechend angepasst, so es vollständig sichtbar bleibt.
- **Abgeschnitten**: Wird der Bereich verändert, behält das Bild seine Proportionen und ändert nur seinen Umriss. Das Bild bleibt immer in der Mitte des Bereichs. Wird die Anzeige stärker als das Originalbild verkleinert, wird es abgeschnitten.
- **Wiederholt**: Wird der Bereich vergrößert, wird das Bild sooft wiederholt, bis es den Bereich ganz ausfüllt. Diese Art eignet sich besonders für Hintergrundbilder in Formularen, da sie nur wenig Speicherplatz beansprucht.

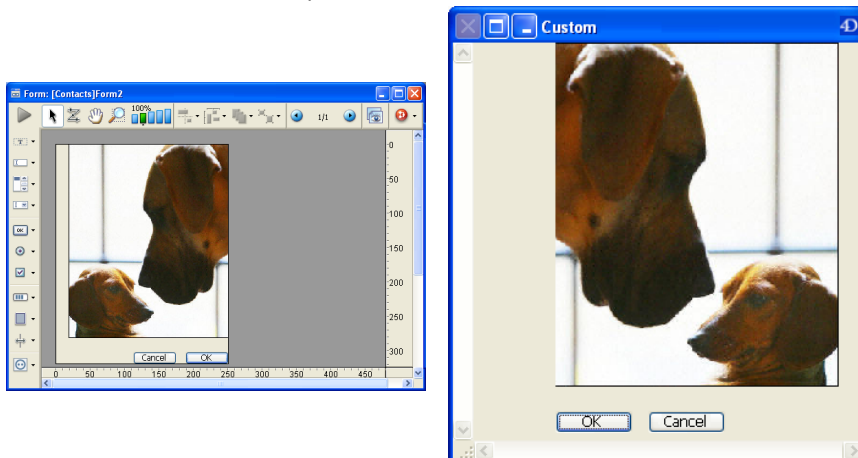
Wird das Bild stärker verkleinert als das Originalbild, wird es abgeschnitten (nicht zentriert).

Spiegeleffekt (Windows)

Die Option **Spiegeleffekt (Windows)** finden Sie in der Eigenschaftenliste für Formulare unter der Gruppe „Bild“.

Diese Option lässt sich in Anwendungen im Modus "rechts-nach-links" mit invertierter Oberfläche verwenden. Weitere Informationen dazu finden Sie im Abschnitt .

Standardmäßig kehrt der Modus "rechts-nach-links" den Inhalt von Bildern in Formularen nicht um. Manchmal ist es jedoch notwendig, bestimmte Bilder umzukehren. Das gilt für Bilder auf der Oberfläche, z.B. Pfeile, die mit der Position von Objekten verbunden sind. Ist diese Option markiert, wird das Bild von rechts-nach-links gespiegelt:



Diese Option ist verfügbar für Bilder aus der Bildbibliothek, statische Bilder, sowie aktive Objekte, die Bilder verwenden (siehe **Spiegeleffekt (Windows)**).

Sie wird nur berücksichtigt:

- Im Anwendungsmodus
- Unter Windows,
- Wenn der Modus "rechts-nach-links" für das Formular aktiviert wurde

Referenzen in statischem Text verwenden

Sie können in Ihre Formulare dynamische Tabellen- und Feldnamen als statischen Text einfügen. Haben Ihre Formulare dynamische Tabellen- oder Feldnamen, werden Sie in Ihrer Anwendung automatisch aktualisiert.

Sie können diese dynamischen Bezeichnungen in folgende Arten von statischem Text einfügen:

- Statische Textbereiche selbst (Objekt vom Typ Text oder Gruppenbox)
- Namen von Formularfenstern
- **Hilfetipps** und Hilfmeldungen (siehe **Eingabekontrollen**).

Hinweis: Dynamische Referenzen, die auf der XLIFF (XML) Architektur basieren, eignen sich besonders für Übersetzung der Oberfläche. Sie lassen sich für diese Bezeichnungen, sowie als Bezeichnungen für Menüs oder Schaltflächen verwenden. Weitere Informationen dazu finden Sie im **PICTURE TO BLOB**.

Im Formulareditor können Sie den Namen oder die Referenz einer Bezeichnung anzeigen. Wählen Sie dazu im Menü **Objekt > Name anzeigen** oder **Ressource anzeigen**.

Werte für Felder oder Variablen

Sie können in statische Textbereiche Feldnamen oder Variablennamen einbinden. Wird der Textbereich angezeigt oder gedruckt, werden die Werte aus dem aktuellen Datensatz eingesetzt. Verwenden Sie eingebundene Felder und Variablen für Serienbriefe und in Kopf- und Fußteilen von Berichten.

Um einen Feld- oder Variablennamen einzubinden, setzen Sie ihn zwischen die Symbole **< >**.

- Sie können hier ein Datenfeld aus einer beliebigen Tabelle der Datenbank einsetzen. Datenfelder aus der Haupttabelle benötigen keine Angabe der Tabelle; diese können Sie eingeben als **<FeldName>**. Datenfelder aus anderen Tabellen müssen den Tabellennamen enthalten; diese geben Sie ein als **<[TabelleName]FeldName>**. Wenn das Formular gedruckt wird, wird das Element **<FeldName>** im Textbereich durch die Information aus dem Feld des jeweiligen Datensatzes ersetzt.
- Eine Variable muss eingefügt werden als **<VariableName>**. Stellen Sie sicher, dass der Wert jeder Variablen über eine Objekt- oder Formularymethode zugewiesen wird.

Sie können festlegen, wie ein eingebundenes Datenfeld bzw. eine Variable dargestellt wird. Dazu setzen Sie direkt nach dem Namen des Datenfeldes oder der Variablen ein Semikolon, gefolgt von einem Anzeigeformat. Zum Beispiel berechnet die eingebundene Variable **<vBonus;€###.##0,00>** die Höhe des Bonus für die betreffende Person und zeigt ihn im Format Euro an. Weitere Informationen dazu finden Sie im Abschnitt **Anzeigeformate**.

Ein Beispiel für Serienbriefe mit eingebundenen Feldern und Variablen finden Sie im Abschnitt **Serienbriefe erstellen**.

Auf Matrix duplizieren

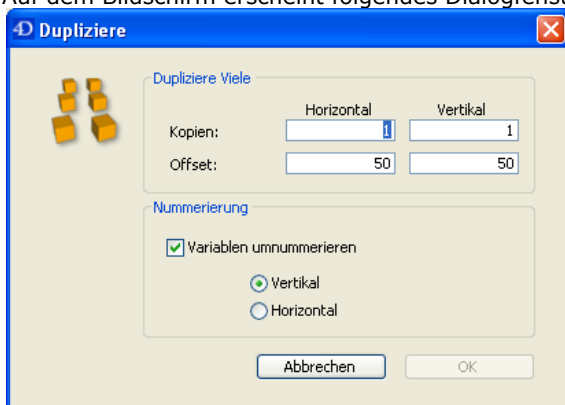
Machmal wollen Sie mehrere ähnliche aktive Objekte gleichzeitig in einem Formular einfügen und sie der Reihe nach durchnummerieren, so dass ihre Namen eindeutig sind. Sie wollen zum Beispiel eine Reihe von Schaltflächen erzeugen, die Datenbankoperationen durchführen. Duplizieren auf Matrix bietet außerdem eine schnelle und einfache Möglichkeit, um Objekte auf einem Raster einzufügen.

Sie können ein aktives Objekt auf Matrix entweder manuell duplizieren oder über das Dialogfenster *Viele duplizieren*, wo Sie die Matrix rasch füllen können.

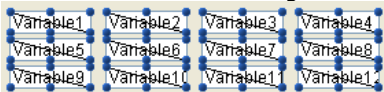
Dialogfenster Dupliziere verwenden

Um ein oder mehrere Objekte über das Dialogfenster *Dupliziere* zu duplizieren, gehen Sie folgendermaßen vor:


1. Wählen Sie das/die gewünschten Objekte aus.
2. Wählen Sie im Menü **Objekt** den Befehl **Viele duplizieren**.
Auf dem Bildschirm erscheint folgendes Dialogfenster:



3. Geben Sie im oberen Bereich die gewünschte Anzahl Objekte in horizontaler und vertikaler Richtung sowie den Abstand dazwischen ein. Weitere Informationen dazu finden Sie im Abschnitt **Formularobjekte einfügen und organisieren**.
4. Aktivieren Sie unter Nummerierung das Kontrollkästchen "Variablen unnummerieren". Diese Option ist nur aktiv, wenn das ausgewählte Objekt eine Variable ist.
5. Markieren Sie die Richtung der Nummerierung.
Wählen Sie **Horizontal**, nummeriert 4D die aktiven Objekte von links nach rechts und oben nach unten (zeilenweise). Wählen Sie **Vertikal**, nummeriert 4D die aktiven Objekte von oben nach unten und links nach rechts (spaltenweise).
Die Objekte werden nach diesen Einstellungen kopiert und nummeriert.
6. Klicken Sie auf die Schaltfläche **OK**.
Das Raster könnte z.B. folgendermaßen aussehen:

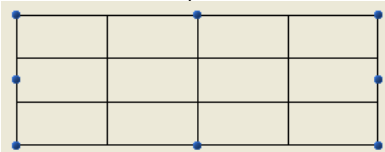


Objekt direkt auf Matrix duplizieren

1. Wählen Sie in der Objektleiste das Tool Matrix  (Variante der Gruppe Viereck) und erstellen ein Raster im Formular.
Stellen Sie sicher, dass es groß genug ist, um die Objekte aufzunehmen, die Sie duplizieren wollen.
2. Geben Sie in der Eigenschaftenliste unter der Gruppe „Raster“ die gewünschte Anzahl Zeilen und Spalten für das Raster ein.



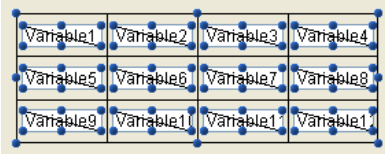
3. Legen Sie eine andere Darstellung für das Raster fest (optional).
Sie können Oberfläche, Rahmenmuster, Farbe, Linienstärke und Füllmuster definieren.
4. Stellen Sie sicher, dass die Zellen des Rasters groß genug sind, um das zu duplizierende Objekt aufzunehmen.



5. Erzeugen Sie ein neues aktives Objekt und setzen es im Raster in die linke obere Zelle
6. Wählen Sie Raster und aktives Objekt gleichzeitig aus.



7. Wählen Sie im Menü **Objekt** den Befehl **Dupliziere auf Matrix**.
4D kopiert das aktive Objekt in jede Zelle im Raster und gibt jedem aktiven Objekt eine einmalige Nummer



Die Objekte werden mit ihrer Größe, den Stileigenschaften und – sofern vorhanden – den zugewiesenen Methoden kopiert.

4D nummeriert die aktiven Objekte von oben nach unten in jeder Spalte durch. Diese Nummern werden dem Objektnamen jedes Objekts hinzugefügt. Somit erhält jede Rasterzelle ein eindeutiges Objekt.

Hinweis: Um die Reihe der aktiven Objekte in jeder Zeile von links nach rechts zu nummerieren, wählen Sie den Menübefehl **Dupliziere auf Matrix** mit gedrückter **Umschalttaste** aus.

Sie können sich jetzt in Methoden auf diese Objekte über ihre zugewiesenen Namen beziehen. Sie können das Raster löschen oder im Formular belassen.

Variablensatz erhöhen

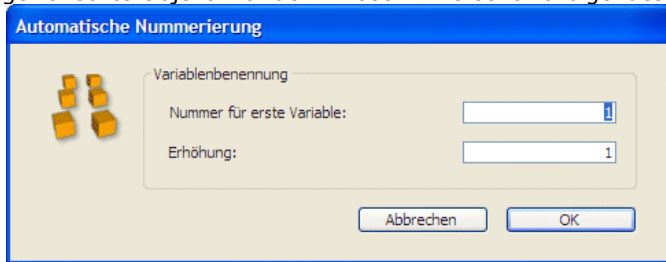
4D enthält ein Tastenkürzel, um Variablengruppen, die in einem Formular erstellt wurden, automatisch zu erhöhen. Damit können Sie z.B. Variablen für Schaltflächen in einem Formular umorganisieren oder sicherstellen, dass jede Variable im Formular einmalig ist. Beachten Sie, dass diese Funktion nur die Namen der zugeordneten Variablen ändert, nicht die Namen der Objekte selbst.

Um einen Variablensatz automatisch zu erhöhen:

1. Wählen Sie alle aktiven Objekte, deren Variable umnummeriert werden soll. Sie können eine beliebige Objektart auswählen.



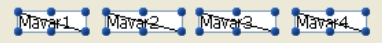
2. Klicken Sie unter Windows mit der Tastenkombination **Strg+Alt**, auf Mac OS mit **Befehl+Wahltaste** auf das gewünschte Objekt. Auf dem Bildschirm erscheint folgendes Dialogfenster:



Hinweis: Dieses Dialogfenster erscheint nicht, wenn die Auswahl ein inaktives Objekt oder Feld enthält.

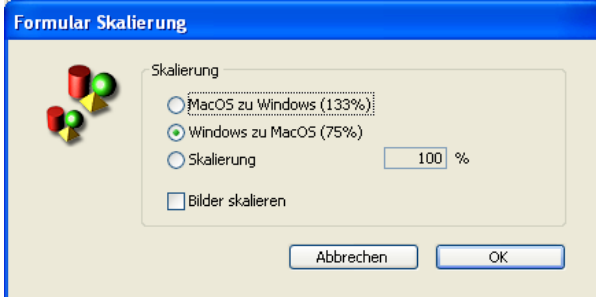
3. Setzen Sie die Startnummer und die gewünschte Erhöhung.
4. Klicken Sie auf die Schaltfläche **OK**.

Die Variablen der gewählten Objekte werden sofort neu nummeriert. Die Nummer wird jeweils am Ende des Variablennamens angehängt. Die Nummerierung läuft von links nach rechts und von oben nach unten.



Formular skalieren

Über das Dialogfenster "Formular Skalierung" können Sie Formularobjekte so skalieren, dass sie gut aussehen, wenn eine Anwendung auf eine andere Plattform übertragen wird.



Formularobjekte, die auf Mac OS erzeugt wurden, werden bei einer Anzeige unter Windows kleiner aussehen und umgekehrt - die Objekte behalten aber ihre eigentliche Größe. Dies liegt daran, dass die Bildschirmauflösung unter Windows etwa 25% höher ist als die Macintosh-Auflösung. So wird ein 12-Punkt-Text auf Mac OS unter Windows als 9-Punkt-Text angezeigt. Wenn die Schriftgröße auf Mac OS gerade groß genug ist, kann sie unter Windows zu klein sein. Umgekehrt kann eine Schriftgröße unter Windows passen und auf Mac OS zu groß sein.

Um Unterschiede der Bildschirmauflösungen auszugleichen, müssen Sie Objekte neu skalieren. Im Menü **Formular** können Sie mit dem Befehl **Skalieren** die Größe aller Formularobjekte in einem Vorgang proportional ändern.

Sie können zwischen folgenden Optionen wählen:

- **Mac OS zu Windows (133%)**: Diese Option ist die Standardoption, wenn Sie 4D auf Macintosh verwenden. Verwenden Sie diese Option, wenn Sie ein Formular in der Größe ändern wollen, das entsprechend der Macintosh-Bildschirmauflösung erzeugt wurde, so dass es aussieht, als ob es entsprechend der Windows-Bildschirmauflösung erzeugt wurde. Bei diesem Vorgang vergrößert das Programm alle Formularobjekte um ein Drittel. Zum Beispiel wird aus einem 9-Punkt Text ein 12-Punkt Text.
- **Windows zu Mac OS (75%)**: Diese Option ist die Standardoption, wenn Sie 4D unter Windows verwenden. Verwenden Sie diese Option, wenn Sie ein Formular in der Größe ändern wollen, das entsprechend der Windows-Bildschirmauflösung erzeugt wurde, so dass es aussieht, als ob es entsprechend der Macintosh-Bildschirmauflösung erzeugt wurde. Bei diesem Vorgang verkleinert das Programm alle Formularobjekte um ein Viertel. Zum Beispiel wird aus einem 12-Punkt Text ein 9-Punkt Text.
- **Skalierung**: Diese Option ermöglicht Ihnen das Ändern einer Formulargröße mit einem Prozentsatz, den Sie im Eingabebereich "%" eingeben. Mit dieser Option können Sie eine Formulargröße ändern, so dass das Formular in jeder ungewöhnlichen Bildschirmauflösung, die Sie auf einer Windows- oder Macintosh-Plattform antreffen können, optimal aussieht. Sie können diese Option zum Ändern der Größe aller Formularobjekte für die Plattform, die Sie einsetzen, verwenden. Sie können z.B. 200% eingeben, wenn Sie die Größe aller Objekte verdoppeln wollen; 50% zum Halbieren der Größe.
- **Bilder skalieren**: Diese Option ist standardmäßig nicht ausgewählt. Normalerweise bringt das Verkleinern oder Vergrößern von Bitmaps aus ästhetischer Sicht keine guten Resultate. Daher ändert das Programm in einem Formular nicht die Größe statischer Bilder, wenn Sie diese Option inaktiv ist. Stattdessen werden sie an ihre neuen "zentriert-relativen" Positionen verschoben. Sie sollten diese Option nur markieren, wenn das Skalieren der Bilder akzeptable Resultate liefert oder wenn die Bilder nicht im Format Bitmap sind.

Objektmethoden verwenden

Sie können jedem aktiven Objekt in einem Formular eine Methode zuweisen. Methoden, die einzelnen Objekten in einem Formular zugewiesen werden, werden *Objektmethoden* genannt.

Hier ein paar gängige Verwendungen für Objektmethoden:

- Beschränkungen bei der Dateneingabe erzwingen
- Oberflächenobjekte wie Registerkarten, PopUp-Menüs, Dropdown-Listen, Combo Boxen, Listboxen, hierarchische Listen und PopUp-Menüs initialisieren und verwalten
- Aktion festlegen, die ausgeführt wird, wenn auf ein Objekt geklickt oder doppelgeklickt wird
- Drag&Drop-Operationen verwalten

Hier ein paar einfache Beispiele, die Operationen an Daten ausführen:

Die folgende Methode berechnet eine Summe anhand von Daten aus zwei anderen Datenfeldern:

```
vGesamt:=[Produkt]Preis * [Positionen]Menge
```

Folgende Methode setzt alle Zeichen in einem Datenfeld *Name* in Großbuchstaben:

```
[Kunden]Name:=Uppercase([Kunden]Name)
```

Die folgende Methode verknüpft Werte in einem Datenfeld *Vorname* mit einem Datenfeld *Nachname* und weist das Ergebnis einer Variablen *vName* zu:

```
vName:=[Angestellte]Vorname+" "+[Angestellte]Nachname
```

Da jede Objektmethode einem Objekt zugeordnet ist, erzeugen Sie Objektmethoden im Formulareditor. Weitere Informationen zur Verwendung von Methoden finden Sie unter **Methoden bearbeiten**.

Objektereignisse

Objektmethoden werden ausgeführt, wenn bestimmte Ereignisse auftreten. Zum Beispiel macht die Aktion, die mit einer Registerkarte verknüpft ist, nur Sinn, wenn ein Benutzer auf eine Registerkarte klickt. In einem rollbaren Bereich möchten Sie vielleicht, dass die Methode nur ausgeführt wird, wenn der Benutzer auf einen Eintrag doppelklickt. In der Eigenschaftsliste unter der Gruppe „Ereignisse“ können Sie festlegen, welches Ereignis für ein bestimmtes Objekt ausgeführt wird. Die Liste zeigt alle möglichen Formularereignisse für das gewählte Objekt an. Manche Ereignisse sind nur für spezifische Objekte verfügbar. Es gibt auch zusätzliche Ereignisse, die nur auf Formularebene erzeugt werden. Sie erscheinen nicht in dieser Liste. Eine ausführliche Beschreibung sämtlicher Ereignisse finden Sie im Handbuch *4D Programmiersprache* unter **Form event**.

Sie aktivieren ein Ereignis durch Anklicken des zugeordneten Kontrollkästchens. Sie können auch alle Ereignisse auf einmal aktivieren/deaktivieren. Klicken Sie dazu unter Windows bei gedrückter **Strg-Taste**, auf Mac OS bei gedrückter **Befehlstaste** auf ein Ereignis.

Müssen Sie je nach Ereignis unterschiedliche Code-Bereiche ausführen, verwenden Sie eine **Case of...Else...End case** Anweisung in Ihrer Methode und prüfen für jedes Ereignis, das Sie auf der Seite Ereignisse markiert haben. Zum Prüfen eines Ereignisses verwenden Sie die Funktion **Form event** und die Konstanten unter dem Thema **Formulareignisse**. Eine Beispiel-Shell für eine Objektmethode könnte folgendermaßen aussehen:

```
Case of
  :(Form event=On Load)
  //Setzen Sie den auszuführenden Code, wenn das Formular hier geöffnet wird.

  :(Form event=On Data Change)
  //Setzen Sie den auszuführenden Code, wenn der Inhalt des Objekts hier geändert wird.

  :(Form event=On Validate)
  //Setzen Sie den auszuführenden Code, wenn das Formular hier bestätigt wird.
End case
```

Objektmethode erstellen

Um eine Objektmethode zu erstellen:

1. Wählen Sie im Formulareditor das Objekt aus, dem Sie eine Objektmethode zuordnen wollen.
2. Klicken Sie in der Eigenschaftsliste unter der Gruppe „Aktion“ neben der Zeile Objektmethode auf die Schaltfläche **Bearbeiten**
ODER

Wählen Sie im Menü **Objekt** den Befehl **Objektmethode**

ODER

Klicken Sie mit der rechten Maustaste in das Objekt und wählen im Kontextmenü den Eintrag **Objektmethode**.

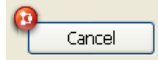
ODER

Drücken Sie unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** und klicken auf das Datenfeld oder Objekt.

4D öffnet ein neues Methodenfenster, das standardmäßig leer ist. Der Methodename enthält das Wort "Methode" und den Namen des Objekts oder Datenfelds.

3. Schreiben Sie die Methode wie unter **Methoden bearbeiten** erläutert und schließen dann das Fenster oder sichern Ihre Arbeit.

Die Methode ist jetzt dem Datenfeld bzw. aktiven Objekt zugeordnet. Sie können für solche Objekte auch die Erkennungsmarke für "Objektmethode" anzeigen:



Sie können eine Methode jederzeit anzeigen oder ändern.

Objektmethode öffnen

Um eine Objektmethode zu öffnen:

1. Wählen Sie im Formulareditor das Objekt aus, dessen Objektmethode Sie öffnen wollen.
2. Klicken Sie in der Eigenschaftenliste unter der Gruppe „Aktion“ neben der Zeile Objektmethode auf die Schaltfläche **Bearbeiten**
ODER
Wählen Sie im Menü **Objekt** den Befehl **Objektmethode**
ODER
Klicken Sie mit der rechten Maustaste in das Objekt und wählen im Kontextmenü den Eintrag **Objektmethode**.
ODER
Drücken Sie unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** und klicken auf das Objekt, dem die Methode zugewiesen ist. Der Methodeneditor erscheint mit Ihrer Methode.

Objektmethode löschen

Sie können eine Objektmethode jederzeit löschen. Wählen Sie hierzu das Objekt mit der zugewiesenen Methode aus und dann im Menü **Objekt** den Eintrag **Objektmethode löschen**.


Formularebenen verwenden

Sie können in 4D Formularen verschiedene Ebenen verwenden. Das macht komplexe Formulare übersichtlicher, denn Sie können Objekte auf mehrere Ebenen verteilen, die Sie je nach Bedarf ein-, ausblenden oder sperren. Sie können Objekte nach Typen anlegen, also z.B. Felder, Variablen, statische Objekte. Alle Objekttypen, inkl. Unterformulare und Plug-In Bereiche lassen sich in Ebenen einfügen.

Wie es funktioniert

Pro Formular gibt es 9 Ebenen, die standardmäßig Ebene 1 bis Ebene 9 lauten (diese Namen sind änderbar). Jede Ebene lässt sich anzeigen, ausblenden oder sperren. Die Verwaltung erfolgt über die Ebenenpalette:



Um die Palette anzuzeigen, klicken Sie in der Werkzeugleiste des Fensters auf das Symbol  oder wählen im Kontextmenü des Editors bzw. im Menü **Formular** den Eintrag **Liste anzeigen**.

Hier ein paar Regeln zum Arbeiten mit Ebenen:

- **Anwendungsbereich:** Ebenen sind ein rein-grafisches Werkzeug, das nur im Formulareditor verwendbar ist; Sie können weder per Programmierung noch in der Anwendungsumgebung auf Ebenen zugreifen.
- **Ebenen und Seiten:** Objekte derselben Ebene können zu verschiedenen Formularseiten gehören. Unabhängig von der Konfiguration der Ebenen lassen sich nur Objekte der aktuellen Seite, und evtl. der Seite 0 anzeigen.
- **Ebenen und Objektebenen:** Ebenen sind unabhängig von Objektebenen, es gibt keine Anzeigehierarchie zwischen verschiedenen Ebenen.
- **Ebenen und Gruppen:** Nur Objekte, die zur aktuellen Ebene gehören, lassen sich gruppieren.

Hinweis: In Standardformularen oder Formularen, die im Formularassistenten erstellt werden, werden der Titel standardmäßig in Ebene 2, die Schaltflächen mit Aktionen (Bildschaltflächen) in Ebene 3 gelegt.

Objekt in Ebene legen

Ein Objekt kann nur einer Ebene angehören. Jedes im Formular angelegte Objekt wird in die aktuelle Ebene gelegt. Standardmäßig ist Ebene 1 gewählt, d.h. alle Objekte werden in Ebene 1 des Formulars gelegt.

Um ein Objekt in einer anderen Ebene zu erstellen, wählen Sie zuerst in der Palette die Zeile mit der gewünschten Ebene aus:

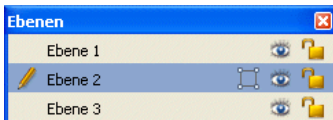


Sie können auch ein bzw. mehrere Objekte von einer Ebene in eine andere verschieben. Dazu wählen Sie das/die Objekt(e), deren Ebene Sie ändern wollen. Die Ebenenliste zeigt die Ebene(n) an, zu der die Auswahl gehört:



Hinweis: Die Auswahl kann Objekte aus verschiedenen Ebenen enthalten.

Dann wählen Sie die Zielebene für die Auswahl und klicken unter der Liste auf die Schaltfläche **Verschieben**. Die Auswahl erscheint dann in der neuen Ebene:

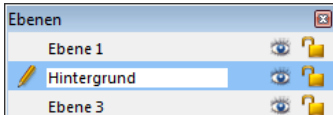


Hinweis: Sie können die Ebene, in der das Objekt gefunden wird, über 4D Erkennungszeichen anzeigen. Weitere Informationen dazu finden Sie im Abschnitt **Erkennungszeichen verwenden**.

Ebenen umbenennen

Sie können jede der 9 Ebenen umbenennen, beispielsweise einen aussagekräftigen Namen wählen, der den Objektkinhalt beschreibt. Sie können in jedem Formular der Datenbank andere Namen für die Ebenen vergeben.

Um eine Ebene umzubennenen, klicken Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** auf den Namen oder doppelklicken Sie auf den Namen der gewählten Ebene. Der Name wird dann eingebbar.



Mit Ebenen arbeiten

Sind alle Objekte in Ebenen angelegt, können Sie über die Ebenenpalette folgendes ausführen:

- Alle Objekte derselben Ebene mit einem Klick auswählen
- Objekte einer Ebene ein-/ausblenden
- Objekte einer Ebene sperren

Alle Objekte derselben Ebene wählen

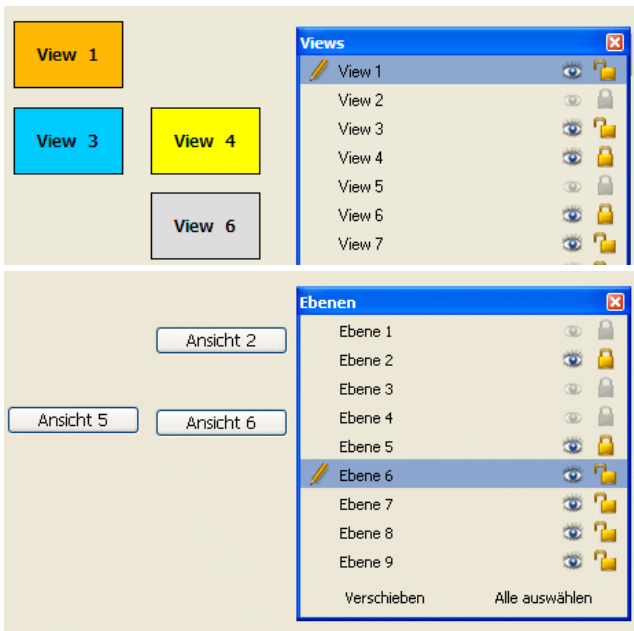
Sie können alle Objekte, die zu einer Ebene gehören, in der aktuellen Seite des Formulars auswählen. Das ist hilfreich für globale Änderungen an den Objekten.

Wählen Sie zuerst die entsprechende Ebene und klicken Sie dann unter der Liste auf die Schaltfläche **Alle auswählen**.

Objekte einer Ebene ein-/ausblenden

Sie können Objekte einer Ebene jederzeit auf der aktuellen Seite des Formulars ein- oder ausblenden. Auf diese Weise können Sie sich beim Bearbeiten des Formulars auf bestimmte Objekte konzentrieren.

Standardmäßig erscheinen alle Ebenen, das erkennen Sie am Icon . Um eine Ebene auszublenden, klicken Sie auf dieses Icon. Die Icons am rechten Rand der Liste sind dann grau unterlegt und die Objekte dieser Ebene erscheinen nicht mehr im Formular:



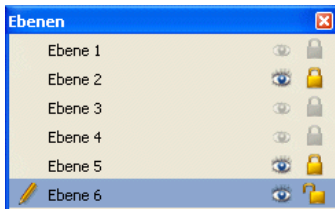
Hinweis: Die aktuelle Ebene, d.h. die in der Liste gewählte Ebene lässt sich nicht ausblenden.

Wollen Sie eine ausgeblendete Ebene wieder anzeigen, klicken Sie darauf bzw. erneut auf das Icon.

Objekte einer Ebene sperren

Sie können Objekte einer Ebene sperren, sie lassen sich dann weder auswählen, verändern oder aus dem Formular löschen. Ein gesperrtes Objekt lässt sich nicht per Mausclick, über ein Rechteck bzw. den Befehl **Wähle ähnliche Objekte** aus dem Kontextmenü wählen. Auf diese Weise vermeiden Sie Bearbeitungsfehler.

Standardmäßig sind alle Ebenen nicht gesperrt, das erkennen Sie am Icon in der Ebenenpalette. Wollen Sie Objekte einer Ebene sperren, klicken Sie auf dieses Icon. Das Schloss schnappt zu, d.h. die Ebene ist jetzt gesperrt:



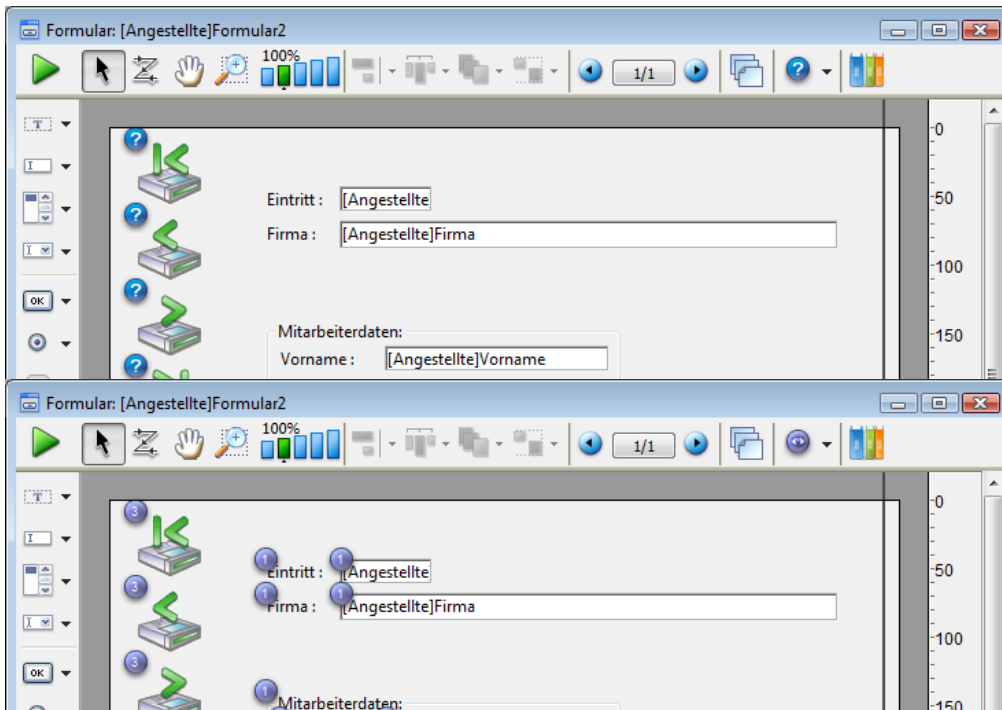
Hinweis: Die aktuelle Ebene, d.h. die in der Liste gewählte Ebene lässt sich nicht sperren. Um eine Ebene zu entsperren, klicken Sie auf die Ebene oder erneut auf das Schloss.

Erkennungszeichen verwenden

Sie können im Formulareditor zum Anzeigen von Objekteigenschaften Erkennungszeichen verwenden.

Diese Funktionalität arbeitet folgendermaßen: Jedem Erkennungszeichen ist eine Eigenschaft zugeordnet. So gibt das Erkennungszeichen Tipps an, dass dem Objekt ein Tipp zugewiesen ist. Aktivieren Sie ein Erkennungszeichen, zeigt 4D in der linken oberen Ecke der Objekte im Formular, die eine Eigenschaft haben, ein kleines Icon (Erkennungszeichen).

- Für Eigenschaften vom Typ "wahr/falsch", z.B. Tipps, erscheint das Erkennungszeichen nur, wenn das Objekt diese Eigenschaft hat.
- Für Eigenschaften mit einem Wert, z.B. Nummer der Ansicht, erscheint das Erkennungszeichen mit dem Wert.














Um ein Erkennungszeichen zu aktivieren, klicken Sie in der Werkzeugleiste auf das Symbol für Erkennungszeichen, bis der gewünschte Typ erscheint. Sie können aber auch über den Pfeil das DropDown-Menü aufklappen und den Typ direkt



Wollen Sie die Erkennungszeichen nicht anzeigen, wählen Sie im DropDown-Menü den letzten Eintrag **Keine Anzeige**.

Hinweis: In den 4D Einstellungen können Sie auf der [Seite Formulare](#) festlegen, welches Erkennungszeichen standardmäßig erscheinen soll.


Es gibt folgende Typen für Erkennungszeichen:

Icon	Name	Wird angezeigt ...
	Objektmethode	Für Objekte mit zugewiesener Objektmethode
	Standard Aktion	Für Objekte mit zugewiesener Standardaktion
	Größe anpassen	Für Objekte mit mindestens einer Größenoption, zeigt die aktuelle Kombination der Eigenschaften
	Eingabefolge	Für eingebbare Objekte, gibt die Nummer der Eingabefolge an
	Ebene Nummer	Für alle Objekte, gibt die Nummer der Ebene an
	Stilvorlage	Für Objekte mit einer zugeordneten Stilvorlage
	Filter	Für eingebbare Objekte mit einem zugewiesenen Eingabefilter
	Help Tip	Für Objekte mit einem zugeordneten Tipp
	Localized	Für Objekte, deren Bezeichnung von einer Referenz kommt, d.h. mit vorangestelltem „:“. Die Referenz kann vom Typ Ressource (STR#) oder XLIFF sein.
	Dynamische Referenz	Für Objekte, die eine dynamische Referenz auf ein Datenfeld, eine Tabelle oder Variable enthalten (Syntax vom Typ "<label>")
	Keine Anzeige	Es erscheinen keine Erkennungszeichen

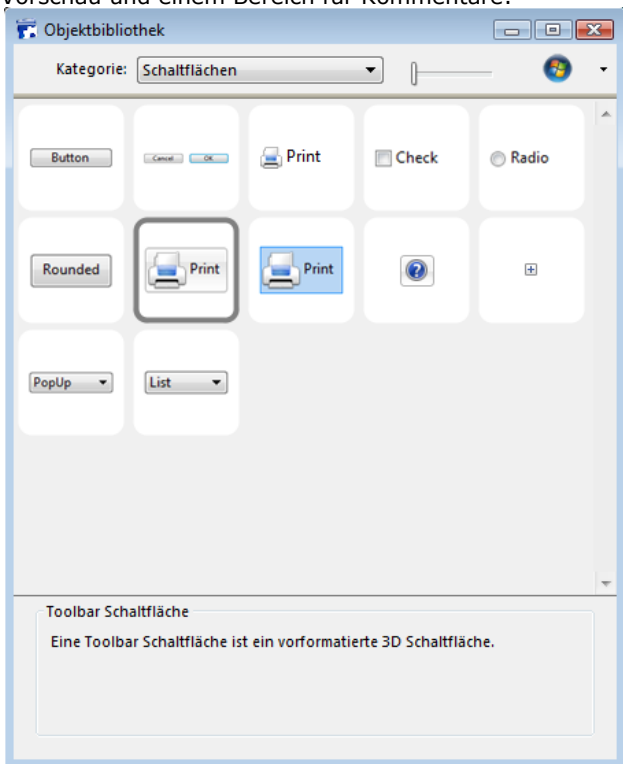
Vorkonfigurierte Objektbibliothek verwenden

Die vorkonfigurierte Objektbibliothek ist ein Werkzeug, um das Hinzufügen von Objekten in 4D Formularen zu vereinfachen. Sie enthält eine Sammlung vorkonfigurierter Objekte, die Sie in Ihren Formularen durch einfaches Drag-and-Drop verwenden können.

Die Bibliothek verwenden

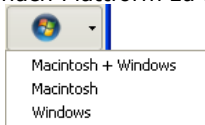
Die vorkonfigurierte Objektbibliothek erscheint als eigenes Fenster. Objekte lassen sich per Drag-and-Drop in den Formulareditor einfügen. Um die vorkonfigurierte Objektbibliothek anzuzeigen, klicken Sie im 4D Formulareditor in der Werkzeugleiste auf die letzte Schaltfläche .

Die vorkonfigurierte Objektbibliothek erscheint dann in einem neuen Fenster. Es besteht aus einem PopUp-Menü, einer Vorschau und einem Bereich für Kommentare:



- **Kategorie:** Die Objekte sind in verschiedene Kategorien gegliedert, z.B. Schaltflächen, Eingabebereiche. Wählen Sie im PopUp-Menü am oberen Rand eine Kategorie, um die Auswahl einzuschränken oder **Alle Kategorien**, um alle Objekte anzuzeigen.

Bestimmte Objekte sind nur für eine Plattform verfügbar (Windows oder Macintosh). Um die angezeigten Objekte nach Plattform zu filtern, wählen Sie rechts oben den passenden Eintrag:



- **Vorschau:** Der zentrale Bereich zeigt eine Vorschau und den Namen der Objekte. Klicken Sie auf das Objekt, erscheint im unteren Teil die Beschreibung dazu. Sie können die Größe der Ansichten im Fenster verändern. Ziehen Sie dazu den Schieber am oberen Rand nach rechts bzw. nach links.
- **Kommentar:** Zeigt Information über das gewählte Objekt an.

Objekte lassen sich einfach per Drag-and-Drop aus den zentralen Bereichen des Fensters in ein Formular einfügen. Das Objekt wird mit seinen vordefinierten Eigenschaften eingefügt. Sie können es dann an Ihre eigenen Bedürfnisse anpassen.

Enthaltene Objekte

Diese Bibliothek verwendet ausschließlich standardmäßige 4D Objekte, wie Schaltflächen, Textbereiche, etc. mit bestimmten vordefinierten Eigenschaften, um das Setzen in Formularen zu beschleunigen und zu vereinfachen. Das Objekt "Kennwort Eingabebereich" ist z.B. eine Textvariable mit einer spezifischen Stilvorlage.

- Ausführliche Informationen zu den verschiedenen Objekten der Bibliothek finden Sie im Kapitel **Objekte der Bibliothek**.

- Komplexe Objekte wie Widgets für Datum und Zeit mit spezifischem API werden im Handbuch **4D Widgets** beschrieben.

Im Gegensatz zu Objektbibliotheken von Benutzern (siehe **Eigene Objektbibliothek anlegen und einsetzen**) lässt sich die vorkonfigurierte Objektbibliothek in 4D nicht verändern, d.h. Sie können weder Objekte daraus entfernen, noch neue hinzufügen.

Eigene Objektbibliothek anlegen und einsetzen

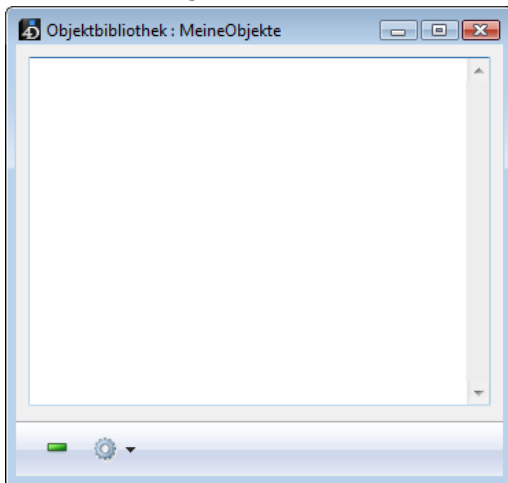
In 4D können Sie Objektbibliotheken erstellen und einsetzen. Eine Objektbibliothek ist eine von 4D angelegte externe Datei. Hier können Sie alle Arten von Objekten in Formularen speichern: Schaltflächen, Texte, Bilder, hierarchische Listen, etc. Sie können diese Objekte dann in verschiedenen Formularen verwenden. Objekte werden mit all ihren Eigenschaften gespeichert, d.h. mit den dazugehörigen Objektmethoden. Bibliotheken werden zusammengelegt, ihr Inhalt lässt sich per Drag&Drop oder Kopieren-Einsetzen verwenden. Sie funktionieren wie eine dauerhafte Zwischenablage. In Bibliotheken können Sie Hintergrundobjekte für Formulare in Bibliotheken nach Grafikfamilien, Verhalten, etc. gruppieren. Da Bibliotheken als externe Datei gespeichert werden, lassen sie sich mühelos mit verschiedenen Anwendungen verwenden.

Sie können Objektbibliotheken auch mit vom Benutzer editierbaren Formularen kombinieren, um Objekte zur eigenen Gestaltung von Anwendungen zu liefern. Weitere Informationen dazu finden Sie im Abschnitt [Benutzerformulare](#).

Objektbibliothek erstellen

Um eine neue Objektbibliothek einzurichten, wählen Sie im Menü **Datei/Ablage** den Befehl **Neu>Objektbibliothek**. Ein Standard-Sichern Dialog erscheint, in dem Sie Name und Speicherort der Objektbibliothek wählen.

Bestätigen Sie das Dialogfenster, erstellt 4D auf Ihrer Festplatte eine neue Objektbibliothek und zeigt ihr Fenster an. Es ist standardmäßig leer:



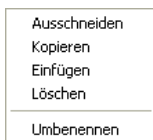
Sie können pro Anwendung beliebig viele Bibliotheken erstellen. Eine auf Mac OS erstellte Anwendung lässt sich unter Windows verwenden und umgekehrt.

Objektbibliothek aufbauen

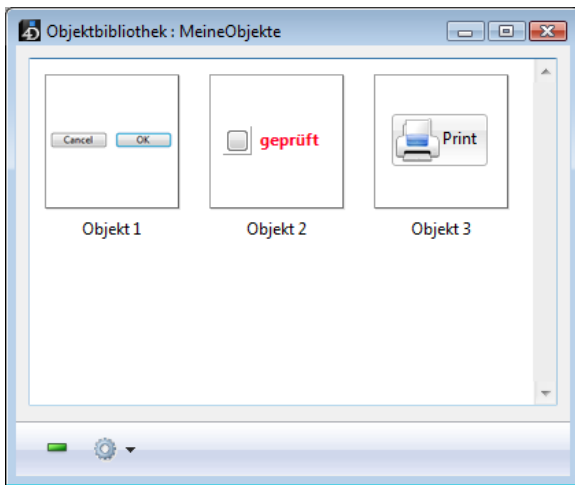
Sie setzen Objekte per Drag&Drop oder Kopieren-Einsetzen in die Objektbibliothek. Diese können aus einem Formular oder einer anderen Objektbibliothek stammen (inkl. **Vorkonfigurierte Objektbibliothek verwenden**). Die Verbindung zum Originalobjekt bleibt nicht erhalten. Wird es geändert, wirkt sich die Änderung nicht auf das in die Bibliothek kopierte Objekt aus.

Hinweis: Damit Sie Objekte per Drag&Drop von Formularen in Objektbibliotheken legen können, müssen Sie in den 4D Einstellungen auf der [Seite Formulare](#) die Option „Starte Drag & Drop“ aktivieren.

Gängige Operationen wie Hinzufügen, Löschen, Umbenennen sind über das Kontextmenü oder das Menü **Optionen** am unteren Rand des Fensters verfügbar:



Sie können in die Bibliothek einzelne Objekte oder Objektsätze legen. Jedes Objekt bzw. jeder Objektsatz zählt als ein Eintrag:



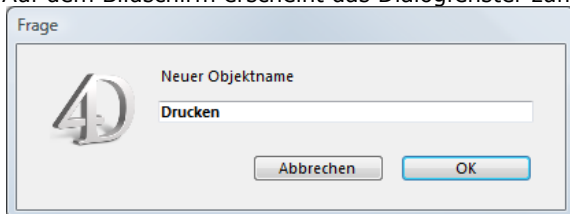
Eine Objektbibliothek kann bis zu 32.000 Einträge haben.

Objekte werden mit allen Eigenschaften kopiert, also mit grafischen Elementen und Funktionalitäten. Diese werden auch beim Kopieren in ein Formular oder eine andere Bibliothek beibehalten.

Objekt umbenennen

Jeder neue Eintrag hat den Namen "Objekt", gefolgt von einer Nummer, z.B. *Objekt2012*. Sie können diesen Eintrag nach Belieben umbenennen. Dazu gehen Sie folgendermaßen vor:

- Doppelklicken Sie auf den gewünschten Eintrag
- Klicken Sie mit der rechten Maustaste auf den Eintrag und wählen dann im Kontextmenü den Befehl **Umbenennen**. Auf dem Bildschirm erscheint das Dialogfenster zum Umbenennen. Tragen Sie den gewünschten Namen ein:



Mehrere Einträge können auch denselben Namen haben.

Abhängige Objekte

Beim Übertragen von Objekten in die Bibliothek werden damit verbundene Objekte mitkopiert. Kopieren Sie z.B. eine Schaltfläche, wird die zugeordnete Objektmethode ebenfalls übertragen. Solche Objekte lassen sich nur direkt oder per Drag&Drop kopieren.

Im folgenden sehen Sie eine Liste der abhängigen Objekte, die zusammen mit dem Hauptobjekt in die Bibliothek übertragen werden:

- Listen
- Stilvorlagen
- Formate/Filter
- Bilder
- Hilfetipps (mit Feld verknüpft)
- STR# Ressourcen
- Objektmethoden

Bei STR# Ressourcen wird die gesamte STR# kopiert und nicht nur der dem Objekt zugeordnete String.

Objektbibliothek öffnen

Um eine vorhandene Objektbibliothek zu öffnen, wählen Sie im Menü **Datei/Ablage** oder in der Werkzeugleiste den Befehl **Öffnen>Objektbibliothek**.

Auf dem Bildschirm erscheint ein Standard-Öffnen Dialog, in dem Sie die gewünschte Objektbibliothek wählen können. Eine bestimmte Bibliothek lässt sich immer nur für jeweils eine Datenbank öffnen. Dagegen lassen sich mehrere verschiedene Bibliotheken in derselben Datenbank öffnen. Die Endung für Bibliotheken ist ".4il".

Auf Objektbibliotheken zugreifen

Sie können auf Objektbibliotheken über die Designumgebung von 4D zugreifen und, wenn der Entwickler es zulässt, auch im Anwendungsmodus über den Formulareditor für Benutzer (siehe **Benutzerformulare**). In der Designumgebung sind alle Einträge der Bibliothek sichtbar. Im Formulareditor für Benutzer sind dagegen nur Einträge sichtbar, denen keine Objektmethode zugeordnet ist.

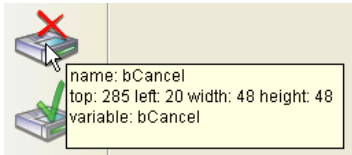
Einträge mit mindestens einer Objektmethode erscheinen in blauer Schrift und sind mit einem Erkennungszeichen versehen. Es ist dasselbe wie bei Objekten mit zugewiesener Objektmethode in einem Formular (siehe **Erkennungszeichen verwenden**). So können Sie leicht unterscheiden zwischen Einträgen in der Designumgebung, die nur für den Entwickler sichtbar sind und Einträgen in der Anwendungsumgebung, die auch für den Benutzer sichtbar sind.

Funktionsweise im Remote Modus

Damit alle Client-Rechner auf eine Objektbibliothek zugreifen können, muss die Bibliothek im Ordner **Resources** der Datenbank liegen (siehe **Beschreibung der 4D Dateien**). Dieser Ordner ermöglicht die gemeinsame Nutzung eigener Elemente durch Server und Client-Rechner, wie Bilder, XLIFF-Dateien, etc. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Ordner Resources verwalten**. Sie können den Inhalt des Ordners **Resources** über den **Ressourcen Explorer** ansehen und verwalten.

Information über Objekte in ausgeführten Formularen

Wird ein Formular ausgeführt, können Sie über Tastenkürzel Informationen über die enthaltenen Objekte erhalten, z.B. Name, Koordinaten, o.ä. Die Information erscheint als Hilfetipp, wenn Sie unter Windows die Tastenkombination **Strg+Umschalttaste**, auf Mac OS **Befehl+Umschalttaste** drücken und den Cursor über ein Objekt ziehen.



Diese Information ist bei geöffneter Designumgebung für jedes Objekt im Formular verfügbar.

Dynamische Formulare

Überblick

Dynamische Formulare sind Formulare, deren Strukturen entweder in einer .json Datei oder in einem 4D Objekt definiert sind. Sie werden bei jedem Aufruf neu generiert und bieten so mehr Flexibilität. Einige bedeutsame Vorteile sind:

- Usability (Nutzbarkeit, Benutzerfreundlichkeit)
 - unmittelbare Updates
 - dynamische Änderungen durch Benutzer
 - leichte Wiederverwendung
 - schnelleres Suchen
- Portability (Übertragbarkeit, Mobilität)
 - einfache gemeinsame Nutzung
 - lässt sich in Source Control Systemen speichern

Jedes Objekt in einem dynamischen Formular, inkl. das Formular selbst, hat eine oder mehrere Eigenschaften. Weitere Informationen dazu finden Sie unter [Seitenverzeichnis](#).

Beispiel Hello World

Das Beispiel zeigt eine einfache JSON Datei ("HW.json", gespeichert im Ordner *Resources*) mit Text, einem Bild und einer Schaltfläche:

```
{  "windowTitle": "Hello World",  "windowMinWidth": 220,  "windowMinHeight": 80,  "method": "HWexample",  "pages": [    null,    {      "objects": {        "text": {          "type": "text",          "text": "Hello World!",          "textAlign": "center",          "left": 50,          "top": 120,          "width": 120,          "height": 80        },        "image": {          "type": "picture",          "pictureFormat": "scaled",          "picture": "/RESOURCES/Images/HW.png",          "alignment": "center",          "left": 70,          "top": 20,          "width": 75,          "height": 75        },        "button": {          "type": "button",          "text": "OK",          "action": "Cancel",          "left": 60,          "top": 160,          "width": 100,          "height": 20        }      }    }  ] }
```

Das Formular lässt sich mit diesem 4D Code in ein Dialogfenster laden:

```
Open form window("/RESOURCES/HW.json")
DIALOG("/RESOURCES/HW.json")
```

Das Formular sieht folgendermaßen aus:



JSON Schema

Beim Erstellen von Formularen aus Dateien gilt folgendes:

- Sie müssen mit dem JSON Schema konform sein. Das 4D JSON Schema für Formulare finden Sie im 4D Programm Ordner unter */Resources/formsSchema.json*. Sie müssen selbst darauf achten, dass das Formular mit dem .json Format konform ist. Weitere Informationen dazu finden Sie unter dem Befehl **JSON Validate**.
- Sie müssen in einer Datei mit der Endung ".json" sein.
- Sie müssen innerhalb des Package der Anwendung gespeichert werden, zu der die Strukturdatei gehört.

Hinweise:

- Kommentare in .json Dateien müssen in der Eigenschaft "comments" enthalten sein. Standardmäßige Zeichen zum Kennzeichnen von Kommentaren, wie `"/"/`, `"/**"`, `"/**"`, etc., machen die Datei ungültig.
- Arrays in JSON arbeiten wie Collections in 4D

Dateipfade für Formulare

Alle Dateipfade können relativ oder absolut sein. Als Pfadtrenner dient das Zeichen '/', das wie folgt aufgelöst wird:

- Ein relativer Pfad darf nicht mit '/' beginnen. Er wird relativ zum JSON Dokument aufgelöst, wo der Pfadstring gefunden wird,
- Ein absoluter Pfad startet mit '/'. Aus Sicherheitsgründen wird nur `"/RESOURCES"` als absoluter Pfad akzeptiert und dieser bezeichnet den Ordner *Ressources* der aktuellen Anwendung. Beispiel: `"/RESOURCES/templates/myfile.json"`

zeigt auf die Datei "myfile.json" im Ordner *Ressources* der aktuellen Anwendung.

Weitere Informationen zum Umwandeln von Systempfaden in POSIX Syntax finden Sie unter dem Befehl **Convert path system to POSIX**.

Hinweise:

- Beim Auflösen des Namens wird die Groß- und Kleinschreibung berücksichtigt.
- 4D löst keinen Pfad zu einer .json Datei auf, die über das Netzwerk verfügbar ist (startet mit "http/https").
- Ist ein Pfad inkorrekt oder die Datei nicht konform zum JSON Schema, wird ein Fehler erzeugt.

JSON Zeiger

Dynamische Formulare können JSON Zeiger enthalten. Sie werden automatisch aufgelöst, wenn Sie einen der 4D Befehle aufrufen, die im Parameter ein dynamisches Formular erlauben (.json Dateipfad oder 4D Objekt):

- **DIALOG**
- **Open form window**
- **FORM SET INPUT**
- **FORM SET OUTPUT**
- **OBJECT GET SUBFORM**
- **OBJECT SET SUBFORM**
- **FORM LOAD**
- **Print form**

Weitere Informationen dazu finden Sie im Abschnitt **JSON Zeiger definieren**.

Seitenverzeichnis

- **Definition**
 - **Formulareigenschaften**
 - **Marker**
 - **Objekte**
 - **Unterformular**
- **Aktion**
 - **Aktion**
 - **Ereignisse**
- **Geometrie**
 - **Koordinaten & Größe**
 - **Formulargröße**
 - **Vergrößerungsoptionen**
- **Wert**
 - **Datenquelle**
 - **Anzeige**
 - **Eingabe**
 - **Wertebereich**
- **Darstellung**
 - **Anzeige**
 - **Hintergrund und Rahmen**
 - **Drucken**
 - **Text und Bild**
- **objektspezifische Eigenschaften**
 - **Animation - Bildschaltfläche**
 - **Schaltfläche, Kontrollkästchen, Optionsfeld**
 - **Skala für Ablaufanzeige, Halbkreis und Lineal**
 - **Gruppe**
 - **Hilfe**
 - **Listbox - allgemein**
 - **Listbox - Rasterlinien**
 - **Listbox - Kopf- und Fußteile**
 - **Bild**
 - **Web Area**

Definition

Die hier gelisteten Eigenschaften dienen als Basis und Struktur eines Formulars.

Formulareigenschaften

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
inheritedForm	String	Bezeichnet das Formular zum Vererben	Name (String) der Tabelle oder des Projektformulars ODER POSIX Pfad (String) zu einer .json Datei bzw. einem Objekt, welche das Formular beschreiben	Formular
inheritedFormTable	Tabelle	Bezeichnet die Tabelle, die ein vererbtes Formular verwendet.	Tabellenname oder -nummer	Formular
memorizeGeometry	Boolean	Sichert die Parameter des Formulars beim Schließen des Formularfensters, siehe Geometrie des Fensters beibehalten	TRUE / FALSE	Formular
windowTitle	String	Verwendet die Funktion Open form window für den Fenstertitel.	variabel	Formular

Marker

Marker geben genaue Stellen auf dem vertikalen Lineal des Formulars an und werden hauptsächlich in Ausgabeformularen verwendet. Sie steuern die gelistete Information und setzen Kopfteil, Umbruch, Hauptteil und Fußteil eines Formulars.

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
markerBody	Ganzzahl	Bereich zwischen den Steuerlinien Kopfteil 1 und Umbruch des Formulars	Minimum: 0	Formular
markerBreak	Ganzzahl/Array Ganzzahl	Bezeichnet Bereiche zum Anzeigen von Informationen, die nicht zu den Datensätzen gehören, oder Linien bzw. grafische Elemente, die die Anzeige auf dem Bildschirm abschließen.	Minimum: 0	Formular
markerFooter	Ganzzahl	Bereich zwischen den Steuerlinien Umbruch 1 und Fußteil des Formulars	Minimum: 0	Formular
markerHeader	Ganzzahl/Array Ganzzahl	Bereich über der Steuerlinie Kopfteil 1 des Formulars.	Ganzzahl Minimum: 0; Array Ganzzahl Minimum: 0	Formular

Objekte

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
memorizeValue	boolean	Sichert den Wert des aktuellen Objekts, wenn <i>memorizeGeometry</i> für das aktuelle Formular aktiviert ist.	TRUE / FALSE	Eingabe, Registerkarte, Kontrollkästchen, Optionsfeld, PopUp Menü, Splitter, Listbox
name	String	Name des Formularobjekts (Optional für das Formular)	Jeder Name, der nicht zu einem bereits vorhandenen Objekt gehört	Formular, Spalte der Listbox, Kopfteil der Listbox, Fußteil der Listbox
pluginAreaKind	String	Beschreibt Typ des PlugIn	variabel	PlugIn
text	String	Titel des Formularobjekts	variabel	Text, Gruppenrahmen, Schaltfläche, Kontrollkästchen, Optionsfeld, Kopfteil Listbox
type	String	Zwingend. Gibt den Typ des Formularobjekts an.	"text", "rectangle", "groupBox", "tab", "line", "button", "checkbox", "radio", "dropdown", "combo", "webArea", "write", "subform", "plugin", "splitter", "buttonGrid", "progress", "ruler", "spinner", "stepper", "list", "pictureButton", "picturePopup", "listbox", "input"	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
variableCalculation	String	Lässt mathematische Berechnungen zu	"none", "minimum", "maximum", "sum", "count", "average", "standardDeviation", "variance", "sumSquare"	Fußteil der Listbox

Unterformular

Ein Unterformular liegt innerhalb eines anderen Formulars.

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
deletableInList	Boolean	Gibt an, ob der Benutzer Unterdatensätze in einem Unterformular des Listenformulars löschen kann.	TRUE / FALSE	Unterformular
detailForm	String	Name eines vorhandenen Detailformulars	Name (String) der Tabelle oder des Projektformulars ODER POSIX Pfad (String) zu einer .json Datei ODER einem Objekt, die das Formular beschreiben	Unterformular, Listbox
doubleClickInEmptyAreaAction	String	Auszuführende Aktion bei Doppelklick auf eine leere Zeile in einem Unterformular	"addSubrecord"	Unterformular
doubleClickInRowAction	String	Auszuführende Aktion bei Doppelklick auf einen Datensatz im Unterformular	"editSubrecord", "displaySubrecord"	Unterformular, Listbox
enterableInList	Boolean	Gibt an, ob der Benutzer Datensätze direkt in der Liste ändern kann, ohne das dazugehörige Detailformular zu verwenden	TRUE / FALSE	Unterformular
listForm	String	Ein vorhandenes Unterformular im Listenformular, über das Sie Daten in anderen Tabellen eingeben, ansehen und ändern können.	Name (String) der Tabelle oder des Projektformulars ODER POSIX Pfad (String) zu einer .json Datei ODER einem Objekt, die das Formular beschreiben	Unterformular
selectionMode	string	Bezeichnet für Benutzer zugelassene Optionen zum Auswählen von Datensätzen.	"multiple", "single", "none"	Unterformular, Listbox

Aktion

Diese Eigenschaften geben Anweisungen für spezifische Aktionen.

Ereignisse

Um ein Ereignis aufzurufen, geben Sie den Namen oder Wert des Ereignisses in die Eigenschaft "events" ein: in camelCase, in eckigen Klammern, zwischen Anführungszeichen und mit dem ersten Buchstaben kleingeschrieben (*Beispiel*, "events": ["onLoad"] oder [1]).

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
events	Array vom Typ String oder Zahl	Die möglichen Ereignisse, um eine Aktion auszulösen.	onActivate, onAfterEdit, onAfterKeystroke, onAfterSort, onAlternateClick, onBeforeDataEntry, onBeforeKeystroke, onBeginDragOver, onBeginURLLoading, onBoundVariableChange, onClick, onCloseBox, onCloseDetail, onCollapse, onColumnMove, onColumnResize, onDataChange, onDeactivate, onDeleteAction, onDisplayDetail, onDoubleClick, onDragOver, onDrop, onEndURLLoading, onExpand, onFooterClick, onGettingFocus, onHeader, onHeaderClick, onLoad, onLoadRecord, onLongClick, onLosingFocus, onMenuSelect, onMouseEnter, onMouseLeave, onMouseMove, onMouseUp, onOpenDetail, onOpenExternalLink, onOutsideCall, onPagechange, onPluginArea, onPrintingBreak, onPrintingDetail, onPrintingFooter, onResize, onRowMove, onScroll, onSelectionChange, onTimer, onUnload, onURLFiltering, onURLLoadingError, onURLResourceLoading, onValidate, onVPRReady, onWindowOpeningDenied	Tabulator, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plugin, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Listboxspalte

Aktion

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
action	String	Typische Aktivitäten	Name einer gültigen Standardaktion. Siehe Standardaktionen .	Schaltfläche, Kontrollkästchen, Optionsfeld, Bildschaltfläche, DropDown Menü, Bildmenü, Schaltflächengitter, Registerkarte, Listbox
continuousExecution	Boolean	Definiert, ob die Methode eines Objekts laufen soll, während der Benutzer das Objekt bedient	TRUE / FALSE	Ablaufanzeige, Lineal, Stepper (numerischer Schritt)
dragging	String	Aktiviert Drag-Funktionen	"none", "custom", "automatic" (außer Liste, Listbox)	Plugin, Eingabe, Liste, Write, Listbox
dropping	String	Aktiviert Drop-Funktionen	"none", "custom", "automatic" (außer Liste, Listbox)	Plugin, Eingabe, Liste, Write, Listbox
method	String	Name einer Projektmethode	Name einer vorhandenen Projektmethode	Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Formular, Listbox, Ansicht, Spalte der Listbox
movableRows	Boolean	Erlaubt, Zeilen während der Ausführung zu bewegen	TRUE / FALSE	Listbox
sortable	Boolean	Erlaubt, Daten in Spalten durch Klick in den Kopfteil zu sortieren	TRUE / FALSE	Listbox

Geometrie

Koordinaten & Größe

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
borderRadius	Ganzzahl	Eckradius für runde Vierecke	minimum:0	Viereck Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
bottom	Ganzzahl	Setzt ein Objekt nach unten (zentriert)	minimum: 0	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
height	Ganzzahl	Gibt die Höhe eines Objekts an	minimum: 0	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
left	Ganzzahl	Setzt ein Objekt nach links	minimum: 0	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
maxWidth	Ganzzahl	Gibt die maximale Breite für Spalten der Listbox an	minimum: 0	Spalten der Listbox
minWidth	Ganzzahl	Gibt die minimale Breite für Spalten der Listbox an	minimum: 0	Spalten der Listbox
right	Ganzzahl	Setzt ein Objekt nach rechts	minimum: 0	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
rowHeight	String	Setzt die Zeilenhöhe in der Listbox	css Wert der Einheit "em" oder "px" (Standard)	Listbox
rowHeightAuto	Boolean	Aktiviert oder deaktiviert die automatische Anpassung der Zeilenhöhe in der Listbox	TRUE / FALSE	Listbox
rowHeightAutoMax	String	Gibt die zulässige Maximalhöhe für Zeilen der Listbox an	css Wert der Einheit "em" oder "px" (Standard). minimum: 0	Listbox
rowHeightAutoMin	String	Gibt die zulässige Mindesthöhe für Zeilen der Listbox an	css value Einheit "em" oder "px" (Standard). minimum: 0	Listbox
rowHeightSource	variabel	Array mit verschiedenen Zeilenhöhen in einer Listbox	Name einer 4D Array Variable	Listbox
startPoint	String	Gibt an, wo eine Linie beginnt	"topLeft", "bottomLeft"	Linie

top	Ganzzahl	Setzt ein Objekt nach oben (zentriert).	minimum: 0	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
width	Ganzzahl	Gibt die Breite des Objekts an	minimum: 0	Formular, Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe

Formulargröße

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
bottomMargin	Ganzzahl	Wert für vertikalen Rand (in Pixel).	minimum: 0	Formular
formSizeAnchor	String	Name des Objekts, dessen Position die Größe des Formulars bestimmt (Mindestbreite: 1)	Name eines 4D Objekts	Formular
rightMargin	Ganzzahl	Wert für horizontalen Rand (in Pixel).	minimum: 0	Formular
windowMaxHeight	Ganzzahl	Gibt die größtmögliche Höhe an	minimum: 0	Formular
windowMaxWidth	Ganzzahl	Gibt die größtmögliche Breite an	minimum: 0	Formular
windowMinHeight	Ganzzahl	Gibt die kleinstmögliche Höhe an	minimum: 0	Formular
windowMinWidth	Ganzzahl	Gibt die kleinstmögliche Breite an	minimum: 0	Formular
windowSizingX	String	Gibt an, ob der Benutzer die Formularhöhe verändern kann (vertikal)	"fixed", "variable"	Formular
windowSizingY	String	Gibt an, ob der Benutzer die Formularbreite verändern kann (horizontal)	"fixed", "variable"	Formular

Vergößerungsoptionen

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
resizable	Boolean	Gibt an, ob der Benutzer die Größe eines Objekts ändern kann	TRUE / FALSE	Spalten der Listbox Formular, Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
sizingX	String	Gibt an, ob das Objekt horizontal bewegt oder angepasst wird, wenn der Benutzer die Formulargröße verändert	"move", "grow", "fixed"	Formular, Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
sizingY	String	Gibt an, ob das Objekt vertikal bewegt oder angepasst wird, wenn der Benutzer die Formulargröße verändert	"move", "grow", "fixed"	Formular, Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe
splitterMode	String	Bei einem Splitterobjekt mit Splittermodus werden andere Objekte rechts davon (vertikaler Splitter) oder darunter (horizontaler Splitter) gleichzeitig mit dem Splitter ohne Stop bewegt	"resize", "move"	Splitter

Wert

Datenquelle

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
automaticInsertion	Boolean	Fügt einen Wert automatisch in einer abgespeicherten Liste hinzu, wenn der Benutzer einen Wert eingibt, der in der dem Objekt zugeordneten Auswahlliste nicht vorhanden ist	TRUE / FALSE	Combo Box, Spalte der Listbox
choiceList	Liste	Weist einer Spalte in der Listbox eine Auswahlliste zu	variabel	Eingabe, Dropdown Menü, Combo Box, Spalte der Listbox
dataSource	String oder String Array für Spalte in hierarchischer Listbox	Eine 4D Variable, Feldname oder beliebiger komplexer Programmierspracheausdruck	variabel	Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Schreiben, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Spalte, Kopfteil, Fußteil der Listbox, Ansicht
dataSourceTypeHint	String	Hinweis auf Typ einer automatisch zugewiesenen Variable, wenn <i>Datenquelle</i> nicht angegeben ist oder auf Auswahl der entsprechenden Eigenschaft des Anzeigeformats (textFormat, numberFormat, timeFormat, dateFormat, pictureFormat, booleanFormat).	"integer", "number", "boolean", "picture", "text", "date", "time", "arrayText", "object", "undefined"	Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Schreiben, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Listbox, Spalte der Listbox, Fußteil der Listbox
labels	list	Weist einer Spalte der Listbox Standardwerte oder Auswahlliste zu	variabel	Registerkarte
list	list	Liste, die einem Formularobjekt hierarchische Liste zugeordnet ist	variabel	Liste
listboxType	String	Definiert das Verhalten der Listbox (Typ)	"array", "currentSelection", "namedSelection", "collection"(*)	Listbox
selectionName	String	Inhalt der Listbox basiert auf der angegebenen Auswahl	Name der Auswahl	Listbox
saveAs	String	Gibt bei Auswählen des Eintrags in einer Liste an, ob Wert oder Referenz des Eintrags wiedergefunden werden soll	"value"; "reference"	Dropdown Menü, Spalte der Listbox
table	String/Ganzzahl	Gibt die Datenquelle für ein Unterformular oder eine Listbox an	Vorhandene Tabellennamen (ohne "[" "]" oder Tabellen ID	Unterformular, Listbox

(Ganzzahl
minimum: 0)

(*) "collection" bezeichnet eine Listbox, die auf einer Collection **oder einer Entity-Selection** basiert.

Anzeige

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
booleanFormat	String	Gibt nur zwei mögliche Werte an	"TRUE"; "FALSE"	Eingabe, Spalte der Listbox
controlType	String	Gibt an, wie der Wert in der Zelle einer Listbox gerendert werden soll	"input", "checkbox" (für boolean / numerische Spalten), "automatic", "popup" (nur für boolean Spalten)	Spalte der Listbox
dateFormat	string	Steuert, wie Datum beim Anzeigen oder Drucken erscheint. Auswählbar sind nur die in 4D integrierten Formate	"systemShort", "systemMedium", "systemLong", "iso8601", "rfc822", "short", "shortCentury", "abbreviated", "long", "blankIfNull" (ist mit anderen möglichen Werten kombinierbar)	Eingabe, Spalte der Listbox, Dropdown Menü, Combo Box, Fußteil der Listbox
numberFormat	String	Steuert, wie Zahlen beim Anzeigen oder Drucken erscheinen	Zahlen (mit Dezimaltrenner oder bei Bedarf mit Minuszeichen)	Eingabe, Dropdown Menü, Combo Box, Ablaufanzeige, Lineal, Spalte der Listbox, Fußteil der Listbox
pictureFormat	String	Steuert, wie Bilder beim Anzeigen oder Drucken erscheinen	"truncatedTopLeft", "scaled", "truncatedCenter", "tiled" (nur Bilder), "proportionalTopLeft" (außer Bilder), "proportionalCenter"(außer Bilder)	Bild, Eingabe, Listbox, Fußteil der Listbox
textFormat	String	Steuert, wie alphanumerische Werte und Variablen beim Anzeigen oder Drucken erscheinen.	"### ####", "(###) ### ####", "### ### ####", "#####", "00000", custom formats	Eingabe, Dropdown Menü, Combo Box, Spalte der Listbox, Fußteil der Listbox
timeFormat	string	Steuert, wie Zeitangaben beim Anzeigen oder Drucken erscheinen. Auswählbar sind nur die in 4D integrierten Formate	"systemShort", "systemMedium", "systemLong", "iso8601", "hh_mm_ss", "hh_mm", "hh_mm_am", "mm_ss", "HH_MM_SS", "HH_MM", "MM_SS", "blankIfNull" (ist mit anderen möglichen Werten kombinierbar)	Eingabe, Dropdown Menü, Spalte der Listbox, Fußteil der Listbox
truncateMode	String	Steuert die Anzeige der Werte, wenn die Spalten der Listbox zum Anzeigen des kompletten Inhalts zu schmal sind	"withEllipsis"; "none"	Spalte der Listbox, Fußteil der Listbox
visibility	String	Erlaubt, das Objekt in der Anwendungsumgebung auszublenden	"visible", "hidden", "selectedRows", "unselectedRows"	allgemein, Spalte der Listbox

Eingabe

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
contextMenu	String	Gibt dem Benutzer Zugriff auf ein Standard Kontextmenü im ausgewählten Bereich	"automatic", "none"	Eingabe, Web Area, Write, Spalte der Listbox
enterable	Boolean	Erlaubt die Eingabe	TRUE / FALSE	Eingabe, Spalte der Listbox, Liste, Stepper (numerischer Schritt), Ablaufanzeige, Lineal, Write
entryFilter	String	Weist dem Objekt oder Zellen einer Listbox einen Eingabefilter zu. Diese Eigenschaft ist nur verfügbar, wenn die Eigenschaft "Enterable" aktiviert ist.	variabel	Eingabe, Combo Box, Liste, Spalte der Listbox
focusable	Boolean	Erlaubt, dass das Objekt den Fokus hat (und über die Tastatur aktivierbar ist).	TRUE / FALSE	Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Write, Unterformular, Plug-In Bereich, Lineal, Liste, Listbox
keyboardDialect	String	Weist einem Feld oder eingebbaren Objekt über Sprach-Identifizier RFC 3066 Bis ein spezifisches Tastatur-Layout zu.	variabel	Eingabe, Write
multiline	String	Definiert, ob Text, der für die Anzeige zu lang ist, abgeschnitten wird (mit oder ohne Zeilenumbrüche)	"automatic", "yes", "no"	Eingabe
placeholder	String	Anzuzeigender Text (in Grauschrift), wenn der Wert für dataSource leer ist	variabel	Eingabe, Combo Box
shortcutAccel	Boolean	Definiert das Betriebssystem, Windows oder Mac	TRUE / FALSE	Kontrollkästchen, Optionsfeld, Bildschaltfläche
shortcutAlt	Boolean	Definiert die Alt-Taste	TRUE / FALSE	Kontrollkästchen, Optionsfeld, Bildschaltfläche
shortcutCommand	Boolean	Definiert die Befehlstaste (Mac)	TRUE / FALSE	Kontrollkästchen, Optionsfeld, Bildschaltfläche
shortcutControl	Boolean	Definiert die Strg-Taste (Windows)	TRUE / FALSE	Kontrollkästchen, Optionsfeld, Bildschaltfläche
shortcutKey	String	Buchstabe oder Name für spezifisches Tastenkürzel	"[F1]" -> "[F15]", "[Return]", "[Enter]", "[Backspace]", "[Tab]", "[Esc]", "[Del]", "[Home]", "[End]", "[Help]", "[Page up]", "[Page down]", "[left arrow]", "[right arrow]", "[up arrow]", "[down arrow]"	Kontrollkästchen, Optionsfeld, Bildschaltfläche
shortcutShift	Boolean	Definiert die Shift-Taste	TRUE / FALSE	Kontrollkästchen, Optionsfeld, Bildschaltfläche
showSelection	Boolean	Hält die Auswahl im Objekt sichtbar, wenn es keinen Fokus mehr hat	TRUE / FALSE	Eingabe, Write
singleClickEdit	Boolean	Aktiviert direkten Wechsel in Eingabemodus	TRUE / FALSE	Listbox
spellcheck	Boolean	Definiert, ob	TRUE / FALSE	Eingabe, Write

Rechtschreibung
automatisch überprüft
wird

Wertebereich

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
excludedList	Liste	Erlaubt eine Liste mit ausgenommenen Werten in der Spalte. Die Eingabe eines ausgenommenen Werts wird nicht akzeptiert und es erscheint eine Fehlermeldung	variabel	Eingabe, Combo Box, Spalte der Listbox
max	String / Zahl	Erlaubter Maximumwert. Das sind bei numerischen Steppern Sekunden, wenn dem Objekt ein Wert vom Typ Zeit zugewiesen ist, und wird ignoriert, wenn dem Objekt ein Wert vom Typ Datum zugewiesen ist.	variabel	Eingabe, Ablaufanzeige, Lineal, Stepper (numerischer Schritt)
min	String / Zahl	Erlaubter Minimumwert. Das sind bei numerischen Steppern Sekunden, wenn dem Objekt ein Wert vom Typ Zeit zugewiesen ist, und wird ignoriert, wenn dem Objekt ein Wert vom Typ Datum zugewiesen ist.	variabel	Eingabe, Ablaufanzeige, Lineal, Stepper (numerischer Schritt)
requiredList	Liste	Erlaubt eine Liste mit erforderlichen Werten. Dann lassen sich nur diese Werte auswählen, die Eingabe über Tastatur ist nicht mehr möglich.	variabel	Eingabe, Spalte der Listbox

Darstellung

Anzeige

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
defaultButton	Boolean	Similar to a standard button except that it has a modified appearance, intended to indicate the recommended choice to the user.	TRUE / FALSE	Schaltfläche
dpi	String	Setzt die Bildschirmauflösung für den Inhalt im 4D Write Pro Bereich	0, 72, 96	Write
hideFocusRing	Boolean	Während der Ausführung wird ein Feld bzw. jedes eingebare Objekt mit Fokus durch ein Auswahlrechteck hervorgehoben. Mit dieser Eigenschaft können Sie dieses Rechteck ausblenden.	TRUE / FALSE	Eingabe, Write, Unterformular, Liste, Listbox
hideSystemHighlight	Boolean	Gilt nur für Listboxen vom Typ Auswahl und dient dazu, hervorgehobene Datensätze in der Listbox auszublenden.	TRUE / FALSE	Listbox
labelsPlacement	String	Gibt die Position der Beschriftung für ein Objekt an	"none", "top", "bottom", "left", "right"	Registerkarte, Ablaufanzeige, Lineal
layoutMode	String	Setzt die Art der Anzeige für das 4D Write Pro Dokument im Formularbereich.	"embedded", "draft", "page"	Write
scrollbarHorizontal	String	Erlaubt dem Benutzer, den Ansichtsbereich nach links oder rechts zu ziehen.	"visible", "hidden", "automatic"	Eingabe, Write, Unterformular, Liste, Listbox
scrollbarVertical	String	Erlaubt dem Benutzer, den Ansichtsbereich nach oben oder unten zu ziehen.	"visible", "hidden", "automatic"	Eingabe, Write, Unterformular, Liste, Listbox
showBackground	Boolean	Blendet Hintergrundbilder und Hintergrundfarbe ein/aus (standardmäßig angezeigt).	TRUE / FALSE	Write
showHeaders	Boolean	Aktiviert/deaktiviert die Sichtbarkeit von Kopfteilen in Dokument oder Listbox	TRUE / FALSE	Write, Listbox
showHiddenChars	Boolean	Blendet unsichtbare Zeichen ein/aus (standardmäßig ausgeblendet).	TRUE / FALSE	Write
showHorizontalRuler	Boolean	Blendet das Lineal ein/aus (standardmäßig angezeigt).	TRUE / FALSE	Write
showHTMLWysiwyg	Boolean	Aktiviert/deaktiviert die HTML WYSIWYG Anzeige, in der alle erweiterten 4D Write Pro Attribute entfernt werden, die nicht konform mit allen Browsern sind (standardmäßig deaktiviert).	TRUE / FALSE	Write
showFooters	Boolean	Aktiviert/deaktiviert die Sichtbarkeit von Fußteilen in Dokument oder Listbox	TRUE / FALSE	Write, Listbox
showPageFrames	Boolean	Blendet den Seitenrahmen ein/aus, wenn der Ansichtsmodus auf "Page" gesetzt ist (standardmäßig ausgeblendet)	TRUE / FALSE	Write
showReferences	Boolean	Zeigt alle eingefügten 4D Ausdrücke im Dokument als Referenzen.	TRUE / FALSE	Write
zoom	String	Setzt Zoom in Prozent zum Anzeigen von Inhalt im 4D Write Pro Bereich. Standard ist 100%.	25, 50, 75, 100, 125, 150, 175, 200, 300, 400	Write

Hintergrund und Rahmen

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
alternateFill	Farbe	Erlaubt, für Zeilen/Spalten in der Listbox mit ungerader Nummer eine andere Hintergrundfarbe zu setzen	jeder css Wert; "transparent"; "automatic"	Listbox, Spalte der Listbox
borderStyle	String	Definiert einen Standardstil für Rahmen des Listbox Objekts	"system", "none", "solid", "dotted", "raised", "sunken", "double"	Text, Eingabe, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox
fill	Farbe	Definiert die Hintergrundfarbe eines Objekts	jeder css Wert; "transparent"; "automatic"	Text, Rechteck, Oval, Eingabe, Liste, Listbox
hideExtraBlankRows	Boolean	Steuert die Sichtbarkeit zusätzlicher Leerzeilen	TRUE / FALSE	Listbox
rowFillSource	Variable	Name eines Array oder Ausdrucks, um eine eigene Hintergrundfarbe auf jede Zeile der Listbox zu setzen	RGB Farbwerte	Listbox, Spalte der Listbox
strokeDashArray	Array Zahl oder String	Definiert Werte für jede Zeile eines Objekts. Beispiel: "6 1" oder [6,1]	variabel	Linie, Rechteck, Oval
strokeWidth	Ganzzahl	Bestimmt die Dicke einer Linie	Ganzzahl oder 0 für kleinste Breite in einem gedruckten Formular	Linie, Rechteck, Oval

Drucken

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
printFrame	String	Verwalten den Druckmodus für Objekte, deren Größe von einem Datensatz zum anderen je nach Inhalt variieren kann	"fixed", "fixedMultiple", "variable"	Unterformular, Eingabe, Write

Text und Bild

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
allowFontColorPicker	Boolean	Erlaubt Änderung durch Benutzer von Schrift oder Farbe eines Objekts für die aktuelle Sitzung	TRUE / FALSE	Eingabe, Listbox
customBackgroundPicture	Bild	Setzt Hintergrundbild für die Schaltfläche	absoluter oder relativer Pfad über Sandbox-Prozess. Muss zusammen mit der Eigenschaft <i>style</i> mit der Option "custom" verwendet werden.	Schaltfläche, Kontrollkästchen, Optionsfeld
customBorderX	Ganzzahl	Setzt die Größe (in Pixel) der inneren horizontalen Ränder eines Objekts. Muss zusammen mit der Eigenschaft <i>style</i> mit der Option "custom" verwendet werden.	variabel	Schaltfläche, Kontrollkästchen, Optionsfeld
customBorderY	Ganzzahl	Setzt die Größe (in Pixel) der inneren vertikalen Ränder eines Objekts. Muss zusammen mit der Eigenschaft <i>style</i> mit der Option "custom" verwendet werden.	variabel	Schaltfläche, Kontrollkästchen, Optionsfeld
customOffset	Ganzzahl	Setzt einen eigenen Wert für Versatz in Pixel. Muss zusammen mit der Eigenschaft <i>style</i> mit der Option "custom" verwendet werden.	variabel	Schaltfläche, Kontrollkästchen, Optionsfeld
fontFamily	String	Name der verwendeten Schrift im Objekt.	variabel	Text, Gruppenrahmen, Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Ablaufanzeige, Lineal, Liste, Listbox
fontSize	Ganzzahl	Schriftgröße in Punkt.	minimum: 0	Text, Gruppenrahmen, Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Ablaufanzeige, Lineal, Liste, Listbox
fontStyle	String	Setzt den ausgewählten Text leicht nach rechts geneigt	"normal", "italic"	Text, Gruppenrahmen, Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Ablaufanzeige, Lineal, Liste, Listbox
fontTheme	String	Definiert einen Stil für Text des Objekts.	"normal", "main", "additional"	Text, Eingabe
fontWeight	String	Setzt den ausgewählten Text auf dunklere und stärkere Anzeige	"normal", "bold"	Text, Gruppenrahmen, Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box,

				Ablaufanzeige, Lineal, Liste, Listbox
rowStrokeSource	String	Name des Array oder Ausdrucks zur Verwaltung von Zeilenfarben	variabel	Listbox, Spalte der Listbox
rowStyleSource	String	Name des Array oder Ausdrucks zur Verwaltung von Stillarten	varies	Listbox, Spalte der Listbox
storeDefaultStyle	Boolean	Speichert die Stil-Tags mit dem Text. Muss zusammen mit der Eigenschaft <i>multistyle</i> verwendet werden.	Text, Viereck, Oval, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Liste, Listbox, Spalte/Fußteil/Kopfteil der Listbox	
styledText	Boolean	Wendet eigene Stilarten auf dynamische Textbereiche (Felder oder Variablen) in der Anwendungsumgebung an	TRUE / FALSE	Eingabe
textAlign	String	Richtet im Bereich enthaltenen Text horizontal aus	"automatic", "right", "center", "justify", "left"	Text, Gruppenrahmen, Eingabe, Listbox
textAngle	String	Ändert die Richtung (Rotation) des Textbereichs.	0, 90, 180, 270	Text, Eingabe
textDecoration	String	Setzt im ausgewählten Text eine Linie unter den Text	"normal", "underline"	Text, Gruppenrahmen, Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Ablaufanzeige, Lineal, Liste, Listbox
verticalAlign	String	Richtet im Bereich enthaltenen Text vertikal aus	"automatic", "top", "middle", "bottom"	Listbox, Spalte/Fußteil/Kopfteil der Listbox
wordwrap	String	Verwaltet die Anzeige von Inhalt, wenn er die Breite des Objekts übersteigt	"automatic" (außer Listbox), "normal", "none"	Eingabe, Fußteil/Kopfteil der Listbox

objektspezifische Eigenschaften

Nachfolgende Eigenschaften gelten jeweils für die angegebenen Objekte.

Animation - Bildschaltfläche

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
frameDelay	Ganzzahl	Durchläuft Inhalt der Bildschaltfläche mit der angegebenen Geschwindigkeit (in Ticks)	minimum: 0	Bildschaltfläche
loopBackToFirstFrame	Boolean	Bilder werden in kontinuierlicher Schleife angezeigt	TRUE / FALSE	Bildschaltfläche
switchBackWhenReleased	Boolean	Zeigt immer das erste Bild, wenn die Schaltfläche nicht angeklickt ist. Zeigt das zweite Bild bis zum Loslassen der Maustaste	TRUE / FALSE	Bildschaltfläche
switchContinuously	Boolean	Erlaubt dem Benutzer, die Bilder mit gedrückter Maustaste fortlaufend anzuzeigen (z.B. als Animation).	TRUE / FALSE	Bildschaltfläche
switchWhenRollover	Boolean	Ändert den Inhalt der Bildschaltfläche, wenn der Mauszeiger darüberzieht. Das Startbild erscheint, wenn der Cursor den Bereich der Schaltfläche verlässt	TRUE / FALSE	Bildschaltfläche
useLastFrameAsDisabled	Boolean	Legt bei Deaktivieren der Schaltfläche das letzte Thumbnail als Anzeige fest	TRUE / FALSE	Bildschaltfläche

Schaltfläche, Kontrollkästchen, Optionsfeld

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
columnCount	Ganzzahl	Setzt die Anzahl Spalten in einer Tabelle mit Thumbnails	minimum: 1	Gruppenrahmen, Bildschaltfläche, Bildmenü
popupPlacement	String	Definiert Anzeige eines Dreiecksymbols in der Schaltfläche, um anzuzeigen, dass ein PopUp-Menü zugewiesen ist.	"none", "linked", "separated"	Schaltfläche
radioGroup	String	Ermöglicht Optionsfelder als koordinierte Sets: im Set lässt sich immer nur eine Schaltfläche auswählen	Optionsfeld Gruppenname	Kontrollkästchen
rowCount	Ganzzahl	Setzt die Anzahl Zeilen in einer Tabelle mit Thumbnails	minimum: 1	Schaltflächengitter, Bildschaltfläche, Bildmenü
style	String	Setzt die allgemeine Darstellung der Schaltfläche. Weitere Informationen unter Schaltflächenstil	"regular", "toolbar", "bevel", "roundedBevel", "gradientBevel", "texturedBevel", "office", "help", "circular", "disclosure", "roundedDisclosure", "custom"	Schaltfläche, Kontrollkästchen, Optionsfeld
textPlacement	String	Ändert die Position des Titels in Bezug zum zugewiesenen Objekt. Hat keine Auswirkung, wenn das Objekt nur einen Titel (kein zugewiesenes Bild) oder ein Bild (kein Titel) enthält.	"left", "right", "top", "bottom", "center"	Schaltfläche, Kontrollkästchen, Optionsfeld
threeState	Boolean	Ermöglicht für Objekt Optionsfeld einen dritten Status.	TRUE / FALSE	Optionsfeld

Skala für Ablaufanzeige, Halbkreis und Lineal

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
graduationStep	Ganzzahl	Maß für Anzeige der Skala	variabel	Ablaufanzeige, Lineal
step	Ganzzahl	Mindestintervall zwischen Werten in Benutzung. Bei numerischen Steppern stellt diese Eigenschaft Sekunden dar, wenn dem Objekt ein Wert vom Typ Zeit zugewiesen ist und Tage bei einem Wert vom Typ Datum	variabel	Ablaufanzeige, Lineal, Stepper (numerischer Schritt)
showGraduations	Boolean	Blendet die Skala zu den Bezeichnungen ein/aus	TRUE / FALSE	Ablaufanzeige, Lineal

Gruppe

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
group	string	Definiert Gruppenname für ein Objekt.	variabel	Text, Viereck, Gruppenrahmen, Registerkarte, Linie, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Web Area, Write, Unterformular, Plug-In Bereich, Splitter, Schaltflächengitter, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Listbox, Eingabe

Hilfe

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
tooltip	String	Zeigt Benutzern zusätzliche Angaben über ein Feld an	variabel	Registerkarte, Eingabe, Schaltfläche, Kontrollkästchen, Optionsfeld, Dropdown Menü, Combo Box, Splitter, Gruppenrahmen, Ablaufanzeige, Lineal, Spinner (drehendes Rad), Stepper (numerischer Schritt), Liste, Bildschaltfläche, Bildmenü, Kopfteil/Fußteil der Listbox

Listbox - allgemein

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
columns	Spalte Array	Enthält die Attribute für Spalten der Listbox	variabel	Listbox
highlightSet	String	Gibt das Set zum Verwalten markierter Datensätze in der Listbox an (Ist die Datenquelle Arrays ausgewählt, wird ein boolean Array mit dem gleichen Namen wie die Listbox verwendet)	variabel	Listbox
lockedColumnCount	Ganzzahl	Anzahl Spalten, die im linken Teil der Listbox erhalten bleiben muss - auch bei horizontalem Scrollen durch den Benutzer	minimum: 0	Listbox
staticColumnCount	Ganzzahl	Anzahl Spalten, die während der Ausführung nicht bewegt werden können	minimum: 0	Listbox

Listbox - Rasterlinien

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
horizontalLineStroke	Farbe	Definiert die Farbe der horizontalen Linien in der Listbox (standardmäßig grau)	jeder css Wert, "transparent", "automatic"	Listbox
verticalLineStroke	Farbe	Definiert die Farbe der vertikalen Linien in der Listbox (standardmäßig grau)	jeder css Wert, "transparent", "automatic"	Listbox

Listbox - Kopf- und Fußteile

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
footer	Objekt	Definiert ein Objekt am unteren Rand und getrennt vom Hauptteil der Listbox	variabel	Spalte der Listbox
footerHeight	String	Setzt die Höhe für Kopf- oder Fußteil der Listbox. Als Einheit kann Zeile oder Pixel dienen	Muster $^(\d+)(px em)?$$ (positive Dezimalzahl + px/em)	Listbox
header	Objekt	Definiert ein Objekt am oberen Rand und getrennt vom Hauptteil der Listbox	variabel	Spalte der Listbox
headerHeight	string	Setzt die Höhe für Kopf- oder Fußteil der Listbox. Als Einheit kann Zeile oder Pixel dienen	Muster $^(\d+)(px em)?$$ (positive Dezimalzahl + px/em)	Listbox
showFooters	Boolean / Objekt	Blendet Fußteile von Spalten ein/aus. Diese Eigenschaft ist boolean für Listboxen und ein Objekt für Spalten der Listbox	TRUE / FALSE	Write, Listbox
showHeaders	Boolean	Blendet Kopfteile von Spalten ein/aus. Diese Eigenschaft ist boolean für Listboxen und ein Objekt für Spalten der Listbox	TRUE / FALSE	Write, Listbox

Bild



















Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
icon	Bild	Name, Nummer oder Pfadname des Bildes	absoluter oder relativer Pfad über Sandbox-Prozess	Kopfteil der Listbox, Schaltfläche, Kontrollkästchen
iconFrames	Ganzzahl	Setzt die genaue Anzahl der im Bild vorhandenen Stadien	minimum: 1	Schaltfläche, Kontrollkästchen, Optionsfeld
iconPlacement	String	Definiert die Position des Icon in Bezug zum Formularobjekt	"none", "left", "right"	Kopfteil der Listbox

Web Area

Eigenschaft	Typ	Beschreibung	Mögliche Werte	Unterstützte Objekte
methodsAccessibility	String	Erstellt Instanz eines speziellen JavaScript Objekts (<code>\$4d</code>), um Aufrufe von 4D Projektmethoden zu verwalten. Muss zusammen mit der Eigenschaft <i>webEngine</i> verwendet werden	"all", "none"	Web Area
progressSource	String	Wert zwischen 0 und 100, der das Laden der Seite in den Web Bereich in Prozent anzeigt. Wird von 4D automatisch aktualisiert, lässt sich nicht manuell verändern	variabel	Web Area
urlSource	String	Definiert die geladene oder zu ladende URL durch den zugewiesenen Web Bereich	variabel	Web Area
webEngine	String	Zum Wählen zwischen zwei Rendering Engines für den Web Bereich, abhängig von den Spezifikationen der Applikation	"embedded", "system"	Web Area

Mit aktiven Objekten arbeiten

Was sind aktive Objekte

-  Objekte vom Typ Feld und Variable
-  Schaltflächen
-  3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder
-  Bildschaltflächen
-  Schaltflächengitter
-  Kontrollkästchen
-  Optionsfelder und Optionsbilder
-  PopUp-Menüs, Dropdown-Listen
-  Combo Boxen
-  Hierarchische PopUp-Menüs und hierarchische Listen
-  PopUp-Menü vom Typ Bild
-  Indikatoren
-  Registerkarten
-  Splitter
-  Web Areas
-  Plug-In Bereich
-  Listboxen
-  Unterformulare

🌿 Was sind aktive Objekte

Ein aktives Objekt ist ein Element in einem Formular, das eine Datenbankaufgabe oder eine Oberflächenfunktion erfüllt. Es gibt viele verschiedene Arten von aktiven Objekten. Datenfelder zählen zu den aktiven Objekten. Andere aktive Objekte - eingebare Objekte (Variablen), Combo Boxen, Dropdown-Listen, Bildschaltflächen etc. - speichern Daten vorübergehend oder führen eine Aktion aus: Sie öffnen z.B. ein Dialogfenster, drucken einen Bericht oder starten einen Hintergrundprozess.

In einigen Fällen können Sie die Aktion des aktiven Objekts über die Eigenschaftenliste festlegen. Sie können zum Beispiel einer Schaltfläche eine integrierte automatische Schaltflächenaktion zuordnen oder die Aktion eines Objekts durch Schreiben einer Methode festlegen, die fest mit dem Objekt verknüpft ist.

Sie können folgende aktiven Objekte in 4D verwenden:

- **Get list item font**
- **Schaltflächen**
- **3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder**
- **Bildschaltflächen**
- **Schaltflächengitter**
- **Kontrollkästchen**
- **Optionsfelder und Optionsbilder**
- **PopUp-Menüs, Dropdown-Listen**
- **Combo Boxen**
- **Hierarchische PopUp-Menüs und hierarchische Listen**
- **PopUp-Menü vom Typ Bild**
- **Indikatoren**
- **Registerkarten**
- **Splitter**
- **Web Areas**
- **Plug-In Bereich**
- **Listboxen** (Beschreibung in einem eigenen Kapitel)
- **Unterformulare und Widgets** (Beschreibung in einem eigenen Kapitel)

Eine Reihe von Eigenschaften verwalten die Darstellung und Funktionsweise aktiver Objekte. Es gibt spezifische Eigenschaften, die beim jeweiligen Objekt beschrieben werden, sowie allgemeine Grundeigenschaften, wie Dateneingabekontrolle oder Anzeigeformate. Weitere Informationen dazu finden Sie unter **Eigenschaften für aktive Objekte**.

Native Objekt Animationen (4D 64-bit Versionen von OS X)

64-bit 4D Anwendungen auf OS X nutzen native Animationen, um die Benutzerführung in 4D Formularen zu verbessern. Das gilt für folgende Elemente:


- Felder und eingebare Variablen, Optionsfelder und Kontrollkästchen werden animiert, wenn sie den Fokus erhalten
- Optionsfelder and Kontrollkästchen lösen bei Aktionen mit der Maus Animation aus
- Rollbalken haben eine spezifische Animation (nur in der Yosemite Version)
- Einsetzen des Tabulators in Listenformularen sorgt für angenehmes Gleiten zwischen den Spalten

Nachfolgende Sequenz zeigt die verschiedenen Animationen:

Hinweis: Die [Guidelines](#) von Apple bieten nützliche Tipps zum Einsetzen von Animationen in Ihren Anwendungen.

Objekte vom Typ Feld und Variable

In Formularen arbeiten Felder und Variablen in vielem auf dieselbe Art und Weise.

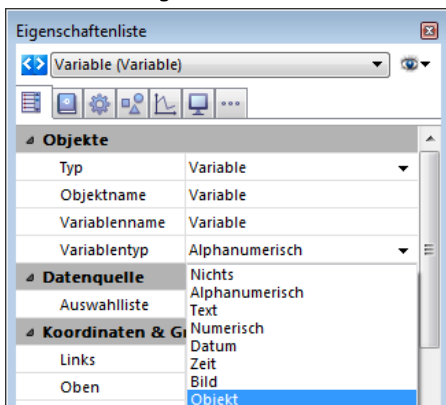
- Felder dienen zum Eingeben und Anzeigen der Daten eines Datensatzes. Wenn Sie ein neues Formular über den Formularassistenten erstellen, wählen Sie die Felder, die Sie als Standardobjekte in das Formular einfügen wollen. Ist das Formular angelegt, können Sie über den Formulareditor weitere Eigenschaften wie Anzeigeformat und Dateneingabekontrolle festlegen. Diese Eigenschaften gelten nur für das Formular, in dem sie definiert wurden. Für andere Formulare können Sie dieselben Eigenschaften verwenden oder neue festlegen. Sie können die Eigenschaften der Felder verändern oder jederzeit löschen bzw. neue vergeben.
- Variablen können eingebbar oder nicht-eingebbar sein und Daten vom Typ Alpha, Text, Zahl, Datum, Zeit und/oder Bild empfangen. Sie werden in Formularen über das Werkzeug Variable  der Objektleiste angelegt. Über die Funktion **Auf Matrix duplizieren** können Sie eine große Anzahl Variablen automatisch erstellen. Analog zu Feldern können Sie auch in Variablen Daten eingeben und anzeigen. Variablen werden zum temporären Speichern von Daten verwendet. Sie dienen häufig für Berechnungen, z.B.:
 $vTotal := Anzahl * Preis$
Sie erzeugen eine Variable, die das Ergebnis der Berechnung anzeigt, benennen die Variable vTotal und verwenden eine Methode zur Berechnung.
Sie verwenden Methoden zum Verwalten von eingebbaren und nicht eingebbaren Variablen. Eine eingebbare Variable übernimmt Daten. Sie können Eingabekontrollen für die Variable wie für ein Datenfeld festlegen. Die eingegebenen Daten werden mit dem Objektname verknüpft. Sie können die Daten mit Objekt- oder Formularmethoden verwalten, indem Sie den Objektname als Variable verwenden.

Sie können Felder und Variablen in Ihren Formularen auch in anderer Form darstellen:

- Sie können Daten aus Datenfeldern direkt in Listboxen in Spalten vom Typ Auswahl anzeigen und eingeben (siehe **Anzeige der Felder in Listboxen**).
- Ab 4D v14 können Sie über die Objekte **PopUp-Menüs**, **Dropdown-Listen** und **Combo Boxen** ein Feld bzw. eine Variable mit Liste direkt im Formular anzeigen.

Variablentyp

Über die Eigenschaft **Variablentyp** unter der Gruppe "Objekte" der Eigenschaftensliste können Sie den Datentyp für die Variable festlegen:



Diese Einstellung dient dazu, in der Eigenschaftensliste verfügbare Gruppen und Optionen zu konfigurieren, damit sie zum Datentyp passen. Die Variable selbst wird nicht typisiert. Beim Kompilieren der Datenbank müssen Sie die Befehle aus dem Kapitel **Compiler** verwenden.

In zwei spezifischen Fällen hat das DropDown-Menü **Variablentyp** jedoch eine Typisierungsfunktion:

- Bildvariablen: Sie können dieses Menü verwenden, um diese Variablen vor Laden des Formulars im interpretierten Modus zu deklarieren (siehe unten)
- Dynamische Variablen: Sie können dieses Menü verwenden, um den Typ der dynamischen Variablen anzugeben (siehe **Dynamische Variablen**).

Bildvariablen in Formularen typisieren

Spezifische native Arbeitsweisen steuern die Anzeige von Bildvariablen in Formularen. Das erfordert beim Erstellen der Variablen größere Genauigkeit als bei anderen Variablentypen. Sie müssen schon vor dem Laden des Formulars deklariert sein — z.B. bereits vor dem Formularereignis On Load.

Es gibt zwei Möglichkeiten:

- Sie führen die Anweisung **C_PICTURE**(varName) vor Laden des Formulars aus (in der Regel in einer Methode, die den 4D Befehl **DIALOG** aufruft),
- Oder Sie typisieren die Variable auf Formularebene in der Eigenschaftensliste im PopUp-Menü **Variablentyp**

Andernfalls wird die Bildvariable im interpretierten Modus nicht korrekt dargestellt.

Anzeige

- Variablen und Felder können beliebige Ausmaße haben. Wenn sie Zeichen anzeigen, kann die Größe des Bereichs in Schritten gemäß der verwendeten Schriftgröße variieren. Variablen und Felder können Anzeigeformate nutzen (siehe **Anzeigeformate**). Objekte vom Typ Text und Bild können horizontale und vertikale Rollbalken haben (siehe **Rollbalken**) und mit unterschiedlichem Rahmen gedruckt werden (siehe Abschnitt).
Hinweis: Textvariablen und Felder in Formularen werden angelegt, um den Inhalt in "vernünftiger" Länge anzuzeigen. Übersteigt er mehrere 10.000 Zeichen (abhängig vom System), ist nur ein Teil des Textes im Formular verfügbar.
- Ist die Eigenschaft **Mehrfachstil** aktiviert, sind für eingebbare Variablen und Felder vom Typ Text oder Alpha eigene Stilelemente möglich (zusätzlich zum allgemeinen Stil, der auf Objektebene definiert wird). Weitere Informationen dazu finden Sie unter **Text mit Stil (Rich Text)**.
- Enthält ein Datenfeld oder eine Variable vom Typ Alpha oder Text eine URL, d.h. ein String, der mit http, ftp, www oder mailto beginnt, wird sie automatisch hervorgehoben. Klickt der Benutzer unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** in den Bereich, wird die URL direkt im standardmäßigen Web Browser ausgeführt. Unter Windows erscheinen URLs in blau und unterstrichen:




Hinweis: E-Mail Adressen müssen die Form mailto:Adresse haben.

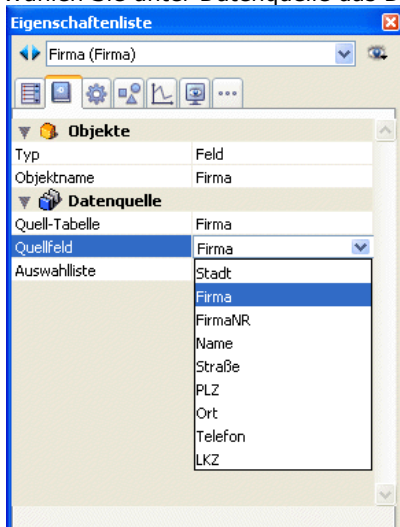
Einem Formular Datenfelder hinzufügen

Sie können in einem Formular jederzeit Datenfelder hinzufügen oder löschen. Sie fügen zum Beispiel Datenfelder hin, wenn:

- Sie feststellen, dass Sie ein Datenfeld benötigen, das Sie im Formularsistenten nicht ausgewählt haben.
- Sie der Datenbankstruktur ein Datenfeld hinzufügen. Dann müssen Sie es einem Formular hinzufügen, damit es verwendbar ist.

Um ein Datenfeld im Formular hinzuzufügen:

1. Wählen Sie in der Objektleiste auf das Symbol Datenfeld hinzufügen  und zeichnen den Bereich für das Feld im Formular.
4D zeigt automatisch die Eigenschaften des neuen Feldes in der Eigenschaftenliste.
2. Wählen Sie unter Datenquelle das Datenfeld, das Sie einfügen wollen:



Hinweis: Sie können kein Feld vom Typ BLOB auswählen.

3. Wählen Sie bei Bedarf spezifische Eigenschaften, die Sie diesem Datenfeld zuweisen wollen.
Ist das Datenfeld angelegt, müssen Sie normalerweise weitere Eigenschaften festlegen. Sie können

Eingabekontrollen festlegen, Hilfetexte schreiben, eine Methode zuordnen, Vergrößerungs- oder Verschiebeoptionen festlegen, die Plattform, Schrift oder Darstellungsoptionen festlegen (siehe **Eigenschaften für aktive Objekte**).
Hinweis: Sie können ein Feld auch per Drag-and-Drop aus der **Seite Tabellen** des Explorers einfügen.

Das neue Datenfeld erscheint im Formular dort, wo Sie den Bereich gezeichnet haben. Er zeigt den gewählten Datenfeldnamen mit vorangestelltem Tabellennamen.

4D fügt standardmäßig keine Bezeichnung für das Feld hinzu, Sie können jedoch über den statischen Textbereich eine anlegen. Beachten Sie, dass sich die Bezeichnung selbst auch dynamisch definieren lässt (siehe **Referenzen in statischem Text verwenden**).

Nachdem Sie das Datenfeld in das Formular eingefügt haben, können Sie es wie jedes andere Formularobjekt ändern. Sie können seine Größe oder die Schrift ändern, Farben für die Anzeige auf einem Farbmonitor auswählen, usw.

Datenfeld in Variable umwandeln und umgekehrt

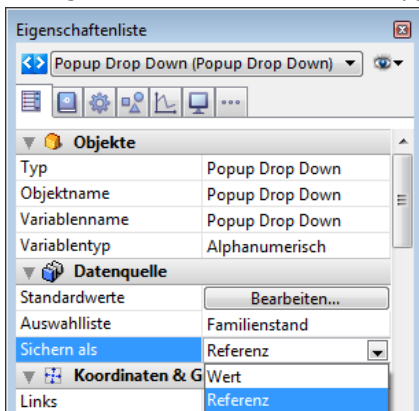
Sie können jeden Objekttyp (aktiv oder inaktiv) in einen anderen Objekttyp umwandeln. Sie können auch ein Feld in eine Variable und umgekehrt umwandeln. Das ist z.B. sinnvoll, wenn Sie in Ihrem Formular erst ein Datenfeld anlegen, und dieses dann in eine Variable umwandeln wollen, da Sie den Wert nicht speichern müssen. 4D behält beim Umwandeln die Eigenschaften des ursprünglichen Objekts bei, wie z.B. Koordinaten, Objektmethode, Darstellung oder Farbe. Der dem Feld zugewiesene Datentyp wird für die Variable beibehalten: Ein Feld vom Typ Bild wird in eine Bildvariable umgewandelt, usw.

Um eine Variable in ein Datenfeld oder ein Datenfeld in eine Variable umzuwandeln, wählen Sie das Objekt und dann in der Eigenschaftsliste unter der Gruppe *Objekte* in der Zeile Typ **Variable** oder **Feld**. Die Liste wird nun neu aufgebaut, um die Eigenschaften des neuen Objekttyps anzuzeigen. Objektname, Objektmethode und seine Eigenschaften (eingebbar, Größe, Farbe, etc.) bleiben erhalten.

Ändern Sie eine Variable in ein Feld, weist 4D dem Objekt standardmäßig das erste Feld in der ersten Tabelle zu. In der Gruppe *Datenquelle* unter **Quelltabelle** und **Quellfeld** können Sie Tabelle und Feld manuell definieren.

Sichern als

Ist einem Feld bzw. einer Variablen eine Auswahlliste und eine Liste mit erforderlichen Werten zugewiesen, können Sie in der Eigenschaftsliste unter der Gruppe "Datenquelle" die Option **Sichern als Wert/Referenz** verwenden.



Damit lässt sich die Größe der gespeicherten Daten optimieren. Weitere Informationen dazu finden Sie im Abschnitt **Als Wert oder Referenz sichern**.

Überblick

Über den Formulareditor können Sie eine breite Palette von Schaltflächen zu Ihrem Formular hinzufügen. Sie können jeder Schaltfläche eine Standardaktion zuweisen. Über solche Schaltflächen kann der Benutzer die Eingabe in Datensätze bestätigen oder annullieren, einen Datensatz löschen, zwischen Datensätzen wechseln, in einem mehrseitigen Formular in den Seiten blättern und Datensätze in einem Unterformular öffnen, löschen oder hinzufügen, Schrifteigenschaften in Textbereichen verwalten, etc.

Normalerweise fügen Sie Schaltflächen hinzu, wenn Sie ein Formular mit dem Formularassistenten erzeugen. Sie können diese Aktionen für Schaltflächen in der Eigenschaftenliste ändern. Sie können zum Beispiel die automatische Aktion einer Schaltfläche löschen und eine Objektmethode schreiben, die eine andere Aktion festlegt.

Sie können Schaltflächen auch mit dem Formulareditor hinzufügen und Schaltflächenaktionen zuweisen. Benötigen Sie zum Beispiel mehr als ein Unterformular in einem Formular, können Sie die zusätzlichen Unterformulare und automatischen Schaltflächen im Formulareditor hinzufügen. Sie fügen einfach jede Schaltfläche in das Formular ein und weisen jeder Schaltfläche die Standardaktion zu.

In 4D gibt es folgende Typen:

- **Schaltflächen:** Diese Schaltflächen werden mit der Oberfläche der aktuellen Plattform angezeigt. Der Text erscheint mit der ausgewählten Schrift, Schriftgröße, Farbe und dem ausgewählten Stil.



Die Bezeichnung wird in der Eigenschaftenliste unter der Gruppe „Objekte“ definiert. Sie können diese jederzeit ändern.

- **Standardschaltflächen:** Eine Standardschaltfläche sieht wie eine gängige Schaltfläche aus, nur die Darstellung ist anders. Dies zeigt dem Benutzer, dass die Schaltfläche die empfohlene Wahl ist. Der Unterschied hängt vom Betriebssystem ab. Nachfolgende Darstellung vergleicht eine Schaltfläche mit einer Standardschaltfläche.



Auf macOS sind Standardschaltflächen blau:



Der Objekttyp **Standardschaltfläche** als solcher existiert nicht, es ist aber eine Eigenschaft, die für Schaltflächen verfügbar ist.

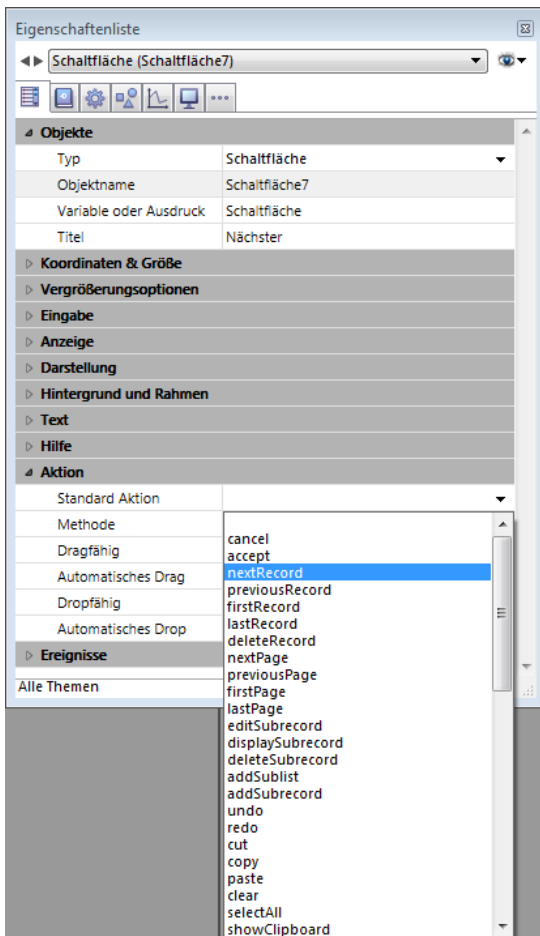
Hinweis: Es kann pro Formularseite immer nur eine Standardschaltfläche geben.

- **Invertierte und unsichtbare Schaltflächen:** Diese Schaltflächen sind zur Positionierung auf Grafikobjekten gedacht. Invertierte Schaltflächen sind erst bei Anklicken sichtbar. Wenn der Benutzer auf eine invertierte Schaltfläche klickt, wird die Schaltfläche hervorgehoben. Unsichtbare Schaltflächen bleiben auch bei Anklicken unsichtbar. Das Ergebnis der Aktion, z.B. das Öffnen einer anderen Seite, zeigt an, dass auf die Schaltfläche geklickt wurde. Eine unsichtbare Schaltfläche muss auf Text oder eine Grafik gelegt werden, die deren Funktion angeben; klickt der Benutzer auf den Text oder die Grafik, wird die Schaltfläche aktiviert.
- **3D Schaltflächen und Bildschaltflächen:** Die Familie der 3D Schaltflächen – 3D Schaltflächen, 3D Kontrollkästchen und 3D Optionsfelder, sowie Bildschaltflächen enthalten eine ganze Reihe spezifischer Eigenschaften. Weitere Informationen dazu finden Sie in den Abschnitten **3D Schaltflächen**, **3D Kontrollkästchen**, **3D Optionsfelder** und **Bildschaltflächen**.

Schaltflächenaktionen verwalten

Schaltflächen mit Standardaktionen werden bei Bedarf während der Ausführung des Formulars grau dargestellt. Wird zum Beispiel der erste Datensatz in einer Tabelle angezeigt, ist die Schaltfläche mit der Standardaktion **Erster Datensatz** grau dargestellt.

Sie erzeugen eine Schaltfläche durch Auswählen des gewünschten Schaltflächentyps in der Dropdown-Liste **Typ**. Anschließend wählen Sie in der Dropdown-Liste Standardaktion die gewünschte automatische Aktion aus (siehe **Standardaktionen**):



Soll die Schaltfläche eine Aktion ausführen, die nicht als Standardaktion verfügbar ist, lassen Sie das Feld Standardaktion leer und schreiben eine Objektmethode, um die Schaltflächenaktion festzulegen. Weitere Informationen dazu finden Sie im Abschnitt **Methoden bearbeiten**.

In der Regel wählen Sie unter der Gruppe „Ereignisse“ das Ereignis On Clicked aus, so dass die zugeordnete Methode nur bei Anklicken der Schaltfläche ausgeführt wird. Sie können jeder Schaltfläche eine Methode zuordnen.

Alle Variablen, die mit Schaltflächen verknüpft sind (Standard-, invertierte, unsichtbare Schaltflächen, Optionsfelder, Optionsbilder oder Kontrollkästchen) werden auf 0 gesetzt, wenn das Formular in der Design - oder Anwendungsumgebung zum ersten Mal geöffnet wird. Klickt der Benutzer auf eine Schaltfläche, wird die Variable auf 1 gesetzt.

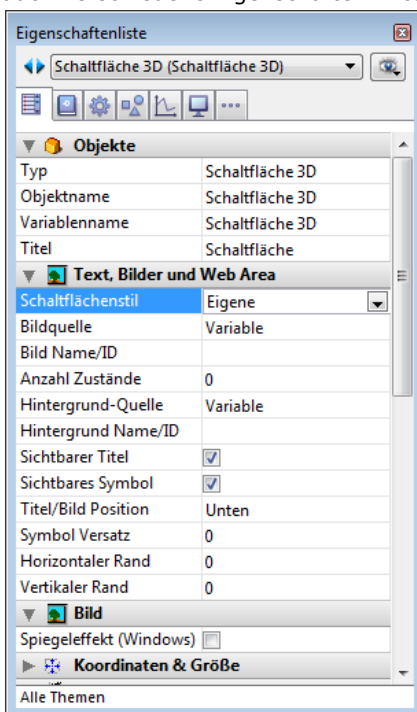
Die Beschreibung der einzelnen Aktionen finden Sie unter **Standardaktionen**.

3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder

Die 3D Schaltflächenfamilie enthält 3D Schaltflächen, 3D Kontrollkästchen und 3D Optionsfelder. Diese Objekte sind von der Struktur her gleich, sie unterscheiden sich nur in der Verwendung der zugewiesenen Variable:

- Die einer Schaltfläche zugeordnete Variable hat beim Öffnen des Formulars den Wert 0 (Standardstatus). Klickt der Benutzer darauf (gedrückter Status), hat sie den Wert 1; sie kehrt dann wieder zum Standardstatus zurück und hat wieder den Wert 0.
- Die einem Kontrollkästchen zugeordnete Variable hat den Wert 0, wenn das Kästchen nicht markiert ist, den Wert 1, wenn es markiert ist. Im Unterschied zu Schaltflächen behält das Kontrollkästchen den Status 0 oder 1 bei, bis der Benutzer es erneut auswählt. Weitere Informationen dazu finden Sie im Abschnitt **Kontrollkästchen**.
- Optionsfelder funktionieren gruppenweise. Die dem ausgewählten Feld zugewiesene Variable hat den Wert 1, die anderen haben den Wert 0. Beim Öffnen des Formulars kann die Variable den Wert 0 oder 1 haben. Weitere Informationen dazu finden Sie im Abschnitt **Optionsfelder und Optionsbilder**.

Die 3D Schaltflächenfamilie bietet eine ganze Reihe spezifischer Eigenschaften für Oberflächen, die sich perfekt an die verschiedenen Betriebssysteme anpassen. Es ist vorallem möglich, verschiedene vordefinierte Arten für 3D Schaltflächen (Bevel Buttons, Push Buttons, etc.) anzuwenden, oder PopUp-Menüs zuzuordnen. Über die Eigenschaftliste lassen sich auch verschiedene Eigenschaften miteinander kombinieren.



Titel

Damit können Sie in der Schaltfläche eine Bezeichnung einfügen. Schriftart und -stil können Sie unter der Gruppe "Text" definieren. Mit dem Zeichen \ (umgekehrter Schrägstrich) können Sie in der Bezeichnung eine Zeilenschaltung erzwingen.

Voransicht
Zeigen

Sie müssen den Schrägstrich zweimal eingeben (\\).

Die Bezeichnung wird standardmäßig zentriert angeordnet. Bei einer Schaltfläche mit Icon können Sie die relative Position beider Elemente über die Eigenschaft **Titel/Bildposition** bestimmen.

Wollen Sie den Titel ausblenden, deaktivieren Sie die Eigenschaft **Titel sichtbar**. Dann wird das Icon automatisch in die Mitte der Schaltfläche gelegt.

Zur Lokalisierung der Datenbank in andere Sprachen, können Sie im Titelbereich der Schaltfläche eine XLIFF- oder STR#-Referenz eingeben (siehe **PICTURE TO BLOB**).

Schaltflächenstil

Mit dieser Schaltfläche legen Sie die allgemeine Darstellung der Schaltfläche fest. Bei bestimmten Optionen ist der Stil ausschlaggebend für die Verfügbarkeit. Es gibt folgende Stilarten (Darstellung unter Windows bzw. macOS):

- **Keine** (Standardwert).



Eine 3D Schaltfläche mit dem Stil "Keine" ähnelt einer unsichtbaren Schaltfläche, d.h. die Markierung wird nicht optisch angezeigt. Er nutzt jedoch die Optionen für 3D Schaltflächen.
Auf macOS lässt sich kein Dreieck anzeigen, das auf ein verknüpftes DropDown-Menü hinweist.

- **Hintergrund Versatz**



Dieser Stil entspricht den invertierten Schaltflächen mit dem Unterschied, dass beim Klicken auf diesen Typ der 3D Effekt durch Verschieben des Bildes unter der Schaltfläche erzielt wird.
Ist dieser Stil gewählt, sind keine Optionen verfügbar.

- **Push Button**



Eine 3D Schaltfläche mit dem Stil "Push button" erscheint als standardmäßige Systemschaltfläche. Sie kann jedoch die Optionen für 3D Schaltflächen nutzen, mit Ausnahme der Eigenschaft "Mit PopUp-Menü".

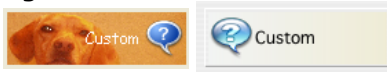
- **Toolbar Button**



Damit integrieren Sie die Schaltfläche in eine Werkzeugleiste (toolbar). Unter Windows wird sie beim Darüberziehen der Maus hervorgehoben. Ist die Eigenschaft "Mit PopUp Menü" zugewiesen, erscheint am Rand der Schaltfläche rechts mittig ein Dreieck.

Auf macOS wird die Schaltfläche nicht hervorgehoben. Ist die Eigenschaft "Mit PopUp Menü" zugewiesen, erscheint am Rand der Schaltfläche rechts unten ein Dreieck.

- **Eigene**



Damit können Sie den Hintergrund und andere Parameter (Icon und Randversatz) selbst festlegen. Weitere Informationen dazu finden Sie im unteren Abschnitt "Eigene 3D Schaltflächen".
Dieser Stil kann die Optionen für 3D Schaltflächen nutzen, mit Ausnahme der Eigenschaft "Mit PopUp-Menü".

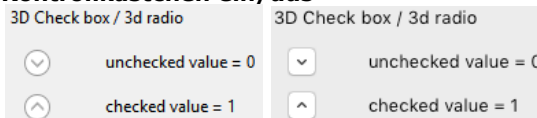
- **Kreis**



Auf macOS erscheint eine 3D Schaltfläche mit Kreis als runde Systemschaltfläche. Es gibt zwei Größen, die sich nach der Schaltflächengröße im Formular richten.

Dieser Stil kann die Optionen für 3D Schaltflächen nutzen, mit Ausnahme der Eigenschaft "Mit PopUp-Menü".
Unter Windows ist er identisch mit dem Stil "Keine", d.h. der Kreis im Hintergrund wird nicht berücksichtigt.

- **Kontrollkästchen ein/aus**



Auf macOS und Windows erscheint eine 3D Schaltfläche mit dem Stil "Kontrollkästchen" als Systemschaltfläche, in der Regel um zusätzliche Information ein/auszublenden.

Beim Verwenden als 3D Schaltfläche oder Optionsfeld zeigt das Symbol mit Wert 0 nach unten, mit Wert 1 nach oben.

Mit diesem Stil sind keine Optionen verfügbar.

- **Office XP**



Eine 3D Schaltfläche mit dem Stil "Office XP" hat folgende Merkmale:

- Die Farbe für Hervorhebung und Hintergrund richtet sich nach den Farben des Betriebssystems.
- Unter Windows erscheint die Hervorhebung nur, wenn die Maus darübergezogen wird.

Dieser Stil kann die Optionen für 3D Schaltflächen nutzen.

- **Bevel**



Auf macOS erscheint eine Schaltfläche vom Typ "Bevel" als standardmäßige Systemschaltfläche. Sie kann die Optionen für 3D Schaltflächen nutzen, inkl. der Eigenschaft „Mit PopUp-Menü“.

Unter Windows ähnelt dieser Stil dem Stil "Toolbar Button", mit dem Unterschied, dass das Dreieck zum Aufklappen des zugeordneten PopUp-Menüs in der Schaltfläche in der rechten unteren Ecke liegt.

- **Abgerundeter Bevel**



Auf macOS ähnelt er dem Stil "Bevel" mit dem Unterschied, dass die Ecken abgerundet sind. Unter Windows ist dieser Stil identisch mit dem Stil "Bevel".

- **Zuklappen/Aufklappen**



Damit können Sie einen standardmäßigen Icon für Auf-/Zuklappen hinzufügen. Diese Schaltflächen werden in hierarchischen Listen nativ verwendet. Unter Windows erscheint er als [+] oder [-]; auf macOS als nach rechts bzw. nach unten gerichtetes Dreieck. Dieser Stil dient nur für 3D Kontrollkästchen, wo die beiden Stadien der Schaltfläche dem Status gewählt/nicht gewählt des Kontrollkästchens entsprechen.

- **Hilfe**



Damit können Sie eine standardmäßige Hilfe-Schaltfläche des Systems anzeigen. So können Sie in Ihren Formularen Hilfe-Schaltflächen des Systems hinzufügen.

- **OS X Textured**



Auf macOS ist eine Schaltfläche vom Typ "Textured" eine standardmäßige Systemschaltfläche mit Graustufen. Ihre Höhe ist vorgegeben, sie kann weder vergrößert, noch verkleinert werden. Dieser Stil kann alle für 3D Schaltflächen verfügbaren Optionen nutzen.

Unter Windows entspricht dieser Stil einer Push-Schaltfläche, die auch ein PopUp-Menü haben kann. Sie hat unter Vista den speziellen Stil transparent.

- **OS X Gradient**

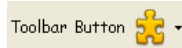


Auf macOS ist eine "Gradient" Schaltfläche eine Systemschaltfläche mit zwei Farbstufen. Dieser Stil kann alle für 3D Schaltflächen verfügbaren Optionen nutzen.

Unter Windows entspricht dieser Stil einer Push-Schaltfläche, die auch ein PopUp-Menü haben kann.

Der 3D Schaltfläche ein Icon zuordnen

Sie können jedem Stil für eine 3D Schaltfläche ein Icon zuordnen, mit Ausnahme von "Hintergrund Versatz".



Die zugewiesenen Icons lassen sich mit den Eigenschaften **Bildquelle**, **Bildname/ID** und **Versatz** verwalten.

- **Bildquelle**

Das Icon einer 4D Schaltfläche kann, wie für Bildschaltflächen, aus folgenden Quellen stammen: **Variable**, **Bildbibliothek**, **Ressourcendatei** oder **Datei**. Ist die Quelle gesetzt, können Sie Bildname oder -nummer in der Eigenschaft "Bildname/ID" angeben.

- **Bildname/ID**

Ist die Bildquelle gewählt, geben Sie hier den Namen ein, wenn das Bild eine Variable ist oder aus der Bildbibliothek stammt, die Nummer, wenn es aus der Bildbibliothek oder einer Ressourcendatei stammt, oder den Pfadnamen des Bildes in diesem Bereich, wenn es aus einer Bilddatei stammt; in diesem Fall muss der Pfad zum Ordner **Resources** der Datenbank passen. Weitere Informationen dazu finden Sie im Abschnitt **Automatisches Verweisen auf Bilder**. Tragen Sie 0 (Null) ein, wenn Sie kein Bild hinzufügen wollen.

Hinweis: Sie können einer Schaltfläche ein Bild per Drag-and-Drop aus der Bildbibliothek oder von der Festplatte zuordnen.

- **Anzahl Zustände**

Damit können Sie die genaue Anzahl der Stadien setzen, die im Bild vorhanden sind, das als Icon für 3D Schaltflächen verwendet wird. Im Quellbild müssen die Stadien vertikal gestapelt sein:



- **Sichtbarer Titel/Sichtbares Symbol**

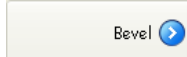
Bei Schaltflächen mit Titel und Bild können sie ein bzw. beide Elemente ausblenden, indem Sie die jeweilige Option deaktivieren. Ist ein Element ausgeblendet, wird es automatisch durch das andere in der Mitte der Schaltfläche ersetzt.

Titel/Bildposition

Damit können Sie die relative Position des Schaltflächentitels in Bezug auf das zugewiesene Icon definieren. Diese Eigenschaft hat keine Auswirkung, wenn die Schaltfläche nur einen Titel oder nur ein Bild enthält. Bei einer 3D Schaltfläche mit Titel und Bild erscheint der Text standardmäßig über dem Bild.

Im folgenden sehen Sie das Ergebnis der verschiedenen Möglichkeiten:

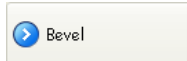
- **Links:** Der Text liegt links vor dem Icon. Beide Teile sind rechtsbündig ausgerichtet.



- **Oben:** Der Text liegt über dem Icon. Beide Teile sind zentriert.



- **Rechts:** Der Text liegt rechts vor dem Icon. Beide Teile sind linksbündig ausgerichtet.



- **Unten:** Der Text liegt unter dem Icon. Beide Teile sind zentriert.



- **Zentriert:** Der Text des Icon ist in der Schaltfläche vertikal und horizontal zentriert. Dieser Parameter ist hilfreich für Text innerhalb eines Icon.



Mit PopUp Menü

Damit zeigen Sie in der 3D Schaltfläche ein Symbol in Form eines Dreiecks an, das auf ein zugeordnetes PopUp-Menü hinweist:



Auftreten und Position dieses Symbols richten sich nach dem Stil der Schaltfläche und der aktuellen Plattform. Folgende Stilarten erlauben die Eigenschaft "Mit PopUp-Menü":

- Keine
- Toolbar Button
- Bevel
- Bevel abgerundet
- Office XP
- OSX Textured
- OSX Gradient

Verknüpft und Getrennt

Es gibt zwei Optionen, um einer 3D Schaltfläche ein Symbol für PopUp-Menü zuzuweisen: **Verknüpft** und **Getrennt**.



Hinweis: Ob die Option "getrennt" tatsächlich verfügbar ist, richtet sich nach dem Stil der Schaltfläche und der Plattform. Jede Option gibt die Verbindung zwischen der Schaltfläche und dem zugeordneten PopUp-Menü an:

- Bei getrenntem PopUp-Menü wird bei Mausklick auf den linken Teil der Schaltfläche direkt die aktuelle Aktion der Schaltfläche ausgeführt; über das PopUp-Menü im rechten Teil können Sie die Aktion ändern.
- Bei verknüpftem PopUp-Menü wird bei Mausklick nur das PopUp-Menü angezeigt. Sie müssen erst eine Aktion wählen, damit sie ausgeführt wird.

Diese Optionen beeinflussen auch die Verwaltung von Formularereignissen für Schaltflächen. Weitere Informationen dazu finden Sie unter im Handbuch *4D Programmiersprache*.

PopUp-Menü verwalten

Beachten Sie, dass die Eigenschaft "Mit PopUp-Menü" nur die grafische Darstellung verwaltet. Anzeige und Werte des PopUp-Menüs muss komplett der Entwickler verwalten und zwar über Formularereignisse und die 4D Funktionen **Dynamic pop up menu** und **Pop up menu**.

Eigene 3D Schaltflächen

Haben Sie die Eigenschaft "Eigene" gewählt, sind zusätzliche Optionen verfügbar: **Hintergrund-Quelle**, **Hintergrund Name/ID**, **Symbol Versatz**, **Horizontaler Rand** und **Vertikaler Rand**.

- **Hintergrund-Quelle:** Damit können Sie das Bild für den Hintergrund der Schaltfläche setzen. Sie können, wie für Icons, angeben, ob das Bild von einer Variablen, einer Bildbibliothek oder einer Ressourcendatei bzw. Datei stammt.
- **Hintergrund Name/ID:** Liegt die Quelle fest, tragen Sie hier Bildname bzw. -nummer ein. Hintergrundbilder können, wie für Icons, vier unterschiedliche vertikale Bereiche enthalten, über die 4D die vier standardmäßigen Zustände der Schaltflächen darstellt: Aktiv, Gedrückt, Rollover und Inaktiv. Beachten Sie, dass sich die durch Klick ausgelöste Wirkung auch über die Eigenschaft **Symbol Versatz** verwalten lässt.
- **Symbol Versatz:** Damit legen Sie einen eigenen Wert für Versatz in Pixel fest, der bei Anklicken der Schaltfläche verwendet wird: Der Name der Schaltfläche wird um die definierte Anzahl Pixel nach rechts und nach unten geschoben. Auf diese Weise erhalten Sie einen angepassten 3D Effekt, wenn auf die Schaltfläche geklickt wird.
- **Horizontaler / Vertikaler Rand:** Damit setzen Sie die Größe (in Pixel) des inneren Rahmens der Schaltfläche. Er zeigt die Begrenzung, innerhalb der Text und Icon liegen müssen. Dieser Parameter ist hilfreich, wenn z.B. das Hintergrundbild Ränder enthält:



Eigene 3D Schaltfläche ohne Rand



3D Schalfläche mit 13 Pixel Rand

Verwendung

Eine Bildschaltfläche kann verschiedene Statusmöglichkeiten haben. Bei einer Standardschaltfläche sind es drei Stadien: aktiviert, inaktiv und angeklickt. Wie der Name angibt, wird bei einer Bildschaltfläche jeder Status durch ein anderes Bild dargestellt.

Bildschaltflächen lassen sich auf zwei Arten verwenden:

- Als Steuerschaltfläche in einem Formular. In diesem Fall bietet die Schaltfläche in der Regel vier Stadien: aktiviert, inaktiv, angeklickt und darüberbewegt. Das ist die gängige Art für die meisten Formularvorlagen.
- Als Bildschaltfläche, die den Benutzer zwischen mehreren Optionen wählen lässt. In diesem Fall kann die Schaltfläche anstelle eines PopUp-Bildmenüs verwendet werden. Über **PopUp-Menü vom Typ Bild** erscheinen alle Wahlmöglichkeiten gleichzeitig (wie die Einträge im PopUp-Menü), während die Bildschaltfläche die Wahlmöglichkeiten nacheinander anzeigt (Wenn der Benutzer auf die Schaltfläche klickt).

Es folgt ein Beispiel für eine Bildschaltfläche. Angenommen, Sie wollen dem Benutzer einer eigenen Anwendung die Möglichkeit geben, die Sprache für die Anwendung auszuwählen.

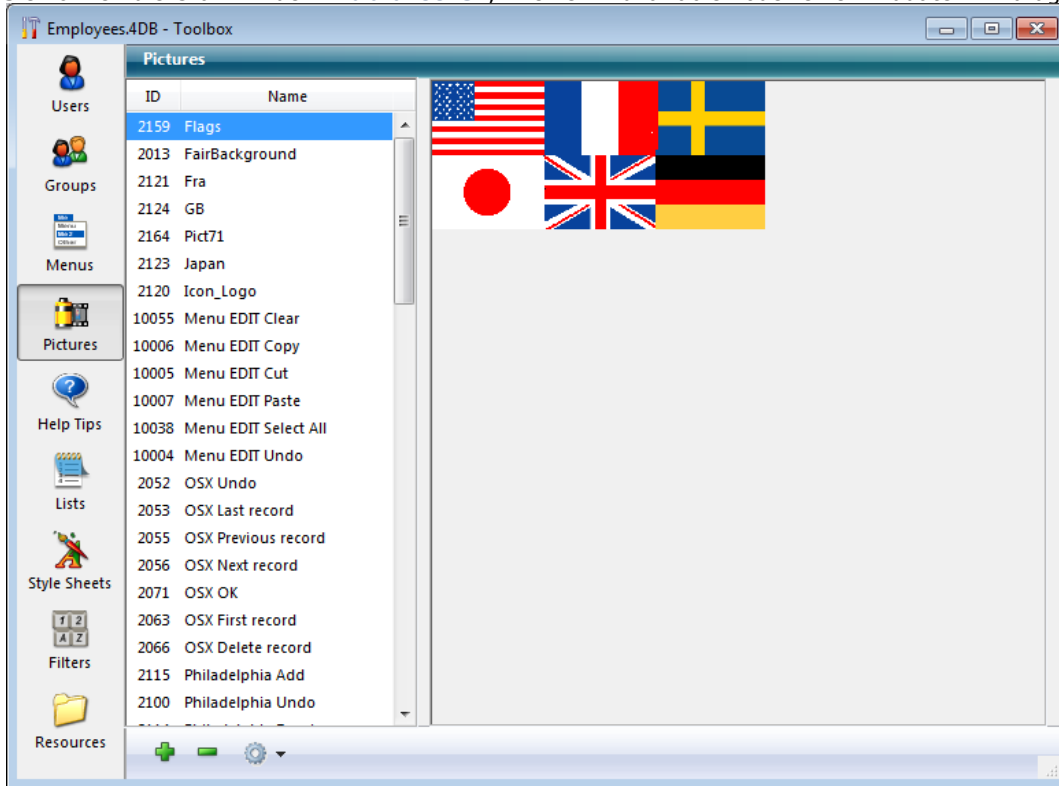
Sie binden die Option als Bildschaltfläche in Ihr eigenes Dialogfenster Eigenschaften wie folgt ein:



Durch Anklicken des Objekts ändert sich das Bild.

Sie binden die Bildschaltfläche folgendermaßen ein. Zuerst bereiten Sie eine Grafik vor, in der die Bilderserie in einem Zeilen-, Spalten- oder Zeilen-Spalten-Gitter angeordnet ist.

Sie können die Grafik in der **Bildbibliothek**, in einer Bildvariablen oder einer Bilddatei hinzufügen.



Legen Sie Bilder als Gitter an, werden sie zeilenweise von links oben nach rechts unten nummeriert, beginnend mit Null. Zum Beispiel hat das zweite Bild der zweiten Zeile in einem Gitter aus vier Zeilen und drei Spalten die Nummer 4.

Hinweis: Die **Bildbibliothek** bietet Hilfswerkzeuge, um ein Bild in mehrere Einheiten aufzuteilen. Sie können Ihre Einstellungen sofort im Vorschaubereich betrachten.

Wurde ein Bild als Tabelle definiert, können Sie direkt eine Bildschaltfläche erstellen. Ziehen Sie dazu den Bildnamen aus der Bildbibliothek in den Formulareditor.

Eigenschaften

Für Bildschaltflächen können Sie folgende Eigenschaften festlegen:

- Unter der Gruppe „Bild“ definieren Sie in der DropDown Liste „Quelle“ die Herkunft des Bildes. Sie können wählen zwischen Variable, Bildbibliothek, Ressourcendatei oder Datei. Ist die Bildquelle gewählt, geben Sie im Bereich „Name/ID“ den Namen ein, wenn das Bild eine Variable ist oder aus der Bildbibliothek stammt, die Nummer, wenn es aus der Bildbibliothek oder einer Ressourcendatei stammt, oder den Pfadnamen des Bildes, wenn es aus einer

Bilddatei stammt; in diesem Fall muss der Pfad zum Ordner **Resources** der Datenbank passen. Weitere Informationen dazu finden Sie im Abschnitt **Automatisches Verweisen auf Bilder**.

- Unter der Gruppe „Abschneiden“ legen Sie die Anzahl der Zeilen und Spalten der Bildertabelle fest.
- Unter der Gruppe „Animation“ legen Sie die Darstellung und Funktionsweise der Bildschaltfläche fest. Sie können auch mehrere Optionen miteinander kombinieren. Mehr dazu erfahren Sie im folgenden Abschnitt.

Hinweis: Die einer Bildschaltfläche zugeordnete Variable gibt in der Tabelle mit Thumbnails die Nummer des aktuell angezeigten Bildes zurück. Die Nummerierung beginnt mit Null.

Animation

Sie können für Bildschaltflächen verschiedene Darstellungen und Funktionsweisen setzen und auch miteinander kombinieren:

- **<Keine Option>**
Zeigt das nächste Bild in einer Serie, wenn der Benutzer darauf klickt. Zeigt das vorherige Bild in der Serie, wenn der Benutzer bei gedrückter **Umschalttaste** darauf klickt. Wenn der Benutzer das letzte Bild in der Serie erreicht, ändert sich das Bild nicht bei erneutem Klick, d.h. es wird nicht zum ersten Bild in der Serie zurückgeblättert.
- **Wechseln solange geklickt**
Ähneln der vorigen Option, außer dass der Benutzer die Maustaste gedrückt halten kann, um die Bilder durchlaufen zu lassen (wie in einer Animation). Ist das letzte Bild erreicht, stoppt der Durchlauf.
- **Zurück zum ersten Bild**
Ähneln der vorigen Option, außer dass die Bilder in einer Endlosschleife angezeigt werden. Wenn der Benutzer das letzte Bild erreicht und erneut klickt, erscheint wieder das erste Bild usw..
- **Wechseln, wenn Maus darüber**
Wechselt den Inhalt der Bildschaltfläche, wenn die Maus ohne Klicken darüberggezogen wird. Das Ausgangsbild erscheint wieder, wenn der Cursor die Schaltfläche verlässt. Dieser Modus, auch „rollover“ genannt, wird häufig in Multimedia-Anwendungen oder HTML Dokumenten verwendet. Das angezeigte Bild ist das letzte der Bildertabelle, außer die Option „Benutze letztes Bild, wenn inaktiv“ ist auch ausgewählt – in diesem Fall dient das vorletzte Bild als „Wechsler“.
- **Zurückwechseln, wenn losgelassen wird**
Dieser Modus funktioniert mit zwei Bildern; er gibt an, dass die Schaltfläche immer das erste Bild anzeigen soll, außer der Benutzer klickt darauf. Die Schaltfläche zeigt also standardmäßig Bild A, bei Mausklick Bild B, wird die Maustaste losgelassen wieder Bild A. Mit diesem Modus erstellen Sie eine Aktionsschaltfläche, die je nach Status (normal oder gedrückt) ein anderes Bild zeigt. So können Sie einen eigenen 3D Effekt erstellen.
- **Benutze letztes Bild wenn inaktiv**
Dieser Modus legt fest, dass bei inaktiver Schaltfläche das letzte Bild der Serie benutzt wird. Kombinieren Sie diese Option mit den Werten 0, 1 oder 2, wird das letzte Bild aus der Sequenz für die Bildschaltfläche ausgeschlossen. Es erscheint nur bei inaktiver Schaltfläche.
- **Wechseln alle n-Ticks**
Die Bilder werden in einer Endlosschleife in der festgelegten Geschwindigkeit (Ticks) angezeigt. Geben Sie z.B. 10 ein, wechselt der Inhalt der Bildschaltfläche alle 10 Ticks. In diesem Modus werden alle anderen Optionen ignoriert.

Hinweis: Mit der Option „Transparent“ unter der Gruppe „Darstellung“ können Sie den Hintergrund der Bildschaltfläche transparent machen.

Beispiel: Sie wollen eine Bildschaltfläche definieren mit den Optionen „Zurückwechseln, wenn losgelassen wird“, „Wechseln, wenn Maus darüber“ und „Benutze letztes Bild, wenn inaktiv“.

Sie haben eine Bildertabelle mit einer Zeile und 4 Spalten. Jedes Bild entspricht in dieser Reihenfolge einem anderen Status: „standardmäßig aktiv“, „angeklickt“, „Cursor geht darüber“ und „inaktiv“.

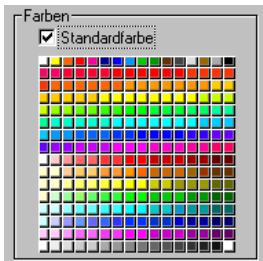
Weisen Sie in der Eigenschaftsliste folgende Eigenschaften zu: 1 Zeile, 4 Spalten, Option **Wechseln, wenn Maus darüber**, **Zurückwechseln, wenn losgelassen wird** und **Benutze letztes Bild, wenn inaktiv**.

Schaltflächengitter

Verwendung

Ein Schaltflächengitter ist ein transparentes Objekt, das auf eine Grafik gelegt wird. Sie muss die Form eines Array vom Typ Zeile-nach-Spalte haben. Sie verwenden ein Schaltflächengitter, um festzulegen, wo der Benutzer auf die Grafik klickt. Ihre Objektmethode verwendet das Ereignis On Clicked und verwaltet die Aktion je nach der Stelle des Klicks.

4D verwendet z.B. ein Schaltflächengitter als Farbpalette:



Um ein Schaltflächengitter zu erzeugen, fügen Sie dem Formular eine Hintergrundgrafik hinzu und setzen auf das Bild ein Schaltflächengitter.

Unter der Gruppe „Abschneiden“ geben Sie die Anzahl der Zeilen und Spalten des Gitters ein.

Aktion Gehe zu Seite

Sie können einem Objekt vom Typ Schaltflächengitter die Aktion **Gehe zu Seite** zuordnen. Ist diese Aktion aktiviert, zeigt 4D automatisch die Formularseite, die der Nummer des gewählten Elements im Schaltflächengitter zugeordnet ist. Wählt der Benutzer z.B. das 10. Element des Gitters, zeigt 4D - sofern vorhanden- Seite 10 des aktuellen Formulars. Wollen Sie selbst steuern, was beim Auswählen eines Elements passiert, behalten Sie die Standardoption **Keine Aktion** bei.

Weitere Informationen finden Sie unter [Standardaktionen](#).

Kontrollkästchen

Verwendung

Ein Kontrollkästchen wird zur Eingabe oder Anzeige binärer Daten verwendet, wie z.B. Wahr/Falsch. Es ist eine Art Schaltfläche, die entweder aktiviert oder deaktiviert ist. Das Verhalten eines Kontrollkästchens wird durch eine Methode gesteuert. Wie alle Schaltflächen wird ein Kontrollkästchen auf 0 gesetzt, wenn das Formular zum ersten Mal geöffnet wird. Die zugeordnete Methode wird ausgeführt, wenn das Kontrollkästchen markiert wird.

Ein Kontrollkästchen zeigt neben dem Kästchen Text an. Der Text wird in der Eigenschaftenliste unter der Gruppe „Objekte“ definiert. Hier können Sie auch eine XLIFF-Referenz eingeben. Weitere Informationen dazu finden Sie im Abschnitt [Anhang C: XLIFF Architektur](#).

Klickt der Benutzer auf das Kontrollkästchen, wird es markiert. Ist es aktiviert, hat es den Wert 1, ist es nicht aktiviert, hat es den Wert 0.

Sie können Kontrollkästchen auch eine Variable vom Typ Boolean zuweisen. In diesem Fall ist die Variable wahr, wenn das Kästchen markiert ist und falsch, wenn es nicht markiert ist.

Jedes Kontrollkästchen in einem Formular hat einen dieser Zustände:



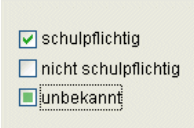
Bei einer Gruppe kann der Benutzer mehrere Optionen auswählen.

Im Unterschied zu einem Booleschen Datenfeld, das als Kontrollkästchen formatiert ist, werden die Werte der Variablen für das Kontrollkästchen nicht automatisch gespeichert. Zum Verwalten der Variablen verwenden Sie eine Methode.

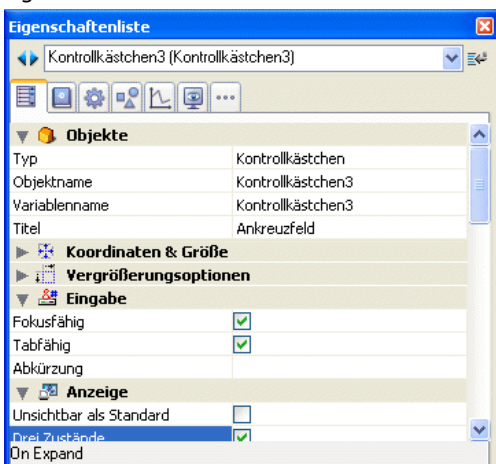
Hinweis: 3D Kontrollkästchen funktionieren auf dieselbe Art wie Kontrollkästchen. Ihre Darstellung hingegen, also z.B. der Status markiert/nicht markiert, wird über die Eigenschaften der 3D Schaltflächenfamilie definiert. Weitere Informationen dazu finden Sie im Abschnitt [3D Schaltflächen](#), [3D Kontrollkästchen](#), [3D Optionsfelder](#).

Kontrollkästchen mit drei Zuständen

Objekte vom Typ Kontrollkästchen lassen einen 3. Status zu. Das ist ein Zusatzstatus, der in der Regel für die Anzeige verwendet wird. Damit können Sie z.B. einzelnen Objekten eine bestimmte Eigenschaft zuweisen, die nicht für alle Objekte gelten soll.



Damit ein Kontrollkästchen diesen 3. Status steuert, müssen Sie in der Eigenschaftenliste unter dem Thema "Anzeige" die Eigenschaft **Drei Zustände** aktivieren:



Diese Eigenschaft ist nur für standardmäßige Kontrollkästchen mit numerischen Variablen möglich – Kontrollkästchen vom Typ 3D oder für Boolean Felder können diese Eigenschaft nicht verwenden. Ein Feld vom Typ Boolean lässt keinen 3. Zustand zu.

Die dem Kontrollkästchen zugeordnete Variable gibt den Wert 2 zurück, wenn der 3. Zustand vorliegt.

Tipp: Kontrollkästchen mit drei Zuständen zeigen jeden Status sequentiell in folgender Reihenfolge an: nicht-markiert/ markiert/ alternativ/ nicht-markiert, etc. Der 3. Zustand ist im Eingabemodus in der Regel nicht sinnvoll; setzen Sie deshalb im Code die Variable auf 0 (Null), wenn sie den Wert 2 annimmt, um direkt vom Status markiert in den Status nicht-markiert zu gelangen.

Eine Standardaktion verwenden

Sie können einem Kontrollkästchen bzw. einem 3D Kontrollkästchen eine Standardaktion zum Verwalten von Attributen in Textbereichen zuweisen - entweder über den Befehl **OBJECT SET ACTION** oder über die Gruppe "Aktion" der

Eigenschaftensliste. Wählen Sie z.B. die Standardaktion "fontBold", verwaltet das Kontrollkästchen in Echtzeit das Attribut "fett" des ausgewählten Textes im aktuellen Bereich.

Hinweis: Die Option Drei Zustände wird nur für Kontrollkästchen, nicht für 3D Kontrollkästchen unterstützt.

Dieses Objekt unterstützt nur Aktionen, die einen Status wahr/falsch darstellen können (Aktionen zum Ankreuzen):

Unterstützte Aktionen	Anwendungsbereich (falls vorhanden)
avoidPageBreakInsideEnabled	nur 4D Write Pro Areas
fontItalic	
fontBold	
fontLinethrough	
fontSubscript	nur 4D Write Pro Areas
fontSuperscript	nur 4D Write Pro Areas
fontUnderline	
font/showDialog	Nur Mac
htmlWYSIWIGEnabled	nur 4D Write Pro Areas
section/differentFirstPage	nur 4D Write Pro Areas
section/differentLeftRightPages	nur 4D Write Pro Areas
spell/autoCorrectionEnabled	
spell/autoDashSubstitutionsEnabled	Nur Mac
spell/autoLanguageEnabled	Nur Mac
spell/autoQuoteSubstitutionsEnabled	Nur Mac
spell/autoSubstitutionsEnabled	
spell/enabled	
spell/grammarEnabled	Nur Mac
spell/showDialog	Nur Mac
spell/visibleSubstitutions	
visibleBackground	nur 4D Write Pro Areas
visibleFooters	nur 4D Write Pro Areas
visibleHeaders	nur 4D Write Pro Areas
visibleHiddenChars	nur 4D Write Pro Areas
visibleHorizontalRuler	nur 4D Write Pro Areas
visiblePageFrames	nur 4D Write Pro Areas
visibleReferences	
widowAndOrphanControlEnabled	nur 4D Write Pro Areas

Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

Optionsfelder und Optionsbilder

Optionsfelder und Optionsbilder sind Objekte, die dem Benutzer ermöglichen, aus einer Gruppe von Schaltflächen oder Bildern einen Wert oder ein Bild auszuwählen. Ein Optionsfeld zeigt ein kleines Bullauge und Text. Optionsbilder zeigen ein Symbol oder Bild und werden auf ein Bild gelegt.

Hinweis: 3D Optionsfelder funktionieren auf dieselbe Art wie Optionsfelder. Ihre Darstellung hingegen wird über die Eigenschaften der 3D Schaltflächenfamilie definiert. Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder**.

Jede Art von Optionsfeld wird durch Anklicken ausgewählt. Ist ein Optionsbild ausgewählt, können Sie es durch Anklicken wieder abwählen; das ist bei einem Optionsfeld nicht möglich.

Optionsbilder sind wie invertierte Schaltflächen im inaktiven Zustand transparent. Bei Auswahl wird das dahinterliegende Bild hervorgehoben, bis es wieder deaktiviert oder ein anderes Optionsbild aus derselben Gruppe aktiviert wird.

Optionsfelder und -bilder werden in Form aufeinander abgestimmter Elemente verwendet. Es kann immer nur ein Feld ausgewählt werden. Damit Optionsfelder koordiniert funktionieren, müssen sie gruppiert werden. Dazu gehen Sie in den Formulareditor und wählen im Menü **Objekt** den Befehl **Gruppieren** oder in der Werkzeugleiste die entsprechende Schaltfläche des Formulars:



In bisherigen Versionen von 4D war koordiniertes Verhalten einer Gruppe von Optionsfeldern möglich, wenn die zugewiesenen Variablen denselben Anfangsbuchstaben hatten, z.B. m_button1, m_button2, m_button3. Dieses Prinzip wird zur Wahrung der Kompatibilität in konvertierten Datenbanken standardmäßig beibehalten. Sie können aber auch die Verwendung der neuen Arbeitsweise über die Datenbank-Eigenschaften vorschreiben (siehe **Seite Kompatibilität**).

Das Verhalten der Optionsfelder wird über Methoden gesteuert. Wie alle Schaltflächen wird ein Optionsfeld auf 0 gesetzt, wenn das Formular zum ersten Mal geöffnet wird. Die zugeordnete Methode wird ausgeführt, wenn das Optionsfeld ausgewählt ist.

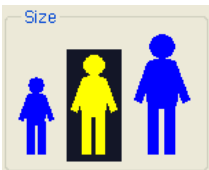
Hier ein Beispiel für einen Satz 3D Optionsfelder in einer Anwendung Video-Sammlung, um die Aufnahmegeschwindigkeit (SP, LP oder EP) einzutragen.



Die Auswahl eines Optionsfelds in einer Gruppe setzt dieses Optionsfeld auf 1 und die anderen in der Gruppe auf 0. Es kann immer nur ein Optionsfeld ausgewählt werden.

Hinweis: Sie können Optionsfeldern auch Variablen vom Typ Boolean zuweisen. Wird in diesem Fall ein Optionsfeld in einer Gruppe ausgewählt, ist dessen Variable wahr und die Variablen für alle anderen Optionsfelder sind falsch.

Hier ein Beispiel für einen Satz Optionsbilder. Das ausgewählte Bild erscheint invertiert:



Der Wert eines Optionsfeldes bzw. -bildes wird nicht automatisch gesichert, außer bei Darstellung eines Feldes vom Typ Boolean. Die Werte müssen in Variablen gespeichert und über Methoden verwaltet werden.

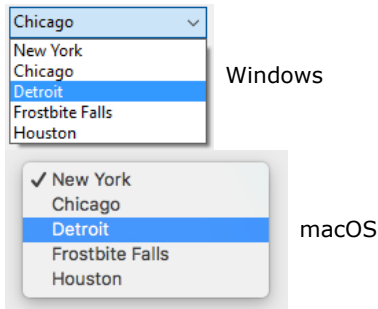
PopUp-Menüs, Dropdown-Listen

Definition

Über PopUp-Menüs und Dropdown-Listen kann der Benutzer ein Element aus einer Liste auswählen. Sie verwalten die Einträge in einem PopUp-Menü über ein Array, eine Auswahlliste oder eine Standardaktion.

Die Namen "PopUp-Menü" und "DropDown-Liste" bezeichnen dasselbe Objekt; PopUp-Menü ist die Bezeichnung auf macOS, DropDown-Liste die Bezeichnung unter Windows.

Wie folgendes Beispiel zeigt, ist die Darstellung dieser Objekte je nach Plattform etwas anders:



Ein Array verwenden

Ein Array ist eine Liste von Werten im Speicher, auf die über den Namen des Array verwiesen wird (siehe Handbuch *4D Programmiersprache*, Abschnitt **Arrays und Formularobjekte**). Ein PopUp-Menü bzw. DropDown-Liste zeigt ein Array als Liste mit Werten, wenn Sie darauf klicken.

Sie initialisieren das Objekt durch Laden einer Werteliste in ein Array. Es gibt folgende Möglichkeiten:

- Sie geben in den Objekteigenschaften eine Liste mit Standardwerten ein. Klicken Sie dazu in der Eigenschaftensliste unter der Gruppe „Datenquelle“ auf die Schaltfläche **Bearbeiten**. Weitere Informationen finden Sie im Abschnitt **Liste mit Standardwerten**. Die Standardwerte werden automatisch in das Array geladen. Über den Namen der Variablen, die dem Objekt zugeordnet ist, können Sie auf das Array verweisen.
- Sie führen vor der Anzeige des Objekts einen Code aus, der den Array-Elementen Werte zuweist. Ein Beispiel:

```
ARRAY TEXT(aCities;6)
aCities{1}:="Philadelphia"
aCities{2}:="Pittsburg"
aCities{3}:="Grand Blanc"
aCities{4}:="Bad Axe"
aCities{5}:="Frostbite Falls"
aCities{6}:="Green Bay"
```

In diesem Fall muss der Name der Variablen für das Objekt im Formular *aCities* lauten. Dieser Code kann in die Formularmethode eingefügt und ausgeführt werden, wenn das Formularereignis **On Load** auftritt.

- Laden Sie vor Anzeige eines Objekts die Werte einer Liste mit dem 4D Befehl **LIST TO ARRAY** in das Array. Ein Beispiel:

```
LIST TO ARRAY("Cities";aCities)
```

Auch hier muss der Name der Variablen für das Objekt im Formular *aCities* lauten. Dieser Code kann dann anstelle des oben genannten ausgeführt werden.

Wollen Sie die Auswahl des Benutzers in einem Datenfeld speichern, schreiben Sie einen Code, um die Werte zuzuweisen und führen ihn nach Bestätigen des Datensatzes aus. Die komplette Anweisung könnte folgendermaßen aussehen:

```
Case of
:(Form event=On Load)
LIST TO ARRAY("Cities";aCities)
If(Record number([People])<0) `neuer Datensatz
aCities:=3 `zeigt einen Standardwert an
Else `Bestehender Datensatz zeigt gespeicherten Wert an
aCities:=Find in array(aCities;City)
End if
:(Form event=On Clicked) `Vom Benutzer geänderte Auswahl
City:=aCities{aCities} `Feld erhält neuen Wert
:(Form event=On Validate)
City:=aCities{aCities}
```



```
:(Form event=On Unload)
CLEAR VARIABLE(aCities)
End case
```

Unter der Gruppe „Ereignisse“ in der Eigenschaftensliste markieren Sie die Ereignisse, die in Ihrer Case-Anweisung berücksichtigt werden.

Arrays enthalten immer eine begrenzte Anzahl an Einträgen. Die Liste der Einträge ist dynamisch und kann durch eine Methode geändert werden. Einträge in einem Array lassen sich ändern und sortieren.

Weitere Informationen zum Erzeugen und Verwenden eines Arrays finden Sie im Kapitel **Arrays** im Handbuch *4D Programmiersprache*.

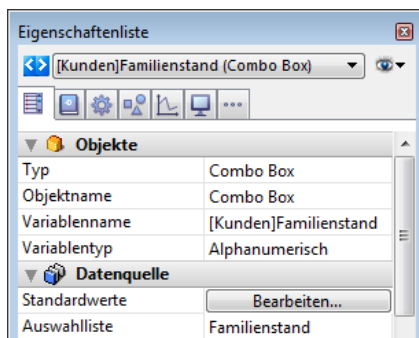
Eine Auswahlliste verwenden

Sie können auf ein Feld oder eine Variable direkt als Datenquelle für Objekte vom Typ PopUp/DropDown-Menü oder Combo Boxen (mit zugewiesener Auswahlliste) verweisen. Auf diese Weise lassen sich Felder bzw. Variablen mit Auswahlliste leichter verwalten.

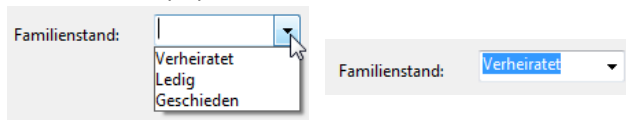
Hinweis: Bei hierarchischen Liste erscheint nur die erste Ebene und nur diese lässt sich auswählen.

Nehmen wir z.B. das Feld "Farbe", für das nur die Werte "Weiß", "Blau", "Grün" oder "Rot" zulässig sind. Dafür erstellen Sie jetzt eine Liste mit diesen Werten und ordnen sie einem Objekt PopUp-Menü zu, das auf das 4D Feld "Farbe" verweist. 4D verwaltet dann die Eingabe und zeigt den aktuellen Wert automatisch im Formular an.

Um einem Feld oder einer Variable ein PopUp-/DropDown-Menü oder eine Combo Box zuzuweisen, geben Sie den Feld- oder Variablennamen direkt in den Bereich Variablenname des Objekts ein:



Beim Ausführen des Formulars verwaltet 4D automatisch das PopUp-Menü oder eine Combo Box während der Eingabe oder Anzeige. Wählt der Benutzer einen Wert, wird er im Feld gesichert; beim Anzeigen des Formulars erscheint dieser Feldwert im PopUp-Menü:



Hinweis: Sie können diese Vorgehensweise nicht mit der Verwendung eines Arrays kombinieren, um das Objekt zu initialisieren. Geben Sie im Bereich **Variablenname** einen Feldnamen ein, müssen Sie eine Auswahlliste verwenden.

Sichern als

Ist das PopUp-Menü bzw. die DropDown-Liste mit einer Auswahlliste und mit einem Feld verbunden, können Sie auch die Option **Sichern als Wert/Referenz** verwenden und so die Größe der gesicherten Daten optimieren.

Weitere Informationen dazu finden Sie im Abschnitt **Als Wert oder Referenz sichern**.

Eine Standardaktion verwenden

Sie können PopUp-Menüs/DropDown-Listen eine Standardaktion zuweisen (Gruppe "Aktion" der Eigenschaftensliste). Diese Objekte unterstützen nur Aktionen, die eine Unterliste mit Einträgen anzeigen (außer die Aktion "Gehe zu Seite"). Wählen Sie z.B. die Standardaktion "backgroundColor", zeigt das Objekt in Echtzeit eine automatische Liste mit Hintergrundfarben. Sie können diese automatische Liste überschreiben, indem Sie zusätzlich eine Auswahlliste zuweisen, in der jedem Eintrag eine eigene Standardaktion zugewiesen ist.

Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

Combo Boxen

Eine Combo Box ähnelt einer Dropdown-Liste, mit dem Unterschied, dass dieses Objekt über Tastatur eingegebenen Text zulässt und zwei spezifische Optionen hat. Sie initialisieren eine Combo Box genauso wie eine Dropdown-Liste (siehe **PopUp-Menüs, Dropdown-Listen**).

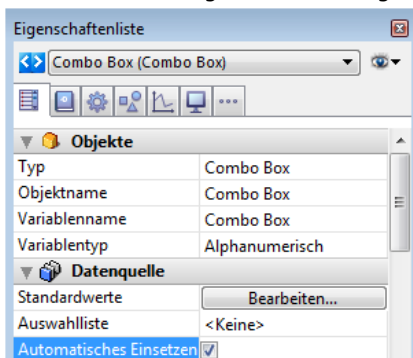
Gibt der Benutzer Text in der Combo Box ein, wird er im Element 0 (Null) des Arrays gespeichert. Eine Combo Box ist also ein eingebbarer Bereich, der sein Array oder eine Auswahlliste als Liste mit Standardwerten verwendet. Verwenden Sie das Ereignis `On Data Change`, um Einträge und eingebbare Bereiche zu verwalten. Weitere Informationen dazu finden Sie unter der Funktion **Form event** im Handbuch *4D Programmiersprache*.

Optionen für Combo Boxen

Für Objekte vom Typ Combo Box gibt es zwei Optionen für zugewiesene Auswahllisten: **Automatisches Einsetzen** und **Ausgenommen** (Liste mit ausgeschlossenen Werten).

Automatisches Einsetzen

Für Combo Boxen gibt es in der Eigenschaftensliste unter der Gruppe Datenquelle die Option **Automatisches Einsetzen**:



Hinweis: Diese Option ist auch für Spalten von Listboxen verfügbar, da die Zellen als Combo Boxen angezeigt werden, wenn eine Spalte mit einer Auswahlliste verknüpft ist

Ist diese Option markiert und gibt der Benutzer einen Wert ein, der in der zugewiesenen Auswahlliste nicht gefunden wird, wird er automatisch in der gespeicherten Liste hinzugefügt. Sie können Auswahllisten über den Befehl **OBJECT SET LIST BY NAME** oder **OBJECT SET LIST BY REFERENCE** hinzufügen.

Nehmen wir z.B. eine Auswahlliste mit den Einträgen "Blau, Grün, Gelb, Schwarz", die der Combo Box "Farbe" zugewiesen ist: Ist die Option **Automatisches Einsetzen** markiert und gibt der Benutzer "Rot" ein, wird der Wert "Rot" automatisch in der Liste im Speicher hinzugefügt:



Der eingegebene Wert darf natürlich nicht zur Liste der ausgeschlossenen Werte gehören, die dem Objekt evtl. zugewiesen ist (siehe nächster Abschnitt).

Hinweis: Wurde die Liste aus einer Liste erstellt, die im Designmodus definiert ist, bleibt das Original unverändert.

Ist die Option **Automatisches Einfügen** nicht markiert, wird der eingegebene Wert nur im Objekt und nicht in der Liste im Speicher gesichert.

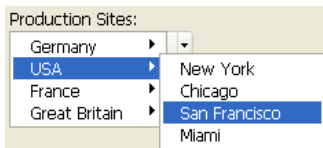
Ausgenommen

Die Option **Ausgenommen** für Combo Boxen liegt jetzt in der Gruppe "Wertebereich"; damit können Sie eine Liste ausgeschlossener Werte zuweisen. Gibt ein Benutzer einen Wert ein, der zu dieser Liste gehört, wird die Eingabe automatisch abgewiesen (siehe **Ausgenommen-Liste**).

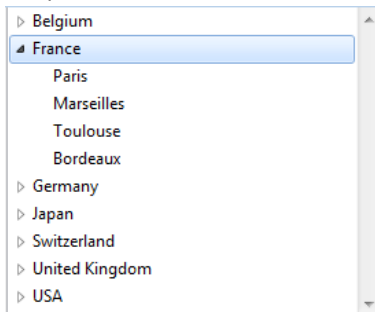
Hinweis: Einer Combo Box lässt sich keine Liste mit erforderlichen Werten zuweisen. Für diesen Fall müssen Sie ein Objekt vom Typ PopUp-Menü verwenden.

Hierarchische PopUp-Menüs und hierarchische Listen

Ein hierarchisches PopUp-Menü weist ein Untermenü auf, das jeder Zeile im Menü zugeordnet ist. Hier ein Beispiel für ein hierarchisches Menü:



Analog dazu enthält eine hierarchische Liste eine Unterliste, die jedem Eintrag in der Liste zugeordnet ist. Hier ein Beispiel für eine hierarchische Liste:



Hinweis: In Formularen sind hierarchische PopUp-Menüs auf zwei Ebenen begrenzt. Objekte vom Typ hierarchische Liste sind dagegen nicht begrenzt.

Sie können die hierarchische Liste erweitern oder schließen. Klicken Sie dazu auf den Pfeil vor dem jeweiligen Eintrag.

Hierarchische Listen können eingebbar sein. Ist ein Eintrag in einer hierarchischen Liste änderbar, kann der Benutzer ihn bearbeiten, wenn er unter Windows die **Alt-Taste**, auf Macintosh die **Wahl Taste** drückt oder lange auf einen Eintrag klickt. Wenn Sie eine hierarchische Liste mit einer Auswahlliste füllen, die im Listeneditor erzeugt wurde, bestimmen Sie mit dem Kontrollkästchen „Eingebbar“, ob ein Eintrag in einer hierarchischen Liste änderbar ist. Weitere Informationen finden Sie im Abschnitt **Listeneigenschaften definieren**.

Sie verwalten hierarchische PopUp-Menüs und hierarchische Listen mit den Programmierbefehlen für hierarchische Listen. Das Prinzip besteht darin, eine hierarchische Listenreferenz der Variablen zuzuordnen, die mit dem Objekt im Formulareditor verbunden ist. Weitere Informationen dazu finden Sie im Kapitel **Hierarchische Listen** des Handbuchs *4D Programmiersprache*.

Referenzen auf hierarchische Listen lassen sich über den Befehl **OBJECT SET LIST BY REFERENCE** oder **OBJECT SET LIST BY NAME** auch mit Auswahllisten (Quellen, erforderliche Werte und ausgeschlossene Werte) von Formularobjekten verbinden.

Standardaktionen verwenden

Sie können hierarchischen PopUp-Menüs eine Standardaktion zuweisen (Gruppe "Aktion" der Eigenschaftenliste). Dieses Objekt unterstützt nur Aktionen, die eine Unterliste mit Einträgen anzeigen. Wählen Sie z.B. die Standardaktion "backgroundColor", zeigt das Objekt in Echtzeit eine automatische Liste mit Hintergrundfarben. Sie können diese automatische Liste überschreiben, indem Sie dem Objekt eine zusätzliche Auswahlliste zuweisen, in der jedem Eintrag eine eigene Standardaktion zugewiesen ist.

Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

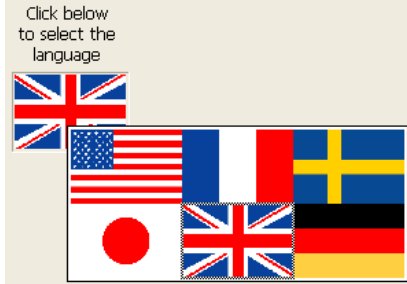
PopUp-Menü vom Typ Bild

Verwendung

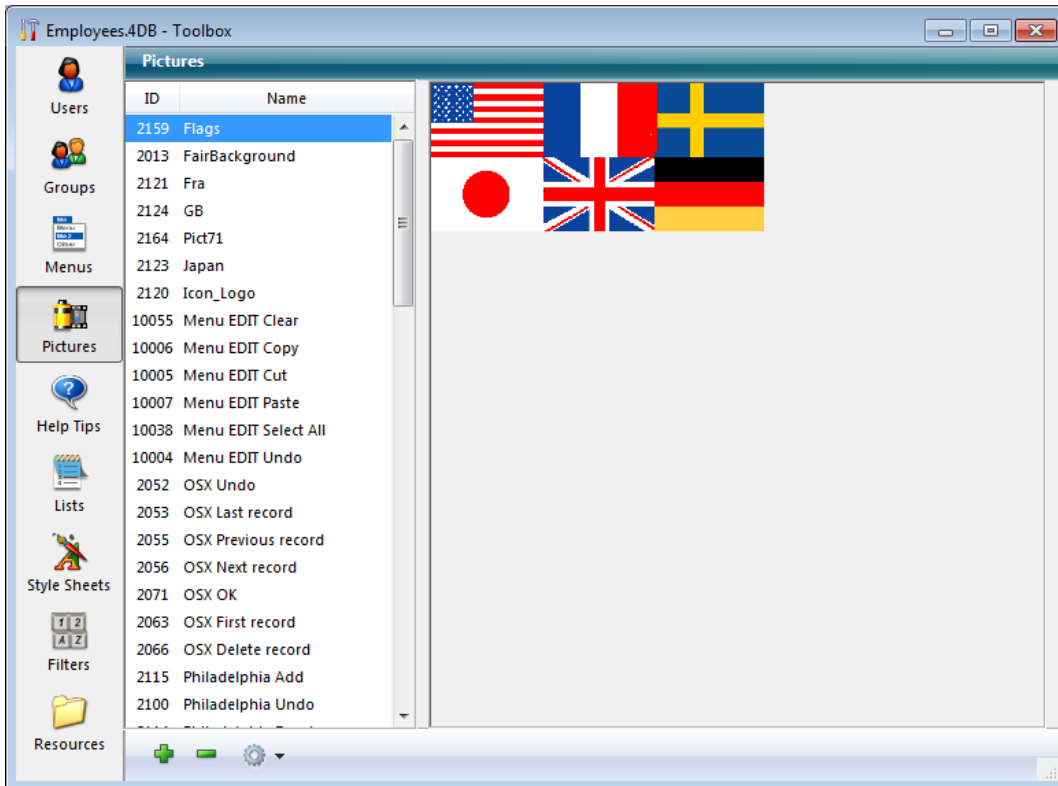
Ein PopUp-Menü vom Typ Bild ist ein PoUp-Menü, das ein zweidimensionales Array mit Bildern anzeigt. Das Konzept ist ähnlich wie beim **Schaltflächengitter**, mit dem Unterschied, dass das Bild nicht als Formularobjekt, sondern als PopUp-Menü verwendet wird.

Um ein PopUp-Menü Bild zu erzeugen, müssen Sie sich auf ein Bild beziehen. Das folgende Beispiel verwendet das Bild, das für Bildschaltflächen definiert wurde (siehe **Bildschaltflächen**). Hier können Sie die jeweilige Sprache für die Oberfläche aus einem PopUp-Menü Bild auswählen.

Jede Sprache ist mit einer Flagge symbolisiert:



Das PopUp-Menü Bild wird – wie für Bildschaltflächen – von einem Bild aus erstellt, das in Spalten und Zeilen unterteilt ist (oder in einem Schaltflächengitter). Sie können das Bild in der **Bildbibliothek**, in einer Bildvariablen oder in einer Bilddatei hinzufügen:



Sie können das PopUp-Menü Bild dann in das Formular einfügen.

Hinweis: Die **Bildbibliothek** bietet Optionen, um ein Bild in mehrere Einheiten aufzuteilen. Sie können Ihre Einstellungen sofort in der Vorschau betrachten. Wurde ein Bild als Tabelle definiert, können Sie direkt ein PopUp-Menü Bild erstellen. Ziehen Sie dazu den Bildnamen mit gedrückter **Umschalttaste** aus der Bildbibliothek in den Formulareditor.

Eigenschaften

Für PopUp-Menüs vom Typ Bild können Sie folgende Eigenschaften festlegen:

- Definieren Sie unter der Gruppe „Bild“ in der DropDown-Liste „Quelle“ die Herkunft des Bildes. Sie können wählen zwischen Variable, Bildbibliothek und Ressourcendatei oder Datei. Ist die Bildquelle gewählt, geben Sie im Bereich „Name/ID“ den Namen ein, wenn das Bild eine Variable ist oder aus der Bildbibliothek stammt, die Nummer, wenn es aus der Bildbibliothek oder einer Ressourcendatei stammt, oder den Pfadnamen des Bildes, wenn es aus einer Bilddatei stammt. In diesem Fall muss der Pfad zum Ordner **Resources** der Datenbank passen. Weitere Informationen dazu finden Sie im Abschnitt **Automatisches Verweisen auf Bilder**.
- Legen Sie unter der Gruppe „Abschneiden“ die Anzahl der Zeilen und Spalten der Bildertabelle fest.

- Die Parameter **Hor. Rand** und **Vert. Rand** erzeugen einen Abstand zwischen Menürändern und Bild. Geben Sie die Werte in Pixel ein.

Aktion Gehe zu Seite

Sie können einem Objekt vom Typ Bildmenü die Standardaktion **Gehe zu Seite** zuordnen. Ist diese Aktion aktiviert, zeigt 4D automatisch die Formularseite, die der Position des gewählten Elements im Bild-Array zugeordnet ist. Elemente werden von links oben nach rechts unten nummeriert. Wählt der Benutzer z.B. das 3. Element, zeigt 4D – sofern vorhanden – Seite 3 des aktuellen Formulars.

Wollen Sie selbst steuern, was beim Auswählen passiert, behalten Sie die Standardoption **Keine Aktion** bei. Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

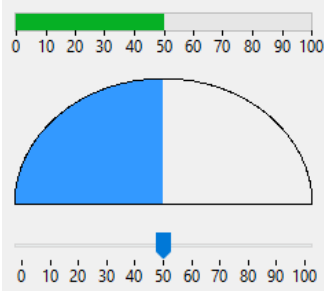
Programmierung

Sie können Bildmenüs über Methoden verwalten. Analog zu Schaltflächengittern nehmen Variablen, die einem PopUp-Menü vom Typ Bild zugeordnet sind, als Wert die Nummer des gewählten Elements an oder 0 (Null), wenn kein Eintrag ausgewählt ist. Die Einträge werden zeilenweise von links nach rechts durchnummeriert, beginnend mit der 1. Zeile.

Indikatoren

Ablaufbalken (auch "Thermometer" genannt), Lineale und Halbkreissskalen sind Objekte, die einen Wert grafisch darstellen. Die drei Objekte funktionieren auf die gleiche Weise; sie unterscheiden sich lediglich in der Darstellung. Wir nennen diese drei Objekte Indikatoren.

Sie können Indikatoren zum Anzeigen oder Setzen von Werten verwenden. Erhält zum Beispiel ein Thermometer über eine Methode einen Wert, zeigt er diesen Wert an. Zieht der Benutzer die Anzeigemarke bei gedrückter Maustaste, ändert sich der Wert. Dieser Wert kann von einem anderen Objekt, wie z.B. einem Datenfeld, einem einzelebaren oder nicht einzelebaren Objekt verwendet werden.



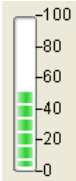
Ablaufbalken / Halbkreissskala / Lineal

Indikatorarten

4D bietet verschiedene Arten von Indikatoren, es gibt drei Hauptarten und dazugehörige Varianten. Weitere Informationen dazu finden Sie im unteren Abschnitt "Indikatorarten definieren".

Ablaufbalken

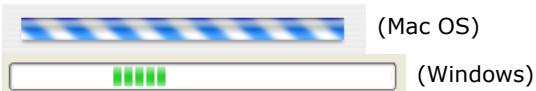
Haupttyp: Ablaufbalken (Thermometer)



Dieser Typ ist standardmäßig ausgewählt. Sie können Ablaufbalken horizontal oder vertikal anzeigen. Die Ausrichtung wird durch die Art, wie Sie den Balken im Formular zeichnen, festgelegt.

Animierter Balken

Haupttyp: Ablaufbalken (Thermometer)



Gerade Ablaufanzeige mit fortlaufender Animation. Dieser Typ wird in der Regel eingesetzt, um dem Benutzer anzuzeigen, dass gerade eine länger andauernde Operation ausgeführt wird. Bei dieser Variante sind die Optionen unter der Gruppe "Skalieren" ausgeblendet.

Beim Ausführen des Formulars ist das Objekt nicht animiert. Zur Animation müssen Sie in der zugeordneten Variablen einen Wert übergeben:

- 1 (oder jeder Wert außer 0) = Animation starten
- 0 = Animation stoppen

Hinweis: Animierte Thermometer funktionieren nur, wenn als Darstellung System oder Drucken gewählt ist.

Drehendes Rad

Haupttyp: Ablaufbalken (Thermometer)



Runde Ablaufanzeige mit fortlaufender Animation. Sie verwenden diesen Typ, um anzuzeigen, dass gerade eine Operation, wie z.B. eine Netzwerkverbindung oder eine Berechnung durchgeführt wird.

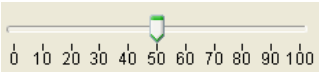
Bei dieser Variante sind die Optionen unter der Gruppe "Skalieren" ausgeblendet.

Beim Ausführen des Formulars ist das Objekt nicht animiert. Zur Animation müssen Sie in der zugeordneten Variablen einen Wert übergeben:

- 1 (oder jeder Wert außer 0) = Animation starten
- 0 = Animation stoppen

Lineal

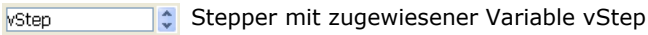
Haupttyp: Lineal



Dieser Typ ist das standardmäßige Objekt, um über einen Cursor, der die Skala entlang bewegt wird, Werte zu setzen oder zu erhalten. Sie können die zugeordnete Variable einem eingebbaren Bereich (Feld oder Variable) zuweisen, um den aktuellen Wert des Objekts zu speichern oder zu ändern.

Numerischer Schritt

Haupttyp: Lineal



Damit kann der Benutzer über die Pfeile in vordefinierten Schritten in numerischen Werten, Zeit- oder Datumsangaben scrollen.

Sie können die dem Objekt zugewiesene Variable einem eingebbaren Bereich (Feld oder Variable) zuordnen, um den aktuellen Wert des Objekts zu speichern bzw. zu verändern.

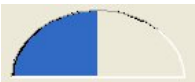
Ein Stepper lässt sich direkt einer Variablen vom Typ Zahl, Zeit oder Datum zuweisen.

- Bei Werten vom Typ Zeit stellen die Eigenschaften Minimum, Maximum und Schritt Sekunden dar. Um zum Beispiel einen Stepper von 8:00 bis 18:00 mit 10-Minuten Intervallen einzurichten, gehen Sie folgendermaßen vor:
 - Minimum = 28 800 ($8 \cdot 60 \cdot 60$)
 - Maximum = 64 800 ($18 \cdot 60 \cdot 60$)
 - Schritt = 600 ($10 \cdot 60$)
- Bei Werten vom Typ Datum stellt der in Schritt eingetragene Wert Tage dar, Minimum und Maximum werden ignoriert.

Hinweis: Damit der Stepper mit einer Zeit- bzw. Datumsvariablen arbeitet, muss sein Typ zwingend in der Eigenschaftenliste gesetzt UND explizit über den 4D Befehl **C_TIME** oder **C_DATE** deklariert werden.


Halbkreis skala

Haupttyp: Halbkreis skala

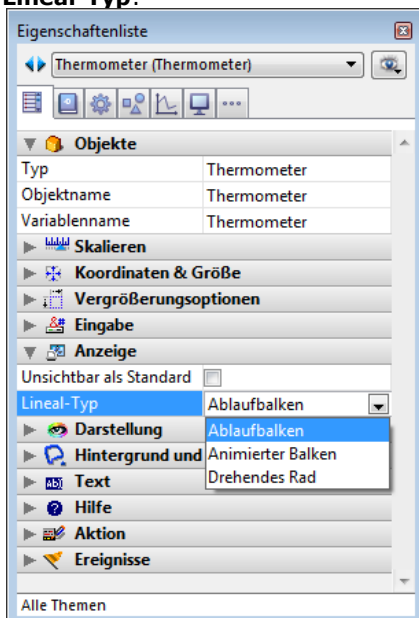


Dieser Typ zeigt die Daten in Form eines Halbkreises an.

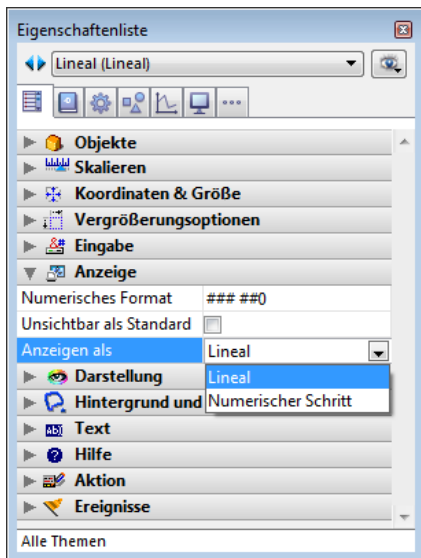
Indikator typen definieren

Halbkreis skalen, Ablaufbalken und Lineale sind in der Objektleiste des Formulareditors über die Schaltfläche  verfügbar. Darüber hinaus gibt es für "Ablaufbalken" (Thermometer) und "Lineal" verschiedene Varianten.

- Für Ablaufbalken setzen Sie die Varianten in der Eigenschaftenliste unter der Gruppe "Anzeige" über die Eigenschaft **Lineal-Typ**:



- Für Lineale setzen Sie die Varianten in der Eigenschaftenliste unter der Gruppe "Anzeige" über die Eigenschaft **Anzeigen als**:



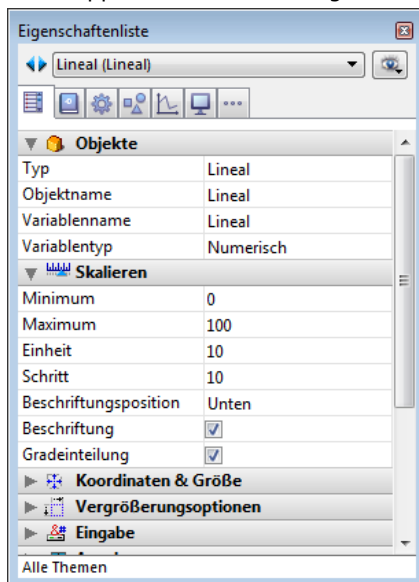
Hinweis: Diese Varianten lassen sich auch über den 4D Befehl **OBJECT SET FORMAT** setzen.

Eigenschaften

Zusätzlich zur Form der Darstellung können Sie für Indikatoren weitere spezifische Eigenschaften setzen: Minimum- und Maximumwert, die Einheit und den Schritt, einen Mindestschritt sowie Position und Anzeige der Beschriftung. Sie können auch das Anzeigeformat einer Beschriftung definieren. Weitere Informationen dazu finden Sie im Abschnitt **GET LIST ITEM PARAMETER**.

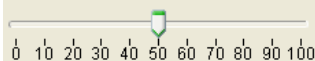
Gradeinteilung

Die Gruppe "Skalieren" der Eigenschaftenliste bietet verschiedene Eigenschaften:

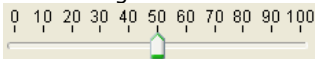


Diese Gruppe erscheint für Indikatoren vom Typ Ablaufbalken, Lineal, Numerischer Schritt und Halbkreissskala. Die Eigenschaften variieren je nach Indikatortyp. Es gibt folgende Möglichkeiten:

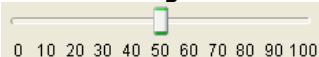
- **Minimum** und **Maximum**: Kleinster und größter Wert der Skala.
Bei numerischen Schritten stellen diese Optionen Sekunden dar, wenn das Objekt einem Wert vom Typ Zeit zugeordnet ist. Bei einem Wert vom Typ Datum werden sie ignoriert.
- **Einheit**: Maßeinheit für die Skala
- **Schritt**: Mindestabstand zwischen zwei Werten
Bei numerischen Schritten stellen diese Optionen Sekunden dar, wenn das Objekt einem Wert vom Typ Zeit zugeordnet ist, und Tage bei einem Wert vom Typ Datum.
- **Beschriftungsposition**: Position der Beschriftung, wenn diese angezeigt wird
 - **Unten**: Zeigt die Einheiten rechts oder unterhalb des Indikators



- **Oben**: Zeigt die Einheiten links oder oberhalb des Indikators



- **Beschriftung**: Blendet die Einheiten ein oder aus
- **Gradeinteilung**: Blendet die Skala zu den Einheiten ein- oder aus



Objektmethode ausführen

Ist ein Indikator ausgewählt, gibt es in der Eigenschaftenliste unter der Gruppe „Aktion“ die Eigenschaft **Objektmethode ausführen**.

Ist diese Option markiert, wird die Objektmethode mit dem Ereignis On Data Change ausgeführt, **während** der Benutzer den Wert des Indikators ändert. Standardmäßig wird die Methode erst **nach** der Änderung ausgeführt.

Indikatoren per Programmierung steuern

Die Variable, die dem Indikator zugewiesen ist, steuert die Anzeige. Mit Methoden setzen Sie Werte in den Indikator ein oder verwenden darin enthaltene Werte. Zum Beispiel könnten Sie über folgende Methode, die einem Datenfeld zugewiesen ist, ein Thermometer steuern:

```
vTherm:=[Angestellte]Gehalt
```

Diese Methode ist dem Datenfeld *Gehalt* zugewiesen. Sie weist den Wert des Datenfeldes *Gehalt* der Variablen *vTerm* zu. Umgekehrt können Sie über den Indikator den Wert in einem Datenfeld steuern. Der Benutzer zieht den Indikator bei gedrückter Maustaste, um den Wert festzulegen. Die Methode lautet dann:

```
[Angestellte]Gehalt:=vTherm
```

Diese Methode weist den Wert des Indikators dem Datenfeld *Gehalt* zu. Zieht der Benutzer den Indikator, ändert sich auch der Wert im Datenfeld *Gehalt*.

Verwendung

Eine Registerkarte erzeugt ein Objekt, das den Benutzer zwischen mehreren virtuellen Seiten wählen lässt. Jede Seite wird durch Klicken auf die entsprechende Registerkarte geöffnet. Ihr mehrseitiges Formular könnte folgendermaßen aussehen:

Um von Seite zu Seite zu wechseln, klickt der Benutzer einfach auf das gewünschte Register.

Über Registerkarten lassen sich Seiten in einem mehrseitigen Formular darstellen oder ein Objekt, das sich ändert, wenn der Benutzer auf eine Registerkarte klickt. Dient die Registerkarte zum seitenweisen Blättern, wird der 4D Befehl **FORM GOTO PAGE** oder die Standardaktion **Gehe zu Seite** verwendet, wenn der Benutzer auf eine Registerkarte klickt.

Eine Registerkarte kann auch zur Steuerung von Daten dienen, die in einem Unterformular oder gruppierten rollbaren Bereichen angezeigt werden. Zum Beispiel könnte ein Rolodex eingerichtet werden. Die Registerkarten zeigen die Buchstaben des Alphabets an. Die Aktion der Registerkarte lädt dann die Daten je nach dem angeklickten Buchstaben. Jede Registerkarte kann nur Bezeichnungen oder Bezeichnungen mit Icon anzeigen. Wenn Sie Icons einbinden, erscheinen sie vor den Bezeichnungen. Hier ein Beispiel für Registerkarten mit Icons:

Wenn Sie eine Registerkarte erzeugen, verwaltet 4D den Abstand und die Position der Registerkarten. Sie brauchen nur die Bezeichnungen in Form eines Array oder Icons und Bezeichnungen in Form einer hierarchischen Liste zur Verfügung stellen.

Wenn die Registerkarte breit genug ist, um Bezeichnung mit Icon darzustellen, zeigt sie beides an. Ansonsten erscheinen nur die Icons. Haben nicht alle Icons Platz, werden rechts neben der letzten sichtbaren Registerkarte Pfeile zum Blättern eingefügt. Damit kann der Benutzer in den Registerkarten nach rechts oder links blättern.

Auf Mac OS lassen sich die Registerkarten auch am linken oder rechten Rand und unten anordnen (siehe unteren Absatz "Anordnung ändern (nur Mac OS)").

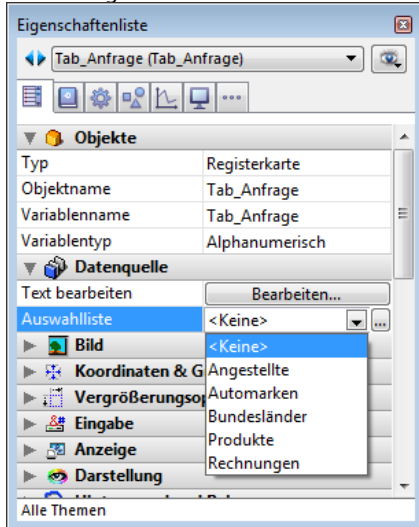
Einer Registerkarte Bezeichnungen hinzufügen

Es gibt verschiedene Möglichkeiten, einer Registerkarte Bezeichnungen hinzuzufügen:

- Sie verwenden in der Eigenschaftsliste unter der Gruppe „Datenquelle“ die Schaltfläche Bearbeiten. Auf dem Bildschirm erscheint das Dialogfenster „Standardwerte“. Weitere Informationen dazu finden Sie im Abschnitt **Liste**

mit Standardwerten. Standardwerte werden automatisch in ein Array geladen. Sie können auf dieses Array über eine Variable zugreifen, die der Registerkarte zugeordnet ist.

- Sie erzeugen mit dem Listeneditor eine Liste und weisen sie der Registerkarte als Auswahlliste zu:



Bei Bedarf können Sie im Listeneditor jedem Listeneintrag ein Icon zuordnen. Weitere Informationen dazu finden Sie im Abschnitt **Icon mit einem Eintrag verknüpfen**.

- Sie können ein Array vom Typ Text erzeugen, das die Namen jeder Formularseite enthält. Dieser Code muss ausgeführt werden, bevor das Formular dem Benutzer angezeigt wird. Sie können zum Beispiel den Code in die Objektmethode der Registerkarte einfügen und diese ausführen, wenn das Ereignis On Load auftritt.

```
ARRAY TEXT(arrPages;3)
arrPages{1}:="Adresse"
arrPages{2}:="Produkte"
arrPages{3}:="Rechnungen"
```

Hinweis: Sie können die Namen der Seiten in einer Liste speichern und den Befehl **Load list** zum Laden der Werte in das Array verwenden.

4D Befehl FORM GOTO PAGE

Sie verwenden den 4D Befehl **FORM GOTO PAGE** in der Objektmethode der Registerkarte:

```
FORM GOTO PAGE(arrPages)
```

Der Befehl wird ausgeführt, wenn ein Ereignis On Clicked auftritt. Sie sollten das Array wieder löschen, wenn ein Ereignis On Unload auftritt.

Hier ein Beispiel für die Objektmethode:

```
Case of
:(Form event=On Load)
LIST TO ARRAY("Tab Labels";arrPages)
:(Form event=On Clicked)
GOTO PAGE(arrPages)
:(Form event=On Unload)
CLEAR VARIABLE(arrPages)
End case
```

Aktion Gehe zu Seite

Sie können einem Objekt vom Typ Registerkarte die Standardaktion **Gehe zu Seite** zuordnen. Ist diese Aktion aktiviert, zeigt 4D automatisch die Formularseite, die der Nummer der gewählten Registerkarte zugeordnet ist. Wählt der Benutzer z.B. die 3. Registerkarte, zeigt 4D – sofern vorhanden – Seite 3 des aktuellen Formulars.

Wollen Sie selbst steuern, was beim Auswählen einer Registerkarte passiert, behalten Sie die Standardoption **Keine Aktion** bei.

Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

Anordnung ändern (nur Mac OS)

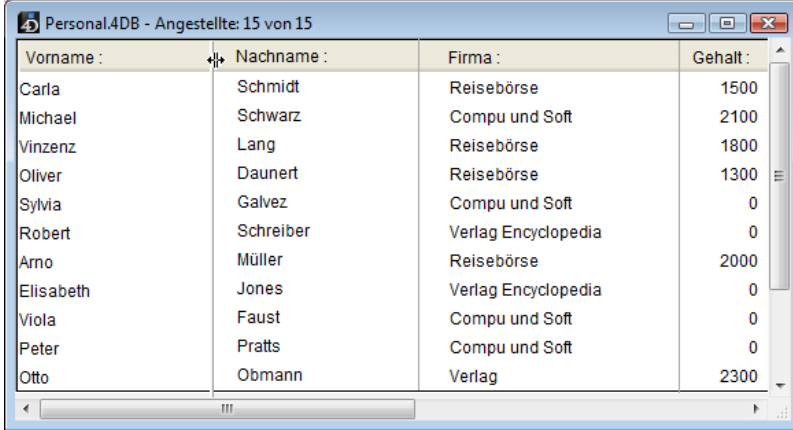
Sie können die Anordnung der Registerkarten in Ihren Formularen verändern. Diese Eigenschaft ist auf allen Plattformen verfügbar, lässt sich jedoch nur auf Mac OS anzeigen, wenn als Plattformoberfläche „System“ ausgewählt ist. Sie können festlegen, ob die Registerkarten **oben** (Standard) oder **unten** liegen.

Die gewünschte Anordnung legen Sie in der Eigenschaftsliste unter der Gruppe „Darstellung“ fest.

Hinweis: Wird ein Formular mit individuell angeordneten Registerkarten unter Windows oder auf Mac OS mit der Option “Druck” angezeigt, erscheinen sie automatisch mit der Standardeinstellung **oben**.

Verwendung

Ein Splitter unterteilt ein Formular in zwei Bereiche. Der Benutzer kann diese Bereiche durch Bewegen des Splitter vergrößern oder verkleinern. Ein Splitter kann horizontal oder vertikal sein. Der Splitter berücksichtigt die Anpassungsoptionen jedes Objekts, bei Bedarf kann er auch die Eigenschaft **Schieber** haben. So können Sie die Oberfläche Ihrer Datenbank komplett selbst gestalten. Er wird im allgemeinen in Ausgabeformularen (im Anwendungsmodus) verwendet, um Spalten anzupassen:



Vorname :	Nachname :	Firma :	Gehalt :
Carla	Schmidt	Reisebörse	1500
Michael	Schwarz	Compu und Soft	2100
Vinzenz	Lang	Reisebörse	1800
Oliver	Daunert	Reisebörse	1300
Sylvia	Galvez	Compu und Soft	0
Robert	Schreiber	Verlag Encyclopedia	0
Arno	Müller	Reisebörse	2000
Elisabeth	Jones	Verlag Encyclopedia	0
Viola	Faust	Compu und Soft	0
Peter	Pratts	Compu und Soft	0
Otto	Obmann	Verlag	2300

Splitter haben folgende Hauptmerkmale:

- Sie können in jeder Art von Formular beliebig viele Splitter setzen. Sie können in einem Formular gleichzeitig horizontale und vertikale Splitter verwenden.
- Ein Splitter kann ein Objekt abschneiden. Wird der Splitter entfernt, wird das Objekt wieder angepasst.
- Die Begrenzung ist so berechnet, dass die bewegten Objekte im Formular vollständig sichtbar bleiben bzw. nicht unter oder nahe einem anderen Splitter liegen. Bei einem Splitter mit Schiebereffekt werden die rechts davon oder darunterliegenden Objekte gleichzeitig mit verschoben.
- Passen Sie ein Formular über einen Splitter an, gelten die neuen Einstellungen nur, solange das Formular angezeigt wird. Sobald es geschlossen wird, werden wieder die ursprünglichen Maße hergestellt.

Der eingefügte Splitter erscheint als Linie. Sie können die Linie dünner machen oder ihre Farbe ändern. Sie können als Splitter auch eine unsichtbare Schaltfläche mit der Standardaktion **automatischer Splitter** verwenden (siehe **Standardaktionen**).

Interaktion mit den Eigenschaften benachbarter Objekte

In einem Formular interagieren Splitter mit den benachbarten Objekten je nach den definierten Anpassungsoptionen:

Anpassungsoptionen der Objekte	Objekte über horizontalen Splittern oder links von vertikalen Splittern (1)	Objekte unter horizontalen Splittern oder rechts von vertikalen Splittern	
		Splitter ohne Schieber	Splitter mit Schieber
Keine	Keine Änderung	Werden mit dem Splitter bis zum nächsten Stopp verschoben (die relative Position zum Splitter wird nicht verändert). Das ist bei Verschieben nach unten oder nach rechts entweder der Fensterrand oder ein anderer Splitter.	Werden mit dem Splitter unbegrenzt verschoben (die relative Position zum Splitter wird nicht verändert). Es wird kein Stopp angewendet (siehe nächster Abschnitt).
Anpassen	Behalten ihre ursprüngliche Position, werden aber gemäß der neuen Position der Splitter angepasst		
Bewegen	Werden mit dem Splitter verschoben		

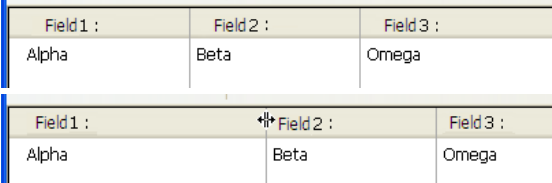
(1) Sie können den Splitter nicht über die rechte (horizontale) oder untere (vertikale) Seite eines Objekts an dieser Stelle ziehen.

Hinweis: Befindet sich ein Objekt vollständig innerhalb des Rechtecks, das den Splitter definiert, wird es zusammen mit dem Splitter bewegt.

Eigenschaft Schieber

Die Eigenschaftsliste enthält unter dem Thema "Vergrößerungsoptionen" für Splitter die Eigenschaft "Schieber". Ein Splitter mit dieser Eigenschaft verschiebt andere Objekte gleichzeitig mit dem Splitter. Das sind bei einem vertikalen Splitter die rechts davon liegenden Objekte, bei einem horizontal liegenden Splitter die darunterliegenden Objekte. Sie werden „geschoben“ oder „gezogen“.

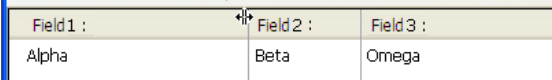
Hier das Ergebnis eines bewegten Splitters mit Schieber:



Field 1 :	Field 2 :	Field 3 :
Alpha	Beta	Omega

Field 1 :	Field 2 :	Field 3 :
Alpha	Beta	Omega

Zum Vergleich das Ergebnis eines bewegten Splitters ohne Schieber:



Field 1 :	Field 2 :	Field 3 :
Alpha	Beta	Omega

Diese Eigenschaft ist für neue Anwendungen standardmäßig aktiv.

Splitter per Programmierung verwalten

Sie können einem Splitter eine Objektmethode zuweisen. Sie wird während der gesamten Verschiebung über das Ereignis [On Clicked](#) aufgerufen.

Jedem Splitter ist eine Variable vom Typ Lange Ganzzahl zugeordnet. Diese Variable können Sie in Ihren Objekt- und/oder Formularmethoden einsetzen. Ihr Wert entspricht der aktuellen Position des Splitters, ausgedrückt in Pixel.

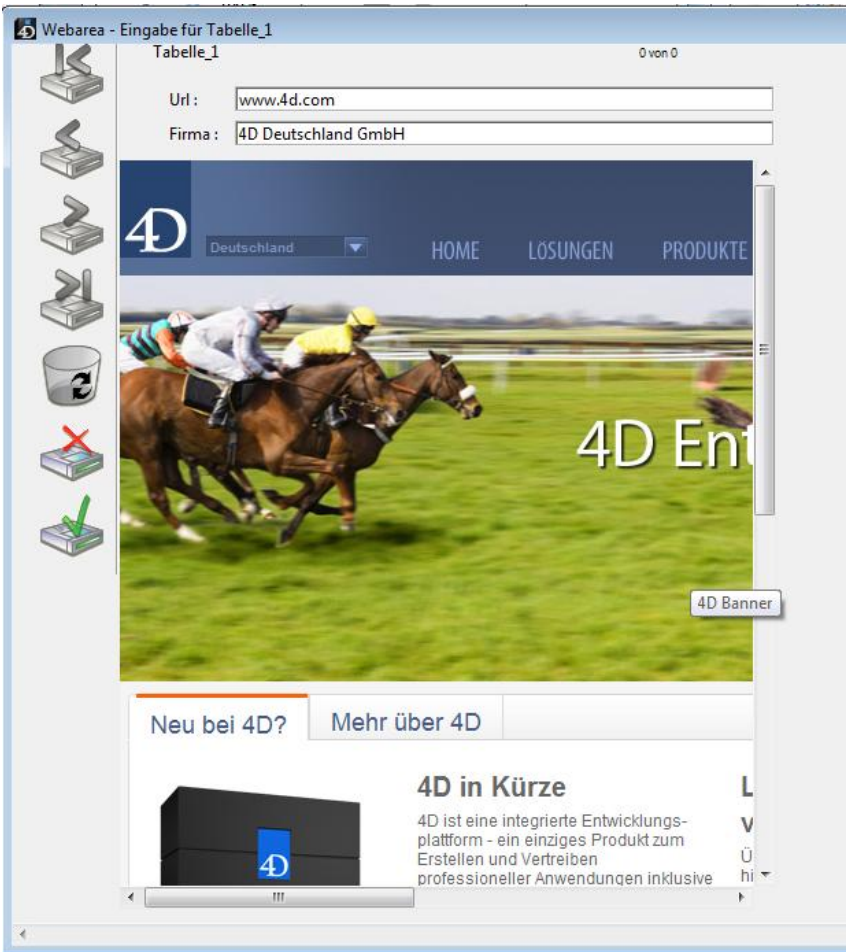
- Ist der Wert negativ, wurde der Splitter nach oben oder nach links bewegt
- Ist der Wert positiv, wurde der Splitter nach unten oder nach rechts bewegt
- Ist der Wert 0 (Null), wurde der Splitter in seine ursprüngliche Position zurückbewegt

Sie können den Splitter auch per Programmierung bewegen: Sie müssen dazu lediglich den Wert der zugewiesenen Variable setzen. Ist zum Beispiel ein vertikaler Splitter einer Variablen Split1 zugewiesen, und führen Sie das Statement: Split1:=-10 aus, wird der Splitter um 10 Pixel nach links bewegt — als ob der Benutzer dies manuell ausführt. Das Verschieben erfolgt am Ende der Ausführung der Formular- oder Objektmethode mit dieser Anweisung.

Mit der Aktion **automatischer Splitter** (siehe [Standardaktionen](#)) können Sie in Ihren Formularen eigene Splitter einrichten. Ordnen Sie diese Aktion einem Objekt vom Typ unsichtbare Schaltfläche zu, agiert er genau wie ein Splitter. Weitere Informationen zu unsichtbaren Schaltflächen finden Sie im Abschnitt [Schaltflächen](#).

Web Areas

Ein Web Bereich kann in Ihren Formularen verschiedene Arten von Web Inhalt (*) anzeigen: HTML Seiten mit statischem oder dynamischem Inhalt, Dateien, Bilder, Javascript, Flash, PDF.



(*) Die Verwendung von Web Plug-Ins und Java Applets in Web Areas wird nicht empfohlen, da dies zu Instabilität in der Arbeitsweise von 4D führen kann, insbesondere beim Verwalten von Ereignissen.

Die Rendering Engine von Web Area richtet sich nach der Ausführungsplattform des Programms und dem Status der Option **Integriertes Web Kit benutzen** (siehe nächsten Abschnitt "Spezifische Eigenschaften"):

In einem Formular lassen sich mehrere Web Bereiche anlegen. Beachten Sie jedoch, dass beim Einfügen von Web Areas bestimmte Einschränkungen gelten. Weitere Informationen dazu finden Sie im nachfolgenden Abschnitt **Nicht unterstützte Positionen**.

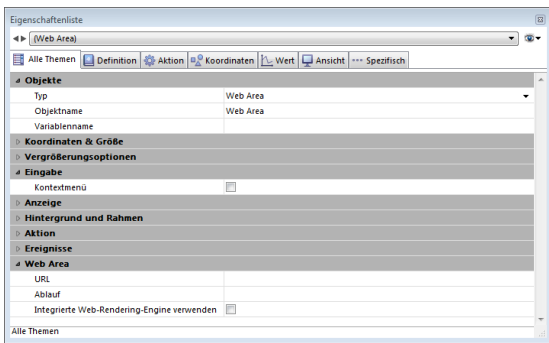
Es gibt verschiedene Standardaktionen, mehrere Programmiersprache-Befehle sowie generische und spezifische Formularereignisse, über die der Entwickler die Arbeitsweise von Web Bereichen steuern kann. Über spezielle Variablen lassen sich Informationen zwischen dem Bereich und der 4D Umgebung austauschen.

Dieser Abschnitt beschreibt die Verwendung und Konfiguration von Objekten des Typs Web Area im Formulareditor. Weitere Informationen zur programmierten Verwaltung dieser Objekte finden Sie im Kapitel **Web Area** des Handbuchs *4D Programmiersprache*.

Für Web Areas müssen verschiedene Regeln beachtet werden. Weitere Informationen dazu finden Sie im Abschnitt **Hinweise zum Verwenden von Web Bereichen** des Handbuchs *4D Programmiersprache*.

Spezifische Eigenschaften

Wählen Sie im Formulareditor ein Objekt vom Typ Web Area, zeigt die Eigenschaftenliste folgende Eigenschaften:



Kontextmenü

Markieren Sie unter der Gruppe "Eingabe" die Eigenschaft **Kontextmenü**, kann der Benutzer beim Ausführen des Formulars im Web Bereich auf ein standardmäßiges Kontextmenü zugreifen.

Der Inhalt des Menüs wird von der Rendering Engine der Plattform gesetzt: Unter Windows ActiveX Web (gleiches Rendering wie der Internet Explorer Browser), auf Mac OS Apple WebKit (gleiches Rendering wie der Safari Browser).

Hinweis: Mit dem 4D Befehl **WA SET PREFERENCE** lässt sich der Zugriff auf das Kontextmenü steuern.

Zugewiesene Variablen

Jedem Web Bereich werden automatisch zwei spezifische Variablen zugewiesen: Die Variable "URL" und die Variable "Ablauf". Damit lassen sich die vom Web Bereich angezeigte URL und der Prozentsatz beim Laden der Seite in den Web Bereich steuern. Diese Variablen haben standardmäßig die Namen *areaName_url* und *areaName_progress*. Bei Bedarf können Sie diese Namen über die Eigenschaftensliste ändern. Sie können diese Variablen in der Eigenschaftensliste setzen, wenn ein Web Bereich ausgewählt ist.

Weitere Informationen über die Funktionsweise dieser Variablen finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Web Areas per Programmierung verwalten**.

Integrierte Web Rendering-Engine verwenden

Mit dieser Option können Sie zur Wiedergabe von Web Areas wählen zwischen:

- **"Integrierte Web Rendering-Engine verwenden" ist nicht markiert (Standard):** In diesem Fall verwendet 4D die "beste" Engine des Betriebssystems. Unter Windows verwendet 4D automatisch die neueste Version des Browsers, die auf dem Rechner gefunden wird (IE11, MS Edge, etc.). Auf Mac OS verwendet 4D die aktuelle Version von WebKit (Safari).
Das bedeutet, dass Sie automatisch von den Neuerungen in Web-Rendering durch HTML5 oder JavaScript profitieren. Unter Umständen gibt es aber Unterschiede in der Wiedergabe durch Internet Explorer/Edge bzw. WebKit.
- **"Integrierte Web Rendering-Engine verwenden" ist markiert:** In diesem Fall verwendet 4D WebKit auf 32-Versionen und Blink (s.u.) auf 64-bit Versionen. Die Verwendung einer integrierten Web Engine bedeutet, dass die Wiedergabe der Web Area und deren Funktionsweise in Ihrer Anwendung identisch sind, unabhängig von der Plattform, auf der 4D läuft. Es kann lediglich geringfügige Pixelvariationen oder Unterschiede in der Netzwerk Implementation geben. Ist diese Option gewählt, nutzen Sie nicht länger die automatischen Updates der Web Engine, die das Betriebssystem durchführt; stattdessen werden neue Versionen der Engine durch 4D geliefert.

WebKit verwenden (integrierte Web Rendering Engine in 4D 32-bit Versionen):

WebKit ist eine open-source und multi-plattformfähige Web Rendering Engine von Apple. Sie wird unter anderem von Safari eingesetzt. Mit dieser Engine sind folgende Details zu beachten:

- Beim Anzeigen von Seiten über das HTTPS Protokoll wird die Gültigkeit des Zertifikats NICHT geprüft.
- Unter Windows lassen sich PDF Dateien nur in einem externen Fenster anzeigen.
- Bei Anzeige in einem neuen Prozess, insbesondere über die Funktion **New process**, muss der Stapelspeicher mindestens 256 KB (256*1024) groß sein, damit der Web Bereich korrekt angezeigt wird.

Blink verwenden (Integrierte Web Rendering Engine in 4D 64-bit Versionen)

In 64-bit Versionen von 4D ist Blink von Google (Integration in 4D noch in *Preview*) als Web Rendering Engine für Web Areas enthalten. Die Verwendung dieser Rendering Engine in 4D Web Bereichen ist für Benutzer transparent. Beachten Sie jedoch, dass einige Unterschiede im Verhalten oder Rendern auftreten können, genauso wie zwischen verschiedenen Browsern.

Features für Web Bereiche und alle Befehle im Kapitel **Web Area** werden unterstützt mit folgenden Einschränkungen für die Blink Engine:

- **WA SET PAGE CONTENT:** Mit diesem Befehl muss bereits mindestens eine Seite im Bereich geladen sein, entweder durch Aufrufen von **WA OPEN URL** oder Zuweisen der URL Variable, die dem Bereich zugeordnet ist.
- Java Applets ausführen: JavaScripte und Plug-ins sind immer aktiviert und lassen sich in Web Bereichen unter Blink nicht deaktivieren. Folgende Selektoren der Befehle **WA SET PREFERENCE** und **WA GET PREFERENCE** werden ignoriert:
 - [WA enable Java applets](#)
 - [WA enable JavaScript](#)
 - [WA enable plugins](#)
- Wird das Ziehen der URL vom Selector [WA enable URL drop](#) des Befehls **WA SET PREFERENCE** aktiviert, muss zuvor mindestens einmal der Befehl **WA OPEN URL** oder die dem Bereich zugeordnete URL Variable gesetzt werden.

Zugriff auf 4D Methoden

Hinweis: Diese Option ist nur verfügbar, wenn die Option **Integrierte Web-Rendering-Engine benutzen** markiert ist.

Ist die Option **Zugriff auf 4D Methoden** markiert, ist im Web Bereich ein JavaScript Objekt mit Namen \$4d vorhanden, über das Sie Aufrufe von 4D Projektmethoden verwalten können. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Web Areas per Programmierung verwalten**.

Standardaktionen

Es gibt vier Standardaktionen, um Web Bereiche automatisch zu verwalten: **Open Back URL**, **Open Next URL**, **Refresh Current URL** und **Stop Loading URL**. Sie können diese Aktionen Schaltflächen oder Menübefehlen zuordnen und so rasch einfache Web Oberflächen integrieren. Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

Ereignisse und Programmierbefehle

Web Areas lassen sich auch über Formularereignisse und spezifische Befehle steuern. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Web Area**.

Nicht unterstützte Positionen

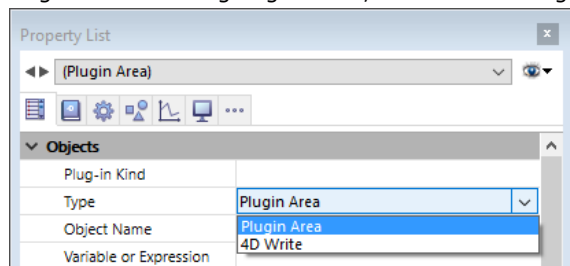
Da die Anzeige von Web Areas über eine externe Rendering Engine verwaltet wird, ist die Anzeige in 4D Formularen eingeschränkt. Beim Definieren von Web Areas müssen Sie folgende Einschränkungen berücksichtigen:

- Web Areas werden in Unterformularen mit Scroll-Funktion nicht unterstützt (Scrollen hat nicht den gewünschten Effekt).
- Web Areas dürfen nicht größer sein als das Unterformular selbst, in das sie integriert sind, d.h. sie müssen vollständig sichtbar sein.
- Die Überlappung einer Web Area mit anderen Formularobjekten (im Vordergrund oder dazwischenliegend) wird nicht unterstützt.

Plug-In Bereich

Overview

Ein **Plug-In Bereich** ist ein Bereich im Formular, der vollständig von einem 4D-Plug-In gesteuert wird. Beim Öffnen einer Datenbank erzeugt 4D eine interne Liste der Plug-Ins, die in Ihrer Datenbank installiert sind. Wenn Sie im Formular einen Plug-In-Bereich eingefügt haben, wird er in der Eigenschaftensliste unter der Gruppe **Objekte** aufgeführt.



Hinweise:

- Einige Plug-Ins, wie z.B. 4D Internet Commands sind nicht als Formularbereiche oder in externen Fenstern verwendbar. In diesen Fall erscheinen sie nicht in der Liste der Plug-Ins in der Eigenschaftensliste.
- In der 64-bit Version von 4D enthält die Liste standardmäßig keine Plug-Ins
- Ist der Objekttyp ein Plug-In-Bereich, kann die Schaltfläche **Erweitert** aktiv sein, wenn der Entwickler des Plug-In weitere Optionen vorgesehen hat. Klicken Sie auf diese Schaltfläche, um evtl. weitere Optionen zu definieren. Diese Optionen werden vom Autor des Plug-In verwaltet, d.h. er ist auch für weitere Informationen dazu zuständig.

Ist der angelegte Plug-In Bereich zu klein, zeigt 4D ihn als Schaltfläche mit dem Namen der Variablen an, die dem Bereich zugewiesen wurde. Der Benutzer kann bei der Ausführung darauf klicken, um ein spezifisches Fenster zur Anzeige des Plug-In zu öffnen.

Plug-Ins installieren

Um Plug-Ins in Ihrer 4D Umgebung zu installieren, müssen Sie zuerst 4D beenden. Plug-Ins werden nur beim Starten von 4D geladen. Weitere Informationen zur Installation und Konfiguration von Plug-Ins finden Sie im Abschnitt **Plug-Ins oder Komponenten installieren**.

Plug-Ins verwenden

Durch Einbinden von Plug-Ins in Formulare haben Sie unbegrenzte Möglichkeiten beim Erstellen eigener Anwendungen. Ein Plug-In kann eine einfache Aufgabe wie das Anzeigen einer Digitaluhr in einem Formular ausführen oder voll funktionale Textverarbeitungs-, Tabellenkalkulations- oder Diagrammfunktionen bieten.

Wenn Sie Ihre eigenen Plug-Ins einrichten möchten, können Sie ausführliche Informationen zum Schreiben und Implementieren von Plug-Ins erhalten. 4D bietet ein komplettes Kit zum Schreiben eigener Plug-Ins. Bitte wenden Sie sich an [4D](#).

Integrierte Plug-In Bereiche (nur 32-bit Versionen)

In 32-bit Versionen enthält 4D standardmäßig folgende Plug-Ins:

- **Report:** Über einen Berichtsbereich können Sie einen Schnellbericht in ein Formular einfügen. Er kann Daten als Tabelle darstellen, Berechnungen ausführen, etc. Der Bereich lässt sich durch eine eigene Menüleiste oder über Befehle der Programmiersprache steuern. Weitere Informationen dazu finden Sie im Kapitel **Schnellberichte**.
Hinweis: In 4D 64-bit Version wird dieses Plug-In durch eine Seite Unterformular ersetzt, die einen neuen Editor anzeigt (siehe **Schnellberichte (64-bit)**).
- **OLE tools** (nur Windows): Unter Windows können Sie über einen OLE Bereich in Ihrer Datenbank ein Fenster mit Dokumenten aus anderen Anwendungen öffnen. Weitere Informationen dazu finden Sie im Handbuch **OLE Tools**.
Hinweis: Ab 4D v16 ist das Plug-In OLE tools überholt.
















Listboxen

Listboxen sind komplexe aktive Objekte, um Daten in Form von Arrays anzuzeigen und einzugeben. Weitere Informationen dazu finden Sie im Kapitel **Listboxen**.

Unterformulare

Ein Unterformular ist ein Formular, das in ein anderes eingebunden ist.
Weitere Informationen dazu finden Sie im Kapitel **Unterformulare und Widgets**.

Eigenschaften für aktive Objekte

-  Objektnamen
-  Eingabekontrollen
-  Anzeigeformate
-  Als Wert oder Referenz sichern
-  Drag and Drop
-  Standardaktionen
-  Rechtschreibprüfung
-  Spiegeleffekt (Windows)
-  Text mit Stil (Rich Text)
-  Mehrzeilige Textbereiche
-  Rollbalken
-  Kontextmenü (Bilder)
-  Drucke variable Größe
-  Tastaturkürzel
-  Geometrie des Fensters beibehalten

Objektnamen

Objektname und Variablenname

Jedem aktiven Formularobjekt ist ein Objektname und ein Variablenname zugeordnet. Der Variablenname kann anders lauten als der Objektname. In einem Formular können Sie dieselbe Variable mehrmals verwenden. Der Objektname muss dagegen einmalig sein.

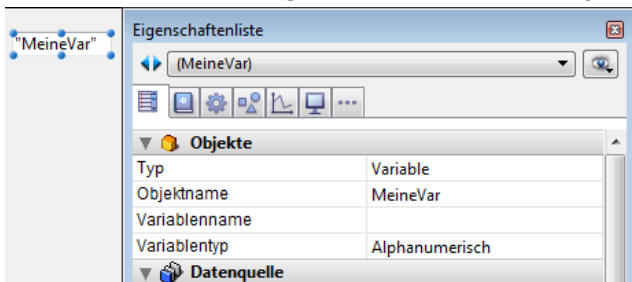
In der Programmiersprache können Sie auf ein aktives Formularobjekt entweder über seinen Variablennamen oder Objektnamen Bezug nehmen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Objekteigenschaften**.

Hinweis: Objektnamen sind auf die Größe 255 bytes begrenzt, Variablennamen auf die Größe 31 bytes.

Weitere Informationen zu den Namensregeln finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **GET REGISTERED CLIENTS** (die Regeln sind gleich) sowie im Abschnitt **Namenskonventionen**.

Dynamische Variablen

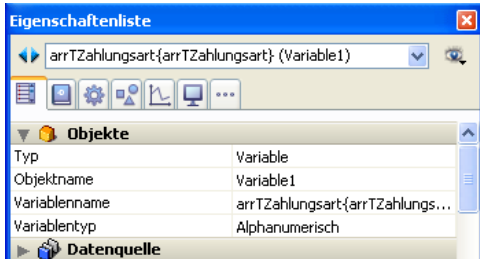
Sie können es auch 4D überlassen, Variablen, die Ihren Formularobjekten wie Schaltflächen, eingebare Variablen, Kontrollfelder zugeordnet sind, dynamisch und gemäß Ihren Bedürfnissen anzulegen. Lassen Sie dazu einfach das Feld "Variablenname" in der Eigenschaftenliste für das Objekt leer:



Weitere Information dazu finden Sie im Abschnitt **Variablen** im Handbuch *4D Programmiersprache*.

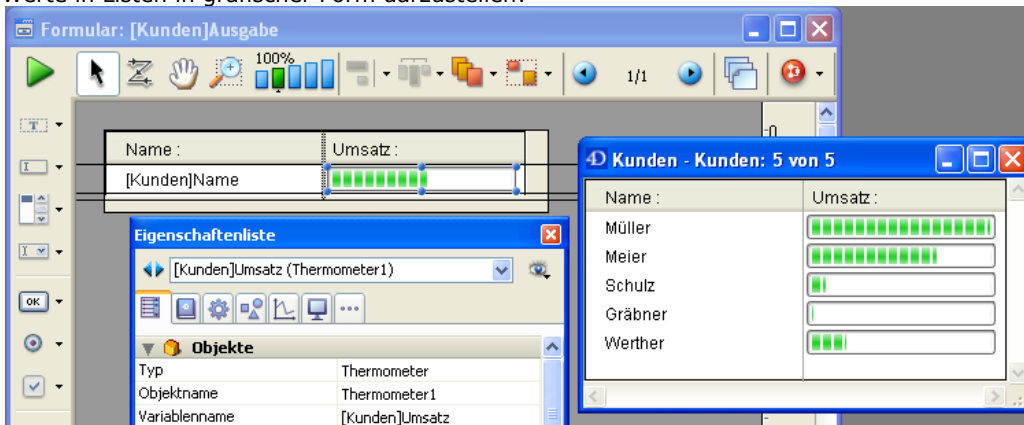
Ausdrücke als Variablennamen verwenden

Variablen, die Formularobjekten zugeordnet sind, können jede Art von Ausdruck enthalten, der einen Wert zurückgibt. Sie können den Ausdruck direkt im Bereich Variablenname in der Eigenschaftenliste für das Objekt eingeben:



Jeder gültige 4D Ausdruck ist erlaubt: Einfacher Ausdruck, Formel, 4D Funktion, Projektmethodenname oder Datenfeld mit der Standardsyntax [Tabelle]Feld. Der Ausdruck wird beim Ausführen des Formulars ausgewertet. In diesem Fall lässt sich der Wert der Variablen nicht per Programmierung steuern.

Das bietet vielfältige Möglichkeiten. Sie können z.B. einem Ablaufbalken ein Datenfeld vom Typ Zahl zuweisen, um die Werte in Listen in grafischer Form darzustellen:



Hinweis: Entspricht der eingegebene Wert sowohl einem Variablennamen als auch einem Methodennamen, wertet 4D diesen als Methodennamen.

Eingabekontrollen

Sie können in der Eigenschaftsliste verschiedene Optionen setzen, die die Dateneingabe für aktive Objekte steuert und vereinfacht. Es gibt folgende Möglichkeiten:

- Attribute zur Auswahl und Eingabe von Daten setzen
- Eingabefilter festlegen, der die erlaubten Zeichen steuert
- Platzhaltertext anzeigen
- Eine Auswahlliste anfügen
- Eine Liste erforderlicher oder ausgenommener Werte erstellen
- Maximal und minimal zulässige Werte oder Standardwerte festlegen
- Modus zur Objektauswahl setzen
- Rechtschreibprüfung ausführen
- Hilfmeldungen anzeigen

Eingebbar

Das Attribut **Eingebbar** gibt an, ob der Benutzer Werte in das Objekt eingeben kann. Es lässt sich in der Eigenschaftsliste unter der Gruppe "Eingabe" für **Get list item font** definieren.

Ein Datenfeld in einer verknüpften Tabelle ist unter Umständen nicht eingebbar, wenn Sie das Kontrollkästchen **Eingebbare verknüpfte Felder** im Formularassistenten deaktiviert haben. Sie können das verknüpfte Datenfeld durch Anklicken des Kontrollkästchens **Eingebbar** eingebbar machen.

Alle aktiven Objekte sind standardmäßig eingebbar. Wollen Sie ein Feld oder Objekt für dieses Formular nicht-eingebbar machen, deaktivieren Sie das Kontrollkästchen **Eingebbar**. Ein nicht-eingebbares Objekt zeigt Daten nur an. Sie können die Daten über Methoden steuern, die den Feld- oder Variablennamen verwenden. Die Formularereignisse [Clicked](#), [On Double Clicked](#), [On Drag Over](#), [On Drop](#), [On Getting Focus](#) and [On Losing Focus](#) können Sie auch mit nicht-eingebbares Objekten verwenden. Das erleichtert die Verwaltung eigener Kontextmenüs und lässt Sie Oberflächen einrichten, wo Sie nicht-eingebbare Variablen auswählen und per Drag-and-Drop bewegen können.

Hinweise:

- Der Inhalt der Eigenschaftsliste ist vom Kontext abhängig. Ist z.B. für ein Objekt oder ein Datenfeld die Eigenschaft Eingebbar nicht gewählt, verschwinden alle Eigenschaften zur Eingabekontrolle (Zwingend, Eingabefilter, Auswahlliste) aus der Liste.
- Sie können auch im Struktureditor für ein Datenfeld die Option **Nur anzeigen** setzen (siehe **UNREGISTER CLIENT**). In diesem Fall erscheint die Option **Eingebbar** für dieses Feld nicht in der Eigenschaftsliste (siehe unten).

Zwingend

Das Kontrollkästchen **Zwingend** macht ein Datenfeld oder eingebbares Objekt für das dazugehörige Formular zwingend. 4D akzeptiert einen Datensatz nicht, wenn dieses Feld bzw. Objekt keinen Wert enthält.

Standardmäßig ist kein Datenfeld oder eingebbares Objekt zwingend. Um das Datenfeld für alle Formulare zwingend zu machen, setzen Sie das Attribut **Zwingend** im Inspektorenfenster des Struktureditors. Wollen Sie ein Datenfeld oder ein eingebbares Objekt für ein spezielles Formular zwingend machen, markieren Sie in der Eigenschaftsliste die Option **Zwingend**.

Attribute Eingebbar und Zwingend und Feldeigenschaften

Die Attribute Eingebbar und Zwingend ähneln den Datenfeldeigenschaften, die Sie im Struktureditor festlegen (siehe **Feldeigenschaften**). Sollen diese Attribute für ein spezielles Formular abweichen, können Sie das in den Eigenschaften des jeweiligen Feldes festlegen. Sie setzen diese Attribute in der Eigenschaftsliste unter der Gruppe "Eingabe" für die **Get list item font**.

Diese Attribute überschreiben jedoch nicht die Datenfeldeigenschaften, die im Struktureditor festgelegt wurden.

- Wurde einem Datenfeld bereits im Struktureditor die Eigenschaft **Nur anzeigen** zugewiesen, können Sie es nicht mehr über das Attribut "Eingebbar" auf Formularebene eingebbar machen.
- Wurde einem Datenfeld bereits im Struktureditor die Eigenschaft **Zwingend** zugewiesen, können Sie diese Eigenschaft nicht durch Deaktivieren des Attributs "Zwingend" auf Formularebene aufheben.

Die Kontrollkästchen *Eingebbar* und *Zwingend* spiegeln nicht unbedingt die Einstellungen der Eigenschaften im Struktureditor wieder.

Tabfähig

Sie können für jedes aktive Objekt des Formulars das Attribut **Tabfähig** festlegen. Es wird dann in die Eingabereihenfolge mitaufgenommen und aktiviert, wenn der Benutzer die Tabulatortaste drückt.

Dieses Attribut ist standardmäßig für alle Felder und eingebbaren Variablen markiert. Deaktivieren Sie es, wird das Datenfeld in der Eingabereihenfolge nicht berücksichtigt. Es lässt sich jedoch weiterhin per Mausklick oder über eine

Methode auswählen.

Weitere Informationen dazu finden Sie im Abschnitt **Eingabereihenfolge der Daten ändern**.

Sie können das Attribut **Tabfähig** für ein nicht-eingebbares Objekt setzen. In diesem Fall lässt sich dieses Objekt über die Tabulatortaste auswählen, es lassen sich jedoch keine Werte eintragen.

Das Attribut **Tabfähig** ist nur verfügbar, wenn das Attribut **Fokussfähig** gewählt wurde, d.h. jedes tabfähige Objekt kann den Fokus haben. Dagegen können Objekte nur fokussfähig und nicht tabfähig sein. So lässt sich ein Objekt z.B. durch Anklicken und nicht über Tabulatortaste auswählen. In diesem Fall gehört das Objekt nicht zur Dateneingabereihenfolge.

Fokussfähig

Dieses Attribut lässt sich in der Eigenschaftensliste unter dem Thema "Eingabe" für aktive Objekte (eingebbar oder nicht-eingebbar) sowie für nicht-eingebbare Felder setzen.

Ist das Attribut **Fokussfähig** für ein Objekt gewählt, kann das Objekt den Fokus haben und z.B. über die Tastatur aktivierbar sein. Das ausgewählte Objekt wird mit einer grauen gestrichelten Linie umrahmt - außer die Option **Fokussrahmen ausblenden** wurde ebenfalls aktiviert (siehe nächster Abschnitt).

Current Member Kontrollkästchen zeigt Fokus bei Auswahl

Current Member Kontrollkästchen ist ausgewählt, kann aber keinen Fokus zeigen

Ist das Attribut **Fokussfähig** für ein nicht-eingebbares Objekt markiert, kann der Benutzer den Inhalt des Bereichs wählen, kopieren oder per Drag-and-Drop bewegen.

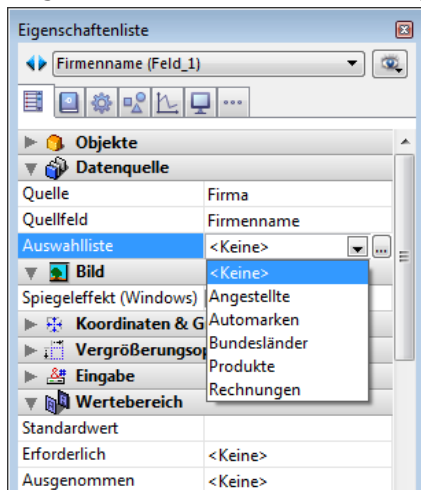
Fokussrahmen ausblenden

Bei der Ausführung (über Tabulator oder Mausklick) wird ein Datenfeld oder jedes eingebbare Objekt mit Fokus mit einem gestrichelten Rechteck umrahmt. Sie können dieses Rechteck über die Option **Fokussrahmen ausblenden** ausblenden. Das kann bei spezifischen Oberflächen manchmal hilfreich sein.

Auswahllisten

Sie können jedem eingebbaren Formularobjekt eine Auswahlliste zuweisen. Es gibt drei Arten: reguläre Auswahlliste, Liste erforderlicher Werte oder Liste ausgenommener Werte.

Reguläre Auswahllisten wählen Sie in der Eigenschaftensliste unter der Gruppe "Datenquelle", Listen erforderlicher bzw. ausgenommener Werte unter der Gruppe "Wertebereich".



Bevor Sie eine Auswahlliste zuweisen können, müssen Sie eine Liste im Listeneditor anlegen. Weitere Informationen dazu finden Sie unter **Auswahllisten**.

Auswahlliste unter Datenquelle

Eine Auswahlliste, die unter der Gruppe "Datenquelle" zugewiesen wurde, erscheint automatisch bei der Dateneingabe. Sie erscheint, wenn im Anwendungsmodus das Datenfeld bzw. eingebbare Objekt im Formular ausgewählt wird. Der Benutzer kann dann in der Liste eine Eingabe auswählen.

Sie können auch **PopUp-Menüs**, **Dropdown-Listen** oder **Combo Boxen** verwenden und diesen eine Auswahlliste zuweisen, um die Eingabe und Anzeige für ein Feld oder eine Variable zu verwalten (bei einer hierarchischen Liste lässt sich nur die erste Ebene verwenden). Dazu geben Sie den Feld- oder Variablennamen direkt in den Bereich "Variablenname" in der Eigenschaftensliste ein (siehe **PopUp-Menüs**, **Dropdown-Listen**) .

Er kann die gewählte Eingabe auch durch Eintippen überschreiben - außer bei einer Erforderlich-Liste.

Ausgenommen-Liste

Eine Auswahlliste mit der Eigenschaft **Ausgenommen** wird in der Eigenschaftensliste unter der Gruppe "Wertebereich" festgelegt. Sie verhindert, dass die in der Liste aufgeführten Einträge eingegeben werden. Sie können zum Beispiel einem Datenfeld in einem Eingabeformular, das Firmenmitarbeiter ausfüllen, eine Auswahlliste zuweisen, deren ausgenommene Werte nur ein Vorgesetzter definieren kann.

Felder und Auswahllisten

Sie können einem Datenfeld eine Auswahlliste auf Tabellen- oder Formularebene zuweisen. Auf Tabellenebene verwenden Sie das Inspektorfenster im Struktureditor (siehe **Feldeigenschaften**). Die Auswahlliste steht dann in allen Formularen

und Sucheditoren der Anwendung zur Verfügung. Auf Formularebene verwenden Sie die Eigenschaftensliste. Die Liste dient dann nur für dieses Formular als Auswahlliste. Beachten Sie, dass Sie in diesem Fall die Dateneingabe oder Anzeige im Feld direkt über **PopUp-Menüs**, **Dropdown-Listen** oder **Combo Boxen** verwalten können.

Erforderlich-Liste

Eine Auswahlliste mit der Eigenschaft **Erforderlich** wählen Sie in der Eigenschaftensliste unter der Gruppe "Wertebereich". Sie beschränkt die Eingabe auf die Einträge in der Liste. Das ist z.B. sinnvoll für Berufsbezeichnungen, um gültige Eingaben auf die Titel zu beschränken, die von der Firmenleitung abgesegnet sind.

Das Erzeugen einer Erforderlich-Liste zeigt die Liste nicht automatisch beim Auswählen des Datenfeldes an. Damit sie erscheint, müssen Sie die gleiche Liste über die Dropdown-Liste *Auswahlliste* unter der "Gruppe Datenquelle" zuweisen.

Eingabefilter

Ein Eingabefilter kontrolliert genau, was der Benutzer bei der Dateneingabe eingeben kann. Anders als die bisher beschriebenen Eingabekontrollen arbeiten Eingabefilter Zeichen für Zeichen. Besteht zum Beispiel eine Artikelnummer aus zwei Buchstaben gefolgt von drei Zahlen, können Sie einen Eingabefilter verwenden, um den Benutzer zur Verwendung dieses Musters zu zwingen. Sie können sogar einzelne Buchstaben und Zahlen vorschreiben.

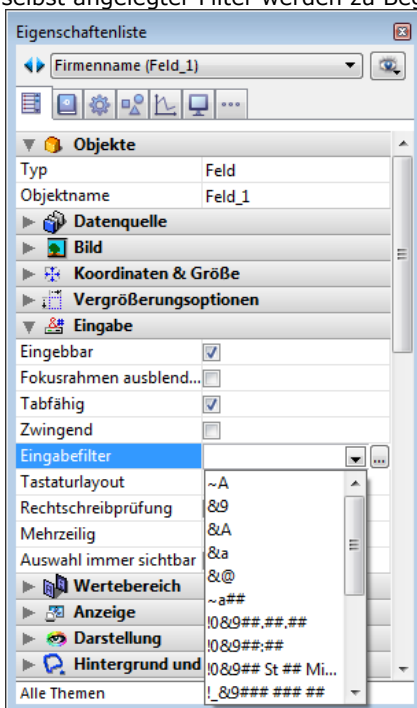
Ein Eingabefilter arbeitet nur bei der Dateneingabe. Er hat keine Auswirkungen auf das Datenformat, wenn der Benutzer das Datenfeld deaktiviert. In der Regel verwenden Sie Eingabefilter und **Anzeigeformate** zusammen. Der Filter beschränkt die Dateneingabe, das Format gewährleistet, dass der Wert nach der Dateneingabe korrekt angezeigt wird.

Bei der Dateneingabe prüft ein Eingabefilter jedes getippte Zeichen. Wenn der Benutzer versucht, ein ungültiges Zeichen einzugeben, beispielsweise eine Zahl statt eines Buchstaben, akzeptiert 4D das nicht. Das Leerzeichen bleibt unverändert, bis der Benutzer ein gültiges Zeichen eingibt.

Eingabefilter können auch zur Anzeige erforderlicher Formatierzeichen verwendet werden, so dass der Benutzer diese nicht eingeben muss. Zum Beispiel besteht eine amerikanische Telefonnummer aus einer dreistelligen Vorwahl gefolgt von einer siebenstelligen Nummer, die jeweils in zwei Gruppen von drei und vier Zeichen aufgeteilt wird. Sie können ein Format definieren, das die Vorwahl in Klammern setzt und einen Bindestrich nach der dritten Stelle der Telefonnummer. Dann muss der Benutzer bei der Eingabe weder Klammern noch Bindestrich setzen.

Eingabefilter auswählen

Sie wählen entweder einen Eingabefilter in der Dropdown-Liste oder geben einen eigenen Code in den Eingabebereich ein. Die Dropdown-Liste enthält Filter für Datum, Zeit und alphanumerische Datenfelder. In den meisten Fällen reichen die vorgegebenen Filter aus. Sie können aber auch eigene Filter erzeugen (siehe **Format- und Filtereditor**). Die Namen selbst angelegter Filter werden zu Beginn der Dropdown-Liste hinzugefügt, sie beginnen mit einem senkrechten Strich (|).



Sie können einen Eingabefilter verändern, nachdem sie ihn in der DropDown Liste gewählt haben. Wollen Sie z.B. einen Filter verwenden, der Groß- und Kleinbuchstaben zulässt, und zusätzlich das Joker-Zeichen (@), wählen Sie folgenden Filter:

```
&"a-z;0-9; ;,;.-"
```

und ändern ihn folgendermaßen:

```
&"a-z;0-9; ;,;.-;@"
```

Codes für Eingabefilter beginnen normalerweise mit einem kaufmännischen Und-Zeichen (&). Dieses Zeichen weist 4D an, die nachfolgenden Zeichen als Eingabefilter zu verwenden. Beginnt der Code mit einer Tilde (~), bedeutet dies das gleiche wie das "&"-Zeichen. Zusätzlich gilt noch, dass jedes Zeichen automatisch in Großbuchstaben umgewandelt wird. Auf das "&"-Zeichen folgt in der Regel ein "A," ein "a," oder eine "9," d.h. erlaube nur Großbuchstaben (A), erlaube Groß- und Kleinbuchstaben (a) oder erlaube nur Zahlen (9). Zum Beispiel erlaubt &9 nur Zahlen und &A nur Großbuchstaben. Das Nummernzeichen (#) gibt an, wieviel Stellen oder Zeichen der Code zulässt. Verwendet der Code kein Nummernzeichen, erlaubt der Filter beliebig viele Stellen bzw. Zeichen. Zum Beispiel lässt &9 alle eingegebenen Stellen zu, der Filter &9## lässt dagegen nur zwei Stellen zu.

Das Ausrufezeichen (!) wird manchmal verwendet, um das regulär auf dem Bildschirm angezeigte Zeichen zu verändern. Ohne ! zeigt 4D einen Unterstrich (_) für jede Stelle bzw. jedes Zeichen, das der Benutzer eingeben kann. Zum Beispiel

zeigt der Filter !?&9## Fragezeichen an beiden Stellen, die der Benutzer eintippen kann, erlaubt nur Zahlen und nur zwei Stellen. Weitere Informationen zum Erstellen von Eingabefiltern finden Sie unter [Codes für Filter und Formate](#).

Beschreibung der Standard Eingabefilter

Nachfolgende Tabelle erläutert die Wahlmöglichkeiten in der Dropdown-Liste der Eingabefilter:

Eingabefilter	Beschreibung
~A	Alle Buchstaben zulassen, aber in Großbuchstaben umwandeln
&9	Nur Zahlen zulassen
&A	Nur Großbuchstaben zulassen
&a	Nur Buchstaben zulassen (Klein- und Großbuchstaben)
&@	Nur alphanumerische Zeichen zulassen. Keine Sonderzeichen
~a##	Abkürzung für Landeskennzahl (z.B. DE). Nur zwei Buchstaben zulassen, aber in Großbuchstaben umwandeln.
!0&9##/##/##	Standardformat für Datumseingabe. Nullen an Eingabeleerstellen anzeigen. Alle Zahlen zulassen.
!0&9 Tag: ## Monat: ## Jahr: ##	Eigenes Eingabeformat für Datum. Nullen an Eingabeleerstellen anzeigen. Alle Zahlen zulassen, Zwei Eingaben nach jedem Wort.
!0&9##:##	Eingabeformat für Zeit. Beschränkt auf Stunden und Minuten. Nullen an Eingabeleerstellen anzeigen. Alle vierstelligen Zahlen, getrennt durch Doppelpunkt, zulassen.
!0&9## St ## Min ## Sek	Eingabeformat für Zeit. Nullen an Eingabeleerstellen anzeigen. Alle zweistelligen Zahlen vor jedem Wort zulassen.
!0&9St: ## Min: ## Sek: ##	Eingabeformat für Zeit. Nullen an Eingabeleerstellen anzeigen. Alle zweistelligen Zahlen nach jedem Wort zulassen.
!0&9###-##-###	Sozialversicherungsnummer. Nullen an Eingabeleerstellen anzeigen. Alle Zahlen zulassen
~"A-Z;0-9; ;,;.-"	Großbuchstaben und Zeichensetzung. Nur Großbuchstaben, Zahlen, Leerzeichen, Komma, Punkt und Strich zulassen.
&"a-z;0-9; ;,;.-"	Klein- und Großbuchstaben und Zeichensetzung. Zahlen, Leerzeichen, Komma, Punkt und Strich zulassen..
&"0-9;.-"	Zahlen. Nur Zahlen, Dezimalpunkt, Strich (Minuszeichen) zulassen.

Eingabefilter und Format zusammen verwenden

Wenn Sie zum Beispiel den Eingabefilter für die Sozialversicherungsnummer (&9###-##-###) verwenden, sollten Sie auch das passende Format für die Sozialversicherungsnummer verwenden (###-##-###). Ohne das Format werden nur Zahlen, aber nicht die Bindestriche im Datenfeld angezeigt.

Hier sehen Sie einige Vorschläge für Eingabefilter und die passenden Formate für häufige Datenfeldtypen:

Datenfeldtyp	Eingabefilter	Anzeigeformat
Land	~a##	(nicht erforderlich)
Postleitzahl	&9#####	(nicht erforderlich)
Sozialversicherungsnummer	&9###-##-###	###-##-###
Datum	!0&9##.##.## !0&9 Tag: ## Monat: ## Jahr: ##	(Jedes Datumsformat)
Zeit	!0&9##:## !0&9## St ## Min ## Sek !0&9St: ## Min: ## Sek: ##	(Jedes Zeitformat)

Sie können Anzeigeformate in Eingabefeldern, Ausgabefeldern und Schnellberichten verwenden. Weitere Informationen dazu finden Sie im Abschnitt [Anzeigeformate definieren](#).

Platzhaltertext

4D n den Feldern Ihrer Formulare Platzhaltertext anzeigen.

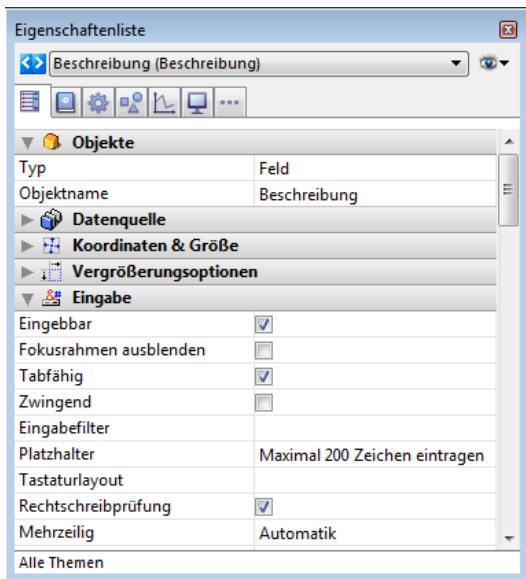
Dieser Text erscheint im Feld in Grauschrift und zeigt einen Hilfetipp, eine Erläuterung oder ein Beispiel zum Eingeben von Daten. Der Text verschwindet, sobald der Benutzer ein Zeichen in den Bereich eingibt:

Beschreibung: Maximal 200 Zeichen eintragen

Beschreibung: Lieferung in 24 h.

Wird der Inhalt des Feldes entfernt, erscheint der Platzhaltertext erneut.

Sie definieren den Platzhaltertext in der Eigenschaftenliste unter dem Thema "Eingabe":



Die Option Platzhalter ist für folgende Objekte verfügbar:

- Variablen
- Felder
- Combo Boxen

Ein Platzhalter lässt sich für folgende Datentypen anzeigen:

- String (Text oder alpha)
- Datum und Zeit, wenn die Eigenschaft **Leer wenn Null** markiert ist

Sie können auch eine xliiff Referenz in Form von ":xliiff:resname" als Platzhalter übergeben, z.B.:

```
:xliiff:PH_Lastname
```

Sie übergeben nur die Referenz im Feld mit der Eigenschaft "Platzhalter"; Sie können eine Referenz nicht mit statischem Text kombinieren.

Hinweis: Sie können den Platzhaltertext auch per Programmierung mit den Routinen **OBJECT SET PLACEHOLDER** und **OBJECT Get placeholder** setzen.

Maximum- und Minimumwerte festlegen

Sie können für Datenfelder oder eingebare Objekte vom Typ Zahl, Datum oder Zeit über Maximum- und Minimumwerte Einschränkungen definieren. Diese definieren Sie in der Eigenschaftenliste unter der Gruppe "Wertebereich".

Bei der Dateneingabe erscheint dann eine Warnung, wenn der Benutzer einen Wert eingibt, der unter dem Minimum bzw. über dem Maximum liegt. 4D führt den Benutzer zum Datenfeld zurück, so dass er einen gültigen Eintrag eingeben kann.

Geben Sie den Grenzwert ein, den Sie als Maximum- und Minimumwert bestimmen wollen. Verwenden Sie ein geeignetes Eingabeformat für den Typ des Datenfelds bzw. eingebaren Objekts. Verwenden Sie zum Beispiel für ein Feld vom Typ Datum das Eingabeformat für Datum.

Die Grenzwerte sind mit eingeschlossen, d.h. der Benutzer kann bis zum Maximum- und Minimumwert gehen. Nur Eingaben unter dem Minimum bzw. über dem Maximum sind nicht erlaubt. Haben Sie zum Beispiel als Maximum 15 festgesetzt, kann der Benutzer 15, aber nicht 16 eingeben.

Sie können auch Methoden zum Beschränken der eingebaren Werte verwenden. Mit einer Methode können Sie dem Benutzer genauere Informationen für die Eingabe geben oder Minimum- und Maximumwerte basierend auf anderen Werten in der Anwendung festlegen. Eine Methode kann zum Beispiel den Kreditrahmen eines Kunden vor Freigabe einer neuen Transaktion prüfen.

Sie können auch eine Auswahlliste mit zwingenden Werten verwenden, um Intervalle für zulässige Werte festzusetzen. Weitere Informationen dazu finden Sie im oberen Abschnitt **Erforderlich-Liste** und im Abschnitt **Listen erstellen und ändern**.

Standardwerte

Sie können einen Standardwert zur Eintragung in ein Datenfeld oder eingebaues Objekt zuweisen. Der Standardwert wird eingetragen, wenn ein neuer Datensatz zum ersten Mal geöffnet wird. Sie können den Wert ändern, außer das Datenfeld oder der Eingabebereich wurde als nicht eingebbar definiert.

Sie erzeugen einen Standardwert in der Eigenschaftenliste unter der Gruppe „Wertebereich“. Der Standardwert muss zum Datenfeldtyp passen. 4D bietet Kürzel für Standardwerte vom Typ Datum, Zeit und Aufzählung. Datum und Zeit werden vom Betriebssystem übernommen. 4D generiert automatisch beliebige fortlaufende Nummern. Es gibt folgende Kürzel für Standardwerte:

Kürzel Bedeutung

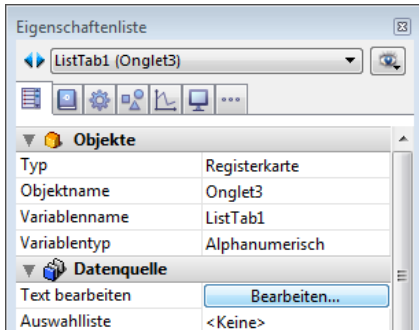
#D	Aktuelles Datum
#H	Aktuelle Zeit
#N	Fortlaufende Nummer

Sie können eine fortlaufende Nummer zum Erzeugen einer einmaligen Zahl für jeden Datensatz verwenden. Diese Nummer ist eine Ganzzahl, die für jeden neuen Datensatz erzeugt wird. Die Nummern beginnen bei (1) und werden schrittweise um (1) nach oben gezählt. Eine fortlaufende Nummer wird niemals wiederholt, selbst wenn der zugewiesene Datensatz aus der Tabelle gelöscht wird. Jede Tabelle hat ihren eigenen Satz fortlaufender Nummern. Weitere Informationen dazu finden Sie unter dem 4D Befehl **Sequence number** im Handbuch *4D Programmiersprache*.

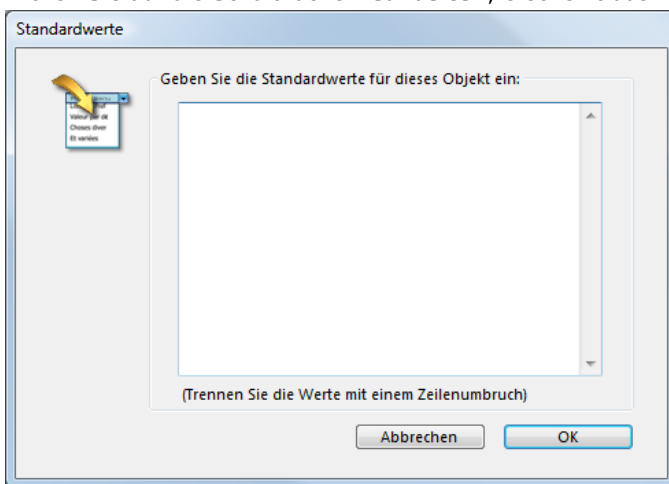
Liste mit Standardwerten

Wenn das Objekt eine Liste von Werten (wie Combo Box, rollbarer Bereich, PopUp-Menü, Registerkarte oder Dropdown-Liste) zeigt, können Sie eine Liste von Werten festlegen, die als Standardwerte verwendet werden. Die Liste wird vor dem Anzeigen im Formular im Objekt geladen.

Bei Objekten, die eine Liste von Standardwerten übernehmen, wird der Bereich Standardwert eine Schaltfläche:



Klicken Sie auf die Schaltfläche **Bearbeiten**, erscheint das Dialogfenster Standardwerte:



Geben Sie die Liste der Standardwerte ein. Jeder Wert sollte in einer eigenen Zeile stehen. Klicken Sie auf die Schaltfläche **OK**, um das Dialogfenster Standardwerte zu schließen und zur Eigenschaftensliste zurückzukehren

Wenn Sie im Dialogfenster Standardwerte eingeben, werden die Werte automatisch in ein Array geladen, dessen Name der Objektname ist. Mit der Programmiersprache können Sie das Objekt unter Bezug auf dieses Array verwalten.

Hinweise:

- Wenn das Objekt eine hierarchische Liste oder eine Registerkarte ist, können Sie zum Festlegen von Standardwerten auch eine Liste verwenden, die Sie mit dem Listeneditor erzeugt haben.
- Sie können Standardwerte auch mit einer Methode festlegen. Bei Objekten, die einen Wert übernehmen, können Sie den Standardwert zuweisen, wenn das Ereignis **On Load** in der Objekt- oder Formularmethode ausgeführt wird.
- Bei Objekten, die Listen akzeptieren, können Sie die Standardwerte über den Listeneditor eingeben, dann über die 4D Funktion **Load list** eine hierarchische Liste erstellen. Sie können die Liste und ihre Inhalte dann mit den Befehlen im Kapitel der *4D Programmiersprache* bearbeiten. Sie können die Einträge in eine hierarchische Liste laden, wenn das Formularereignis **On Load** auftritt oder alle Listen in die **Datenbankmethode On Startup** laden.

Tastaturlayout

Diese Option weist einem Feld oder eingebbaren Objekt ein spezifisches Tastaturlayout zu. Gibt es z.B. in einer internationalen Anwendung ein Formular mit einem Feld, dessen Inhalt in griechischen Zeichen eingegeben werden muss, können Sie diesem Feld das griechische Tastaturlayout zuweisen. Dann wechselt bei der Dateneingabe die Tastaturbelegung automatisch, wenn dieses Feld den Fokus hat.

Der Standardwert <None> gibt an, dass das Objekt das aktuelle Tastaturlayout verwendet.

Hinweis: Diese Option lässt sich mit den Befehlen **OBJECT SET KEYBOARD LAYOUT** und **OBJECT Get keyboard layout** dynamisch setzen.

Rechtschreibprüfung

4D bietet eine integrierte und anpassbare Rechtschreibhilfe an. Sie lässt sich auf Variablen und Felder vom Typ Alpha oder Text anwenden, sowie auf Dokumente in 4D Write Pro.

Die Eigenschaft **Rechtschreibprüfung** unter der Gruppe "Eingabe" aktiviert die Prüfung für jedes Feld/jede Variable. Die Rechtschreibprüfung wird dann automatisch beim Eingeben von Daten ausgeführt. Sie können auch für jedes zu prüfende Objekt den 4D Befehl **SPELL CHECKING** ausführen.

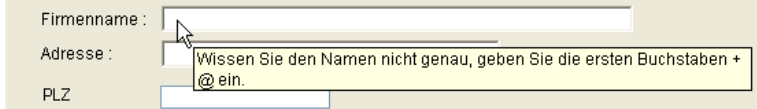
Weitere Informationen dazu finden Sie im Abschnitt [Rechtschreibprüfung](#).

Hilfemeldungen

Sie können Datenfeldern und aktiven Objekten in Ihrem Formular eine Hilfemeldung (Tipp) hinzufügen, um den Benutzern das Arbeiten mit Ihrer Anwendung zu erleichtern.

Hinweis: Über den Selektor [Tips enabled](#) des Befehls **SET DATABASE PARAMETER** lassen sich Hilfetipps für die gesamte Anwendung deaktivieren oder aktivieren.

Sie können zum Beispiel eine Hilfemeldung für ein Datenfeld erzeugen, die den Benutzer daran erinnert, beim Eingeben eines Datums als Trennzeichen einen Punkt (.) zwischen Tag, Monat und Jahr zu setzen. Der Tipp erscheint im Formular immer, wenn die Maus über das Datenfeld oder Objekt gezogen wird:



Sie können jedem aktiven Objekt im Formular einen Hilfetipp zuweisen. Wählen Sie dazu in der Eigenschaftensliste unter der Gruppe "Hilfe" die DropDown-Liste **Hilfetipp**. Sie können entweder:

- Einen vorhandenen Hilfetipp wählen. Die Meldung muss zuvor im **Hilfetipp Editor** von 4D angelegt werden.
- Die Hilfemeldung direkt im Bereich eingeben. Auf diese Weise können Sie die Vorteile der XLIFF Architektur nutzen. Sie können hier eine XLIFF Referenz eingeben, um die Meldung in der Anwendungssprache anzuzeigen. Weitere Informationen dazu finden Sie im [Anhang C: XLIFF Architektur](#). Sie können auch 4D Referenzen verwenden (siehe [Referenzen in statischem Text verwenden](#)).

Die gewählte oder neu angelegte Meldung erscheint im Formular als Tipp für das ausgewählte Objekt bzw. Datenfeld. Die verzögerte Anzeige und maximale Dauer der Anzeige von Hilfemeldungen lässt sich über die Selektoren [Tips delay](#) und [Tips duration](#) des Befehls **SET DATABASE PARAMETER** steuern.

Um die Zuordnung eines Tipps wieder zu löschen, wählen Sie in der DropDown-Liste den ersten Eintrag **<Keine>**.

Hinweis: Auf macOS wird die Anzeige von Hilfetipps in PopUp-Fenstern nicht unterstützt.

Sie können Hilfemeldungen für Formularobjekte auch auf zwei andere Arten zuweisen:

- auf Strukturebene der Datenbank (nur Felder). In diesem Fall wird der Hilfetipp für das Feld in jedem Formular, wo es vorkommt, angezeigt. Weitere Informationen dazu finden Sie im Abschnitt [Feldeigenschaften](#).
- Über den Befehl **OBJECT SET HELP TIP** für den aktuellen Prozess.

Sind demselben Objekt verschiedene Tipps an mehreren Stellen zugewiesen, gilt folgende Reihenfolge:

1. Strukturebene (niedrigste Priorität)
2. Ebene des Formulareditors
3. Befehl **OBJECT SET HELP TIP** (höchste Priorität)

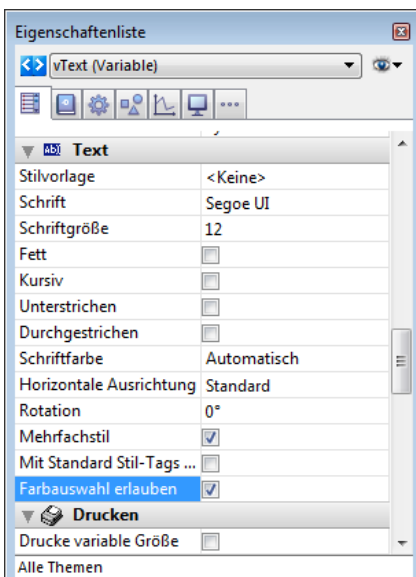
Auswahl immer sichtbar

Diese Eigenschaft ist für Datenfelder oder Variablen vom Typ Alpha oder Text in Formularen verfügbar. Sie erhält die Auswahl im Objekt sichtbar, auch wenn es den Fokus verloren hat. Das erleichtert das Erstellen von Oberflächen, die Stilveränderungen im Text zulassen. Weitere Informationen dazu finden Sie im Abschnitt [Text mit Stil \(Rich Text\)](#).

Farbauswahl und Schriftauswahl erlauben

Die Befehle **OPEN FONT PICKER** und **OPEN COLOR PICKER** zeigen die Farbpalette bzw. die Schriftenpalette des Systems an. Damit können Benutzer die Farbe oder Schrift eines Objekts mit Fokus im Formular direkt ändern.

Diese Funktion ist zum Steuern dieser Benutzeraktion auch in der Eigenschaftensliste unter der Gruppe "Text" verfügbar:

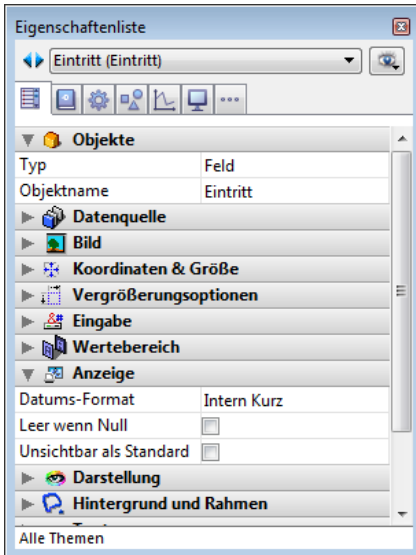


Sie ist für Formularobjekte vom Typ Feld, Variable oder Combo Box verfügbar. Standardmäßig ist sie inaktiv. Sie müssen sie explizit für jedes Objekt markieren, für das die Schrift bzw. Farbe über die jeweilige Palette des Systems änderbar sein soll.

Anzeigeformate

Die Anzeigeformate, die 4D bietet, geben Ihnen viele Möglichkeiten für Bildschirmanzeige und Drucken. Anzeigeformate lassen sich auf Datenfelder und eingebare oder nicht eingebare Bereiche (Variablen) anwenden. Das Format, das Sie zur Anzeige des Inhalts eines Datenfelds verwenden, beeinflusst nicht den tatsächlichen Wert, der von 4D gespeichert wird.

Das Format für ein Datenfeld kann in jedem Formular verschieden sein. Sie möchten zum Beispiel einen Wert ohne Eurozeichen in einem Eingabeformular und mit Eurozeichen in einem Ausgabeformular anzeigen. Sie legen Anzeigeformate in der Eigenschaftsliste unter der Gruppe „Anzeige“ fest



Combobox für Anzeige, Auswahl oder Eingabe eines Formats. Die Schaltfläche [...] bietet Zugriff auf das Fenster Filter und Formateinstellungen (siehe **Format- und Filtereditor**).

Je nach gewähltem Datenfeldtyp bzw. Variable erscheint in der Combobox ein anderes Format.

Hinweis: Den Typ einer Variablen definieren Sie in der Eigenschaftsliste unter der Gruppe „Objekte“.

Die für den Typ integrierten Formate erscheinen immer. Alle Anzeigeformate, die mit dem Format- und Filtereditor hinzugefügt wurden, erscheinen zu Beginn der Liste mit vorangestelltem Unterstrich (siehe **Format- und Filtereditor**).

Hinweis: Sie können im Formulareditor die Anzeigeformate für Objekte anstatt ihrer Bezeichnungen anzeigen. Wählen Sie dazu im Menü **Objekt > Format anzeigen**.

Datumsformate

Datumsformate bestimmen das Aussehen eines Datums bei Anzeige oder Druck. Bei der Dateneingabe geben Sie ein Datum im Format TT.MM.JJJJ ein, unabhängig, welches Format Sie gewählt haben.

Hinweis: Anders als bei alphanumerischen und numerischen Formaten lässt sich beim Anzeigeformat für Datum nur ein in 4D integriertes Format verwenden.

Die folgende Tabelle zeigt die verfügbaren Wahlmöglichkeiten:

Format	Beispiel
Systemabhängig kurz	28.07.10
Systemabhängig abgekürzt (1)	Mi, 28. Juli 2010
Systemabhängig ausgeschrieben	Mittwoch, 28. Juli 2010
Intern kurz spezial	28.07.2032 oder 28.07.1999 (2)
Intern ausgeschrieben	28. Juli 2010
Intern abgekürzt (1)	28. Jul 2010
Intern kurz	28.07.2010
ISO Datum Zeit(3)	28.07.2010T00:00:00

(1) Zur Vermeidung von Zweideutigkeit zeigen abgekürzte französische Datumsformate „jun“ für Juin „jul“ für Juillet (anstatt bisher „jul“ für beide Monate). Diese Besonderheit gilt nur für die französischen Versionen von 4D.

(2) Das Jahr erscheint standardmäßig zweistellig, wenn es zwischen 1930 und 2029 liegt, sonst vierstellig. Diese Standardeinstellung können Sie mit dem 4D Befehl **SET DEFAULT CENTURY** ändern.

(3) Das Format *ISO Datum Zeit* verwendet die XML Standarddarstellung für Datum und Zeit (ISO8601). Es dient vorrangig zum Importieren/Exportieren von Daten im XML Format und in Web Services.

Hinweis: Unabhängig vom Anzeigeformat gilt: Wird das Jahr zweistellig eingegeben, nimmt 4D für die Zeitspanne 00 bis 29 das 21. Jahrhundert an, für die Zeitspanne 30 bis 99 das 20. Jahrhundert.

Die Reihenfolge, d.h. Tag, Monat, Jahr ist immer vom System abhängig. Diese Standardeinstellung können Sie mit dem 4D Befehl **SET DEFAULT CENTURY** ändern.

Leer wenn Null

Ein Nulldatum wird standardmäßig dargestellt als "00.00.00." Die Option **Leer wenn Null** unter der Gruppe "Anzeige" in der Eigenschaftenliste zeigt einen leeren Bereich (""), wenn das Datum Null ist bzw. das Attribut SQL NULL enthält.

Zeitformate

Zeitformate bestimmen das Aussehen von Zeiten bei Anzeige oder Druck. Bei der Dateneingabe geben Sie Zeiten im 24-Stunden-Format hh:mm:ss oder im 12-Stunden-Format hh:mm AM/PM ein, unabhängig, welches Format Sie ausgewählt haben.

Hinweis: Anders als bei alphanumerischen und numerischen Formaten lässt sich ein Anzeigeformat für Zeit nur im PopUp-Menü Format auswählen.

Die folgende Tabelle zeigt die verschiedenen Wahlmöglichkeiten mit Beispielen:

Format	Kommentar	Beispiel für 04:30:25
HH:MM:SS		04:30:25
HH:MM		04:30
Stunde:Minute: Sekunde		4 Stunden 30 Minuten 25 Sekunden
Stunde:Minute		4 Stunden 30 Minuten
HH:MM AM/PM		4:30 a.m.
MM SS	Zeitspanne, als Dauer von 00:00:00 gerechnet	270:25
Min Sek	Zeitspanne, als Dauer von 00:00:00 gerechnet und ausgeschrieben	270 Minuten 25 Sekunden
ISO Datum Zeit	Entspricht dem XML Standard zur Darstellung von zeitspezifischen Daten. Dient hauptsächlich zum Importieren/Exportieren von Daten in XML Format	0000-00-00T04:30:25
System Format kurz	Standard Zeitformat im System definiert	04:30:25
System Format lang abgekürzt	<i>nur Mac OS:</i> Abgekürztes Zeitformat, vom System definiert. Unter Windows ist dieses Format identisch mit System Format kurz	4•30•25 AM
System Format lang	<i>nur Mac OS:</i> Langes Zeitformat im System definiert. Unter Windows ist dieses Format identisch mit System Format kurz	4:30:25 AM HNEC

Leer wenn Null

Eine Nullzeit wird z.B. dargestellt als "00:00:00". Die standardmäßige Anzeige richtet sich nach dem für das Objekt geltenden Format. Die Option **Leer wenn Null** unter der Gruppe "Anzeige" in der Eigenschaftenliste ermöglicht, einen Leerstring ("") anzuzeigen, wenn die Zeit Null ist oder das Attribut SQL NULL enthält.

Zahlenformate

Vorbemerkung: Datenfelder vom Typ Zahl enthalten die Typen Ganzzahl, Lange Ganzzahl, Ganzzahl 64 bit, Zahl und Fließkomma.

Zahlenformate bestimmen das Aussehen von Zahlen bei der Anzeige und beim Druck. Bei der Dateneingabe geben Sie nur Zahlen ein (einschließlich Dezimalkomma oder Minuszeichen bei Bedarf), unabhängig vom ausgewählten Format.

4D bietet in der Eigenschaftenliste unter der Gruppe "Anzeige" verschiedene Standard Zahlenformate. Sie können das Format im PopUp-Menü auswählen oder in der Eigenschaftenliste in der Combobox eingeben bzw. ändern. Sie können auch ein eigenes Zahlenformat wählen, das im **Format- und Filtereditor** der Toolbox definiert wurde. Solch ein Format lässt sich nicht über die Objekteigenschaften ändern. Der Editor öffnet sich, wenn Sie neben der Combobox auf die Schaltfläche [...] klicken.

Eigene Format- und Filternamen erscheinen am Anfang der Liste mit numerischen und alphanumerischen Formaten mit vorangestelltem senkrechtem Strich (|).

Platzhalter

In allen Zahlenformaten werden das Nummernzeichen (#), die Null (0), das Caret-Zeichen (^) und das Sternchen (*) als Platzhalter verwendet. Sie erzeugen Ihre eigenen Zahlenformate mit Platzhaltern für jede Stelle, die Sie anzeigen wollen.

Wenn Sie zum Beispiel dreistellige Zahlen anzeigen wollen, können Sie das Format ### verwenden. Wenn der Benutzer mehr Stellen eingibt, als das Format zulässt, zeigt 4D <<< im Datenfeld an, um zu signalisieren, dass mehr Stellen eingegeben wurden als die Anzahl der Stellen, die im Format festgelegt sind.

Wenn der Benutzer eine negative Zahl eingibt, wird das Zeichen ganz links als Minuszeichen dargestellt (es sei denn es wurde ein negatives Format festgelegt). Wenn das Format # # 0 lautet, wird minus 26 als -26 und minus 260 als <<< dargestellt, weil das Minuszeichen einen Platzhalter belegt und nur drei Platzhalter vorhanden sind.

Hinweis: 4D speichert und übernimmt die Zahl, die im Datenfeld eingegeben wurde, unabhängig vom Format. Es gehen keine Informationen verloren.

Jedes Zeichen für Platzhalter hat eine andere Auswirkung auf die Anzeige der führenden oder schließenden Nullen. Eine führende Null ist eine Null, mit der eine Zahl vor dem Dezimalkomma beginnt; eine schließende Null ist eine Null, mit der eine Zahl nach dem Dezimalkomma endet.

Angenommen, Sie verwenden das Format # # 0 zur Anzeige von drei Stellen. Gibt der Benutzer im Datenfeld nichts ein, zeigt das Datenfeld 0 an. Gibt er 26 ein, zeigt das Datenfeld 26 an.

Die folgende Tabelle erklärt die Auswirkungen der verschiedenen Platzhalter auf führende und schließende Nullen:

Platzhalter	Auswirkung auf führende oder schließende Nullen
#	Zeigt nichts an
0	Zeigt 0 an
^	Zeigt ein Leerzeichen an (1)
*	Zeigt ein Sternchen an

(1) Das Caret-Zeichen (^) erzeugt ein Leerzeichen, das dieselbe Breite wie eine Stelle in den meisten Schriften ausmacht.

Trennzeichen

Die numerischen Anzeigeformate (mit Ausnahme der wissenschaftlichen Notation) basieren automatisch auf den landesspezifischen Parametern des Systems, d.h. 4D ersetzt die Zeichen "." und "," in Zahlenformaten automatisch mit dem Trennzeichen für Tausend und Dezimal, wie es im Betriebssystem definiert wurde. Punkt und Komma werden demzufolge als Platzhalter gewertet, wie in den Formaten 0 oder #.

Hinweise zur Kompatibilität:

- In 4D Versionen vor v11 berücksichtigen numerische Anzeigeformate nicht die landesspezifischen Parameter des Betriebssystems. In konvertierten Datenbanken können Sie diese Funktionsweise über eine Kompatibilitätsoption steuern (siehe [Seite Kompatibilität](#)).
- Ab 4D v14 gibt auf einem Windows Rechner die Taste "Dezimaltrenner" des Nummernblocks für Felder vom Typ Zahl ein anderes Ergebnis als für Felder vom Typ Text:
 - Für Felder vom Typ Zahl wird bei Drücken dieser Taste des Zahlenblocks der auf Systemebene definierte Dezimaltrenner eingefügt.
 - Für andere Feldtypen wird das dieser Taste zugewiesene Zeichen eingefügt, in der Regel ein Punkt (.) oder ein Komma (,).
 Dieser Unterschied macht sich nur in Fällen bemerkbar, wo sich der Dezimaltrenner von dem dieser Taste zugewiesenen Zeichen unterscheidet. Das gilt z.B. für französische Systeme: Hier ist der Dezimaltrenner des Systems ein Komma, und der dem Nummernblock zugewiesene Dezimaltrenner ein Punkt.

Dezimalzeichen und andere Anzeigeeichen

Sie können im numerischen Anzeigeformat ein Dezimalkomma verwenden. Wenn Sie wollen, dass das Komma immer angezeigt wird, egal, ob der Benutzer es eingibt oder nicht, muss es zwischen zwei Nullen stehen.

Sie können im Format beliebige Zeichen verwenden. Bei alleiniger Verwendung oder Einfügen vor oder hinter anderen Platzhaltern erscheinen Zeichen immer. Wenn Sie zum Beispiel das Format €##0 verwenden, erscheint das €-Zeichen immer, weil es vor dem Platzhalter steht.

Wenn Zeichen zwischen Platzhaltern eingefügt werden, erscheinen sie nur, wenn Stellen auf beiden Seiten angezeigt werden. Legen Sie zum Beispiel das Format ###.##0 fest, erscheint der Punkt nur, wenn der Benutzer mindestens vier Stellen eingibt.

Im numerischen Anzeigeformat werden Leerzeichen als Zeichen gewertet.

Formate für Positiv, Negativ und Null

Ein Zahlenformat kann bis zu drei Teile enthalten. Dies ermöglicht Ihnen, Formate für Positiv-, Negativ- und Nullwerte festzulegen. Sie bestimmen die drei Teile durch Trennen mit Strichpunkten wie folgt:

Positiv;Negativ;Null

Sie müssen nicht alle drei Teile des Formats festlegen. Wenn Sie nur einen Teil verwenden, verwendet 4D es für alle Zahlen und setzt ein Minuszeichen vor negative Zahlen.

Wenn Sie zwei Teile verwenden, verwendet 4D den ersten Teil für positive Zahlen und Null und den zweiten Teil für negative Zahlen. Wenn Sie drei Teile verwenden, wird der erste Teil für positive Zahlen, der zweite für negative Zahlen und der dritte für Null verwendet.

Hinweis: Der dritte Teil (Null) wird nicht interpretiert und akzeptiert keine Zeichen zum Ersetzen. Geben Sie z.B. ###;###;# ein, erscheint der Wert Null als "#". Mit anderen Worten, was Sie gerade eingeben, wird für den Wert Null angezeigt.

Hier ein Beispiel für ein Zahlenformat, das €-Zeichen und Punkte verwendet, Negativwerte in Klammern setzt und keine Nullen anzeigt:

€###,##0.00;(€###,##0.00);

Beachten Sie, dass der zweite Strichpunkt am Ende 4D veranlasst, Nichts für die Anzeige von Null zu verwenden.

Das folgende Format ist ähnlich, mit dem Unterschied, dass der zweite Strichpunkt am Ende fehlt. Das veranlasst 4D, das positive Zahlenformat für Null zu verwenden:

\$###,##0.00;(\$###,##0.00)

In diesem Fall würde für Null €0,00 angezeigt.

Wissenschaftliche Notation

Wenn Sie Zahlen in wissenschaftlicher Notation darstellen wollen, müssen Sie das kaufmännische Und-Zeichen (&), gefolgt von einer Zahl verwenden, um die Stellenanzahl festzulegen, die Sie anzeigen wollen. Zum Beispiel würde das Format &3 die Zahl 759,62 als 7,60e+2 darstellen.

Die wissenschaftliche Notation rundet als einziges Format automatisch die angezeigte Zahl. Beachten Sie, dass im obigen Beispiel die Zahl auf 7,60e+2 aufgerundet wird, statt sie auf 7,59e+2 abzuschneiden.

Hexadezimalformat

Sie können mit den Formaten "&x" oder "&\$" eine Zahl hexadezimal anzeigen. Wenn der Benutzer "10" eingibt, wird dies als "A" angezeigt:

- &x zeigt die Zahl hexadezimal im Format "0xFFFF" an
- &\$ zeigt die Zahl hexadezimal im Format "\$FFFF" an

XML Notation

Das Format &xml macht die Zahl kompatibel mit den xml Standardregeln. Als Dezimaltrenner wird dann unabhängig von der Einstellung des Betriebssystems immer der Punkt "." verwendet.

Zahl als Zeit anzeigen

Sie können eine Zahl als Zeit (mit einem Zeitformat) anzeigen, indem Sie "&/" gefolgt von einer Stelle verwenden. Zeit wird durch Berechnung der Zahl der Sekunden seit Mitternacht, die der Wert darstellt, bestimmt. Die Stelle im Format entspricht der Reihenfolge, in der das Zeitformat im Dropdown-Menü Format erscheint. Zum Beispiel entspricht das Format &/5 dem fünften Zeitformat im PopUp-Menü, und zwar der AM/PM-Zeit. Ein Zahlendatenfeld mit diesem Format würde 25000 folgendermaßen darstellen als 6:56 AM.

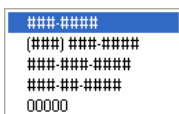
Beispiele

Die folgende Tabelle zeigt, wie verschiedene Formate die Anzeige der Zahlen beeinflussen. Die drei Spalten - Positiv, Negativ und Null - zeigen jeweils, wie 1.234,50, -1.234,50 und Null angezeigt werden:

Eingegebenes Format	Positiv	Negativ	Null
###	<<<	<<<	
####	1234	<<<<	
#####	1234	-1234	
####,##	1234,5	-1234,5	
###0,00	1234,50	-1234,50	0.00
####0	1234	-1234	0
+#####0;-#####0;0	+1234	-1234	0
#####0DB;#####0CR;0	1234DB	1234CR	0
#####0;(#####0)	1234	(1234)	0
###.##0	1.234	-1.234	0
##.##0,00	1.234,50	-1.234,50	0,00
^^^ ^^	1234	-1234	
^^^ ^^0	1234	-1234	0
^^.^^0	1.234	-1.234	0
^^.^^0,00	1.234,50	-1.,234,50	0,00
*****	***1234	** -1234	*****
*****0	***1234	** -1234	*****0
.0	**1.234	* -1.234	*****0
.0,00	*1.234,50	-1.234,50	*****0,00
\$.**0,00;-\$.**0,00	\$1.234,50	-\$1.234,50	\$\$\$**0,00
\$^^^0	\$ 1234	\$-1234	\$ 0
\$^^^0;-\$^^^0	\$1234	-\$1234	\$ 0
\$^^^0 ;(\$^^^0)	\$1234	(\$1234)	\$ 0
\$^.^^0,00 ;(\$^.^^0,00)	\$1.234,50	(\$1.234,50)	\$ 0,00
&2	1.2e+3	-1.2e+3	0.0e+0
&5	1.23450e+3	-1.23450e+3	0.00000
&xml	1234.5	-1234.5	0

Alpha Formate

Alphanumerische Formate bestimmen das Aussehen alphanumerischer Datenfelder und Variablen bei Anzeige und Druck. Hier die Liste der verfügbaren Formate:



Sie können ein Format in dieser Liste auswählen oder Formate für gängige alphanumerische Datenfelder, die Formate benötigen: Sozialversicherungsnummer und Postleitzahlen.

Sie können das Format auch in der Eigenschaftsliste in der Combobox eingeben bzw. ändern. Sie können auch ein eigenes Zahlenformat wählen, das im **Format- und Filtereditor** der Toolbox definiert wurde. Solch ein Format lässt sich nicht über die Objekteigenschaften ändern. Sie können auf den Editor zugreifen, wenn Sie neben der Combobox auf die Schaltfläche [...] klicken.

Eigene Format- und Filternamen erscheinen am Anfang der Liste mit numerischen und alphanumerischen Formaten mit vorangestelltem senkrechtem Strich (|).

Das Nummernzeichen (#) ist der Platzhalter für ein Format für alphanumerische Datenfelder. Sie können entsprechende Striche, Bindestriche, Leerzeichen und andere Satzzeichen einfügen, die Sie anzeigen wollen. Sie verwenden die von Ihnen gewünschten tatsächlichen Satzzeichen und das Nummernzeichen für jedes Zeichen, das Sie anzeigen wollen.

Betrachten Sie zum Beispiel eine Artikelnummer mit folgenden Format: RB-1762-1

Das alphanumerische Format würde folgendermaßen aussehen:

##-####-#

Wenn der Benutzer "RB17621" eingibt, zeigt das Datenfeld: RB-1762-1. Das Datenfeld enthält eigentlich "RB17621."

Gibt der Benutzer mehr Zeichen ein, als das Format zulässt, zeigt 4D die letzten Zeichen an. Lautet das Format z.B. (###) und der Benutzer gibt "GLÜCKLICH" ein, zeigt das Datenfeld (ICH) an.

Das Datenfeld enthält eigentlich "GLÜCKLICH". 4D übernimmt diese Eingabe und speichert sie unabhängig vom Format. Es gehen keine Informationen verloren.

Boolesche Formate

Boolesche Datenfelder können einen oder zwei Werte enthalten: WAHR oder FALSCH. Ein Boolesches Datenfeld kann als ein Paar Optionsfelder oder als Kontrollkästchen angezeigt werden. Legen Sie kein Anzeigeformat fest, lauten die Bezeichnungen standardmäßig Wahr und Falsch.

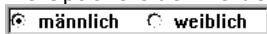
In den Feldeigenschaften definieren Sie Form und Bezeichnung eines Booleschen Datenfeldes. Wollen Sie nur die Schaltfläche oder Kontrollfelder ohne den Datenfeldnamen anzeigen, löschen Sie die Bezeichnungen im Formular.

Optionsfelder

Sie erstellen Form und Bezeichnungen für ein Boolesches Datenfeld in der Eigenschaftensliste unter der Gruppe „Anzeige“ über die DropDown-Liste „Anzeigen als“. Haben Sie Optionsfeld gewählt, stehen zwei Bereiche zur Eingabe der Bezeichnungen zur Verfügung:



Die Optionsfelder werden im Formulareditor nebeneinander wie folgt angezeigt:



Verwenden Sie Bezeichnungen mit verschiedenen Anfangsbuchstaben, können Sie das Optionsfeld durch Eingeben des Anfangsbuchstaben bei der Dateneingabe auswählen. Sie können zum Beispiel männlich durch Drücken von "m" und weiblich durch Drücken von "w" auswählen, wenn das Datenfeld ausgewählt ist.

Die folgenden Regeln gelten, wenn das Feld zum Speichern von Daten verwendet wird: Ist das erste Optionsfeld ausgewählt, ist das Datenfeld wahr; ist das zweite Optionsfeld ausgewählt, ist das Datenfeld falsch. Das Feld ist standardmäßig falsch.

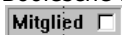
Ankreuzfelder

Wählen Sie in der Eigenschaftensliste unter der Gruppe „Darstellung“ in der DropDown-Liste die Option Ankreuzfeld, erscheint unter der Gruppe „Objekte“ ein Titel für das Kästchen. Standardmäßig ist die Bezeichnung des Datenfeldes eingetragen.

Die folgenden Regeln gelten, wenn das Feld zum Speichern von Daten verwendet wird: Ist das erste Ankreuzfeld ausgewählt, ist das Datenfeld wahr; ist das zweite Ankreuzfeld ausgewählt, ist das Datenfeld falsch. Das Feld ist standardmäßig falsch.



Sie können ein Boolesches Datenfeld als Kontrollkästchen ohne Bezeichnung formatieren. Geben Sie dazu im Titelbereich ein Leerzeichen ein und setzen den Rahmenstil auf transparent. In diesem Fall können Sie die Bezeichnung für das Kontrollkästchen als separates Objekt hinzufügen und an eine beliebige Stelle setzen. Sie können ein Rechteck ziehen und dynamische Referenzen in die Bezeichnung einfügen (siehe #title id="669"/]). Im folgenden Beispiel liegt das Boolesche Datenfeld auf dem Textobjekt.



Bildformate

Bildformate steuern, wie Bilder bei Anzeige oder Druck erscheinen. Formate für Bilddatenfelder bestimmen das Aussehen von Bildern bei Anzeige oder Druck. Bei der Dateneingabe gibt der Benutzer stets Bilder durch Einfügen aus der Zwischenablage oder per Drag&Drop ein, unabhängig vom Anzeigeformat. Bildformate erscheinen in der Eigenschaftensliste unter der Gruppe „Anzeige“.

Die Optionen für Abschneiden und Anpassen wirken sich nicht auf das Bild selbst aus. Der Inhalt eines Bilddatenfeldes bleibt gespeichert. Nur die Anzeige in einem Formular wird durch das Bildformat beeinflusst.

Größe angepasst

Mit diesem Format passt 4D das Bild an, so dass es in den Datenfeldbereich passt.

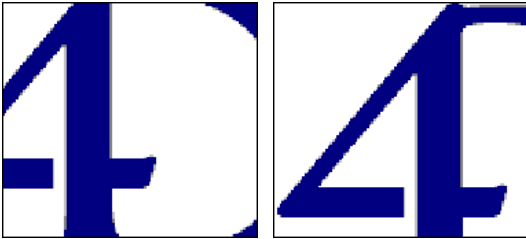


Abgeschnitten (zentriert und nicht-zentriert)

Mit dem Format **Abgeschnitten** (zentriert) zentriert 4D das Bild im Datenfeld und schneidet Teile ab, die nicht in den Datenfeldbereich passen. 4D schneidet von jeder Ecke zu und von oben nach unten.

Mit dem Format **Abgeschnitten** (nicht-zentriert) setzt 4D die linke obere Ecke des Bildes in die linke obere Ecke des Datenfeldes. 4D schneidet von rechts und unten zu.

Hinweis: Das Bildformat **Abgeschnitten** (nicht-zentriert) ermöglicht, dem Bild- oder Variablenbereich Rollbalken hinzuzufügen. Weitere Informationen dazu finden Sie unter **Rollbalken**.



Abgeschnitten (zentriert)

Abgeschnitten (nicht zentriert)

Proportional verkleinern und prop. verkleinern (zentriert)

Mit diesem Format verkleinert 4D das Bild an allen Seiten proportional, damit es in den für das Bild vorgesehenen Bereich passt. Die Option **Proportional verkleinert** (zentriert) führt dasselbe aus, zentriert aber das Bild im Bildbereich.

Ist das Bild kleiner als der im Formular definierte Bereich, wird das Bild nicht geändert. Ist das Bild größer als der im Formular definierte Bereich, wird das Bild proportional verkleinert. Da das Bild proportional verkleinert wird, kommt es zu keinen Verzerrungen.

Haben Sie das Format **Prop. verkleinert** (zentriert) gewählt, wird das Bild außerdem im Bereich zentriert:



Proportional verkleinert

Prop. verkleinert (zentriert)

Auf Hintergrund

Die Option **Auf Hintergrund** macht das Bild transparent. Alle Objekte, die hinter der Grafik eingefügt werden, wie z.B. Datenfelder oder Variablen, bleiben durch die Grafik hindurch sichtbar:

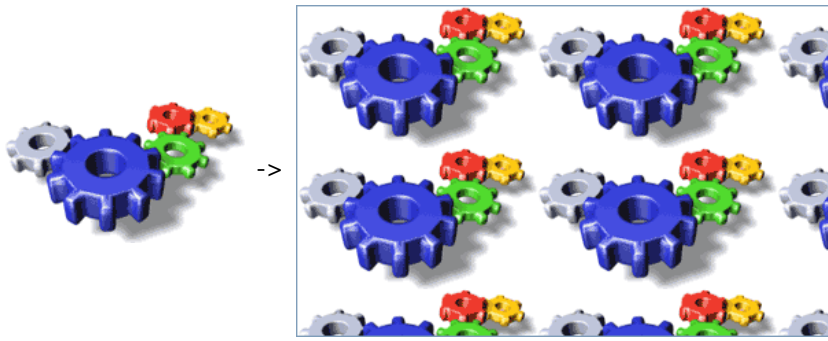


Hinweis: Bilder mit dem Format **Auf Hintergrund** werden als Bitmaps gedruckt.

Hat ein Bilddatenfeld dieses Format, kann der Benutzer das Bild im Innenbereich des Bilddatenfeldes durch Ziehen bei gedrückter Maustaste verschieben. 4D merkt sich die Position des Objekts auf dem Hintergrund.

Wiederholt

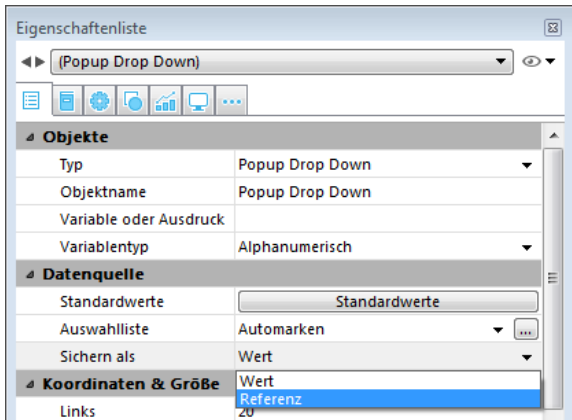
Ein Bild im Format **Wiederholt** wird nicht angepasst, wenn der Bereich, in dem es liegt, vergrößert wird. Es wird so oft wiederholt, bis der Bereich komplett ausgefüllt ist.



Ist der Bereich kleiner als das Originalbild, wird das eingesetzte Bild abgeschnitten, aber nicht zentriert.

Als Wert oder Referenz sichern

Unter der Gruppe "Datenquelle" gibt es die Option **Sichern als Wert/Referenz** für **Objekte vom Typ Feld und Variable** und **PopUp-Menüs, Dropdown-Listen**, sowie für **Spalten von Listboxen** (siehe **Eigenschaften für Spalten der Listbox**):



Diese Option erscheint unter folgenden Bedingungen:

- Dem Objekt ist eine Auswahlliste zugeordnet (siehe **Auswahllisten**)
- Für Variablen, Felder und Spalten von Listboxen ist für das Objekt auch eine Liste mit erforderlichen Werten definiert (in der Regel verwenden beide Optionen dieselbe Liste), so dass der Benutzer nur Werte aus der Liste eintragen kann.

Damit lässt sich bei einem Feld bzw. einer Variable mit zugewiesener Werteliste definieren, wie der Inhalt gesichert wird. Es gibt zwei Möglichkeiten:

- **Sichern als Wert** (Standardoption): Damit wird der Wert, den der Benutzer in der Liste ausgewählt hat, direkt gesichert. Wählt er z.B. den Wert "Blau", wird er im Feld gesichert.
- **Sichern als Referenz**: Damit wird die Referenz auf den Eintrag in der Auswahlliste im Objekt gesichert. Das ist die Nummer, die jedem Eintrag über den Parameter *EintragRef* der Befehle **APPEND TO LIST** oder **SET LIST ITEM** oder im Listeneditor zugeordnet wird (siehe **Dem Eintrag eine Referenznummer hinzufügen**).

Mit dieser Option können Sie die Speichernutzung optimieren, da das Speichern von Nummern weniger Platz braucht als das Speichern von Zeichenketten. Außerdem vereinfacht es die Übersetzung von Applikationen. Sie können mehrere Listen in verschiedenen Sprachen mit derselben Referenz pro Eintrag erstellen, und dann jeweils die Liste laden, die zur Sprache der Anwendung passt.

Bei **Sichern als Referenz** müssen Sie folgende Prinzipien beachten:

- Um die Referenz zu speichern, muss das Feld bzw. die Variable vom Typ Zahl sein - unabhängig, welcher Typ Wert in der Liste angezeigt wird.
- Listeneinträge müssen gültige und einmalige Referenzen haben.
- Aktivieren Sie diese Option für ein PopUp-Menü, muss es einem Feld zugewiesen werden (siehe **Eine Auswahlliste verwenden**).
- Diese Option ist kompatibel mit Auswahllisten, die in der Struktur definiert werden. In diesem Fall können Sie die Option einfach in jedem Formular auswählen, wo das Feld mit Auswahlliste vorkommt.

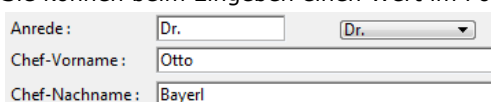
Beispiel

Sie wollen ein Feld zum Klassifizieren von Personen anlegen: Herr, Frau, aber auch Dr., Ing., Dipl-Ing., etc. Dafür erstellen Sie ein Feld vom Typ Lange Ganzzahl mit Namen "Anrede", definieren eine Auswahlliste (genannt "Anreden") mit den möglichen Titeln und weisen sie dem Feld zu.

Im Eingabeformular zeigen wir das Feld "Anrede" zweimal an, um die jeweilige Funktionsweise zu illustrieren: Einmal als PopUp-Menü, einmal als Eingabebereich. Beide Objekte werden derselben Auswahlliste zugeordnet und die Daten als Referenz gesichert:



Sie können beim Eingeben einen Wert im PopUp-Menü auswählen. Er erscheint in beiden Objekten:

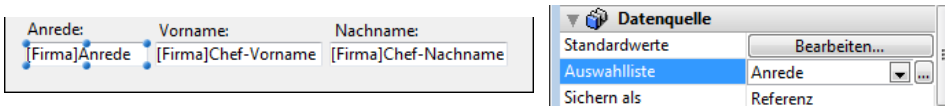


Hinweis: In diesem Formular erscheint das Fenster "Liste", wenn der Eingabebereich den Fokus hat. Damit dieses Fenster nicht mehr erscheint, können Sie ein PopUp-Menü zur Eingabe (und Anzeige) des Wertes verwenden (siehe

PopUp-Menüs, Dropdown-Listen).



Das Ausgabeformular richten Sie nach dem gleichen Prinzip ein: Sie wählen die Option **Referenz** zum Sichern des Feldes Anrede:



Beim Ausführen wird der Wert korrekt angezeigt:

Anrede :	Name :
Dr	Müller
Herr	Schmid
Frau	Rama
Herr	Gaume
Frau	Clapish
Dr	Attaway
M.	Meany

Drag and Drop

In der Eigenschaftsliste unter der Gruppe „Aktion“ gibt es verschiedene Optionen, um die Unterstützung von Drag and Drop zu konfigurieren. Die Anzahl und Aktion dieser Eigenschaften richtet sich nach dem jeweiligen Objekttyp.

Dragfähig und Dropfähig

Diese Eigenschaften steuern, ob der Benutzer das Objekt ziehen kann und ob das Objekt selbst Daten empfangen kann, die der Benutzer dorthin bewegt.

Wollen Sie die Eigenschaften Dragfähig und/oder Dropfähig für ein bestimmtes Objekt einschalten, müssen Sie die entsprechenden Kontrollkästchen markieren. In diesem Fall löst eine Drag- oder Drop-Operation auf das Objekt beim Ausführen des Formulars das entsprechende Formularereignis aus. Sie müssen dann die Drag&Drop-Aktion mit einer Methode steuern. Weitere Informationen dazu finden Sie im Abschnitt **Einführung in Drag and Drop** im Handbuch *4D Programmiersprache*.

Hinweis: Die Eigenschaft Dropfähig lässt sich nicht-eingebbaren Objekte zuordnen. Auf diese Weise kann der Entwickler die gewünschte Antwort auf ein Objekt, das auf ein nicht-eingebbares Feld bzw. eine Variable bewegt wird, programmieren.

Automatisches Drag and Drop

Aktive Textbereiche (Text und Variablen), Combo-Boxen und Bildbereiche (Felder und Variablen) bieten automatisches Drag and Drop, d.h. Sie können Text oder Bilder direkt in Formularbereichen bewegen oder kopieren. Sie können die Drag and Drop-Technik im gleichen 4D Bereich, zwischen zwei 4D Bereichen, oder zwischen 4D und einem anderen Programm, z.B. WordPad®, verwenden.

Sie können z.B. über automatisches Drag and Drop einen Wert zwischen zwei Feldern ohne Programmierung kopieren:



Um automatisches Drag and Drop zu aktivieren, wählen Sie unter der Gruppe "Aktion" der Eigenschaftsliste für das Objekt die entsprechenden Optionen.

Weitere Informationen finden Sie im Abschnitt **Einführung in Drag and Drop** im Handbuch *4D Programmiersprache*.

Standardaktionen

Standardaktionen zuweisen oder ausführen

Standardaktionen lassen sich auf verschiedene Arten verwenden:

- als **Aktionen für Schaltflächen und Formularobjekte**, wie Optionsfelder oder PopUp/DropDown-Listen. Sie lassen sich Formularobjekten über die Eigenschaftenliste des Formulareditors oder über den Befehl **OBJECT SET ACTION** zuweisen.
- als **Aktionen für Menübefehle**. Sie lassen sich Menübefehlen über den Menüeditor (siehe **Aktion für Menübefehle festlegen**) oder den Befehl **SET MENU ITEM PROPERTY** zuweisen.
- als **Aktionen für Listeneinträge** (wenn der Liste eine PopUp/DropDown-Liste oder ein hierarchisches PopUp-Menü zugeordnet ist). Sie lassen sich Listeneinträgen entweder im Listeneditor (siehe **Listen erstellen und ändern**) oder über den Befehl **SET LIST ITEM PARAMETER** zuweisen.
- als Parameter für die Befehle **INVOKE ACTION** und **Get action info**.

Formularobjekten oder Menübefehlen lassen sich eine Standardaktion und eine Methode zuweisen. In diesem Fall wird die Standardaktion immer nach der Methode ausgeführt.

Parameter für Standardaktionen

Einige Standardaktionen akzeptieren einen Parameter, der ihre Ausführung definiert. Die Syntax ist ähnlich zur URL Syntax:

```
standardActionName{?nameParameter=valueParameter}
```

dabei ist:

- *standardActionName* der Name der Standardaktion (String).
- *nameParameter* (optional) der Name des zu übergebenden Parameters (String)
- *valueParameter* (optional) der zu setzende Wert (String, Lange Ganzzahl...)

Um z.B. eine Aktion Gehe zu Seite (5) zu definieren, schreiben Sie:

```
gotoPage?value=5
```

Diese Syntax ist überall verfügbar, wo sich eine Standardaktion definieren lässt, z.B. in der Eigenschaftenliste, im Menüeditor, oder mit Befehlen der Programmiersprache. Hier ein Beispiel für die Eigenschaftenliste:

Aktion	
Standard Aktion	gotoPage?value=5
Methode	Methode
Dragfähig	<input type="checkbox"/>

Formularaktionen

Dieser Abschnitt beschreibt Standardaktionen in 4D Formularen zum Verwalten von Formularseiten und Datensätzen.

Aktionsname	Konstante (falls vorhanden)	Bedingungen zur Aktivierung	Verfügbar mit (*)	Beschreibung
"" (leerer String)	<u>ak none</u>	N/A	Schaltflächen, Menübefehlen	Führt keine Standardaktion durch. Verwenden Sie diesen Wert, wenn Sie eine Methode zum Steuern der Aktion schreiben müssen. Beispiel: Eine Schaltfläche, die in der Anwendungsumgebung ein eigenes Dialogfenster Suchen öffnet, hat keine Standardaktion, da Sie eine Methode zum Öffnen des eigenen Dialogfensters schreiben müssen.
accept	<u>ak accept</u>	Keine (lässt sich über den Befehl OBJECT SET ENABLED verwalten) (**)	Schaltflächen, Menübefehlen	Sichert einen neuen oder geänderten Datensatz, löst also entweder <u>On Saving New Record Event</u> oder <u>On Saving Existing Record Event</u> aus. Akzeptiert auch ein Formular, das über den Befehl DIALOG angezeigt wird. Löst in allen Fällen das Formularereignis <u>On Validate</u> aus.
cancel	<u>ak cancel</u>	Keine (lässt sich über den Befehl OBJECT SET ENABLED verwalten)(**)	Schaltflächen, Menübefehlen	Schließt den aktuellen Datensatz ohne Speichern der Änderungen. Kann auch ein Formular schließen, das mit dem Befehl DIALOG angezeigt wird, bzw. ein Formular, das über den Befehl DISPLAY SELECTION oder MODIFY SELECTION eine Datensatzauswahl anzeigt.
deleteRecord	<u>ak delete record</u>	Ein Datensatz ist ausgewählt und nicht ein neu hinzugefügter Datensatz	Schaltflächen, Menübefehlen	Zeigt eine Meldung, die den Benutzer auffordert, das Löschen zu bestätigen. Klickt er in dieser Meldung auf die Schaltfläche OK , wird der aktuelle Datensatz gelöscht. Nach Anklicken einer Schaltfläche Datensatz löschen kehrt 4D automatisch zum Ausgabeformular zurück.
nextRecord	<u>ak next record</u>	Ein Datensatz ist ausgewählt und nicht der letzte der Auswahl	Schaltflächen, Menübefehlen	Akzeptiert den aktuellen Datensatz und macht dann den nächsten Datensatz zum aktuellen.
previousRecord	<u>ak previous record</u>	Ein Datensatz ist ausgewählt und nicht der erste der Auswahl	Schaltflächen, Menübefehlen	Akzeptiert den aktuellen Datensatz und macht dann den vorigen Datensatz zum aktuellen.
firstRecord	<u>ak first record</u>	Ein Datensatz ist ausgewählt und nicht der erste der Auswahl	Schaltflächen, Menübefehlen	Akzeptiert den aktuellen Datensatz und macht dann den ersten Datensatz zum aktuellen.
lastRecord	<u>ak last record</u>	Ein Datensatz ist ausgewählt und nicht der letzte der Auswahl	Schaltflächen, Menübefehlen	Akzeptiert den aktuellen Datensatz und macht dann den letzten Datensatz zum aktuellen.
nextPage	<u>ak next page</u>	Mehrseitiges Formular und Sie sind nicht auf der letzten Seite	Schaltflächen, Menübefehlen	Zeigt die nächste Seite an.
previousPage	<u>ak previous page</u>	Mehrseitiges Formular und Sie sind nicht auf der ersten Seite	Schaltflächen, Menübefehlen	Zeigt die vorige Seite an.
firstPage	<u>ak next page</u>	Mehrseitiges Formular und Sie sind nicht auf der ersten Seite	Schaltflächen, Menübefehlen	Zeigt die erste Seite an.
lastPage	<u>ak last page</u>	Mehrseitiges Formular und Sie sind nicht auf der letzten Seite	Schaltflächen, Menübefehlen	Zeigt die letzte Seite an.
editSubrecord	<u>ak edit subrecord</u>	<ul style="list-style-type: none"> <i>Listbox</i>: Mindestens eine Zeile in der Listbox vom Typ Auswahl ist markiert. <i>Unterformular</i>: Hat Fokus und darin ist ein 	Schaltflächen, Menübefehlen	<ul style="list-style-type: none"> <i>Listbox</i>: Der entsprechende Datensatz zur Zeile der Listbox erscheint im Detailformular, das für die Listbox definiert wurde. Der Benutzer kann die Werte verändern, dann das Formular bestätigen oder annullieren, um zur Listbox zurückzukehren (siehe auch Standardaktionen verwenden). <i>Unterformular</i>: Der gewählte Unterdatensatz wird editierbar, entweder direkt in der Liste oder im

		<ul style="list-style-type: none"> • Datensatz ausgewählt. • <i>Listenformular:</i> Ein Datensatz ist in der Liste ausgewählt. 		<p>zugeordneten Detailformular – je nach den Eigenschaften des Unterformulars.</p> <ul style="list-style-type: none"> • <i>Listenformular:</i> Der gewählte Datensatz wird editierbar. In Listenformularen, die über die 4D Befehle MODIFY SELECTION / DISPLAY SELECTION angezeigt werden, wird die Änderung in der Liste oder auf der Detail-Seite ausgeführt, je nach dem Wert, der im Parameter <i>EingListe</i> übergeben wurde. Im Anzeigefenster der Datensätze erfolgt die Änderung auf der Detail-Seite (Die Aktion ist dasselbe wie ein Doppelklick). • <i>Listbox:</i> Der entsprechende Datensatz zur Zeile der Listbox erscheint im Detailformular, das für die Listbox definiert wurde, im Nur-Lesen Modus. Der Benutzer kann das Formular nur annullieren, um zur Listbox zurückzukehren. (siehe auch Standardaktionen verwenden). • <i>Unterformular:</i> Der gewählte Unterdatensatz erscheint im zugeordneten Detailformular im Nur-Lesen Modus (wenn in den Eigenschaften des Unterformulars definiert). • <i>Listenformular:</i> Mit Listen, die über die Befehle MODIFY SELECTION / DISPLAY SELECTION angezeigt werden, erscheint der gewählte Datensatz auf der Detail-Seite im Nur-Lesen Modus, je nach dem Wert, der im Parameter <i>EingListe</i> übergeben wurde. Im Anzeigefenster der Datensätze erscheint er im Nur-Lesen Modus auf der Detail-Seite.
displaySubrecord	<u>ak_display_subrecord</u>	<ul style="list-style-type: none"> • <i>Listbox:</i> Mindestens eine Zeile in der Listbox vom Typ Auswahl ist markiert. • <i>Unterformular:</i> Hat Fokus und darin ist ein Datensatz ausgewählt. • <i>Listenformular:</i> Ein Datensatz ist in der Liste ausgewählt. 	Schaltflächen, Menübefehlen	
deleteSubrecord	<u>ak_delete_subrecord</u>	<ul style="list-style-type: none"> • <i>Listbox:</i> Mindestens eine Zeile in der Listbox vom Typ Auswahl ist markiert. • <i>Unterformular:</i> Hat Fokus und darin ist ein Datensatz ausgewählt. • <i>Listenformular:</i> Ein Datensatz ist in der Liste ausgewählt. 	Schaltflächen, Menübefehlen	<ul style="list-style-type: none"> • <i>Listbox:</i> Ein Dialogfenster zum Bestätigen erscheint, so dass der Benutzer das Löschen bestätigen oder annullieren kann (siehe auch Standardaktionen verwenden). • <i>Unterformular:</i> Es erscheint ein Dialogfenster, um Löschen der ausgewählten Unterdatensätze zu bestätigen oder annullieren. • <i>Listenformular:</i> Es erscheint ein Dialogfenster, um Löschen der ausgewählten Unterdatensätze zu bestätigen oder annullieren.
addSubrecord	<u>ak_add_subrecord</u>	<ul style="list-style-type: none"> • <i>Listbox:</i> Mindestens eine Zeile in der Listbox vom Typ Auswahl ist markiert. • <i>Unterformular:</i> Hat Fokus • <i>Listenformular:</i> keins 	Schaltflächen, Menübefehlen	<ul style="list-style-type: none"> • <i>Listbox:</i> Ein neuer leerer Datensatz erscheint im Detailformular, das für die Listbox definiert wurde. Der Benutzer kann Werte eingeben, dann den Datensatz bestätigen und es erscheint automatisch ein neuer leerer Datensatz. Das läuft weiter, bis der Benutzer auf eine Schaltfläche Abbrechen klickt (siehe auch Standardaktionen verwenden). • <i>Unterformular:</i> 4D erstellt einen neuen Datensatz in der Tabelle bzw. der verknüpften Tabelle, entweder direkt in der Liste oder im zugeordneten Detailformular – je nach den Eigenschaften des Unterformulars. • <i>Listenformular:</i> Ein neuer leerer Datensatz wird erstellt. In Listen, die über die 4D Befehle MODIFY SELECTION / DISPLAY SELECTION erscheinen, wird der Datensatz in der Liste oder auf der Detail-Seite

automaticSplitter	ak_automatic_splitter	Keine (lässt sich über den Befehl OBJECT SET ENABLED verwalten)	Unsichtbare Schaltflächen	hinzugefügt, je nach dem Wert, der im Parameter <i>EingListe</i> übergeben wurde. Im Anzeigefenster der Datensätze wird der Datensatz in der Liste hinzugefügt. Damit können Sie in Ihren Formularen eigene Splitter einrichten. Diese Aktion lässt sich nur einem Objekt vom Typ unsichtbare Schaltfläche zuordnen (siehe Schaltflächen). Eine unsichtbare Schaltfläche mit dieser Aktion funktioniert genauso wie ein Splitter. Legen Sie zum Beispiel ein Bild auf eine unsichtbare Schaltfläche, können Sie eine eigene Oberfläche für Ihre Splitter erstellen. Weitere Informationen dazu finden Sie im Abschnitt Splitter .
gotoPage	ak_goto_page	Mehrseitiges Formular	Registerkarten, Listboxen, Schaltflächengitter, PopUp-Menüs	Zeigt die Formularseite, die der Nummer des gewählten Eintrags (Registerkarte, Zeile der Listbox, Schaltfläche im Gitter, Eintrag im PopUp-Menü) entspricht. Weitere Informationen dazu finden Sie im Abschnitt Aktion Gehe zu Seite .
gotoPage? value=<page>	ak_goto_page	Mehrseitiges Formular	Schaltflächen, Menübefehlen	Zeigt die Formularseite, die der Nummer <page> entspricht

(*) Siehe unten **Hinweise zu Objekten und Aktionen**

(**) Mit der Funktion **Dynamic pop up menu** wird je nach Kontext ein Element dieser Aktion nicht automatisch ausgeblendet.

Aktionen für Web Area

Folgende Standardaktionen sind nur für **Web Areas** verfügbar.

Aktionsname	Konstante (falls vorhanden)	Bedingungen zur Aktivierung	Verfügbar mit (*)	Beschreibung
openBackURL	ak_open_back_url	Eine vorige URL wurde geladen	Schaltflächen, Menübefehlen	Öffnet die vorherige URL in der Browser Sequenz, die vom Benutzer im Web Bereich ausgeführt wurde. Deaktiviert, wenn es keine vorige URL gibt; d.h. der Benutzer hat nur eine einzelne Seite im Bereich aufgerufen.
openForwardURL	ak_open_forward_url	Zuvor ausgeführte Aktion OpenBackURL	Schaltflächen, Menübefehlen	Öffnet die nächste URL in der Browser Sequenz, die vom Benutzer im Web Bereich ausgeführt wurde. Deaktiviert, wenn es keine nächste URL gibt; d.h. der Benutzer ist nie eine Seite zurückgesprungen.
refreshCurrentURL	ak_refresh_current_url	Zuvor ausgeführte Aktion OpenBackURL (lässt sich über den Befehl OBJECT SET ENABLED verwalten)	Schaltflächen, Menübefehlen	Lädt erneut den aktuellen Inhalt des Web Bereichs.
StopLoadingURL	ak_stop_loading_url	URL wird gerade geladen	Schaltflächen, Menübefehlen	Stoppt das Laden der Seite bzw. Objekte der aktuellen URL im Web Bereich.

(*) Siehe unten **Hinweise zu Objekten und Aktionen**

Aktionen der Anwendung

Dieser Abschnitt beschreibt Standardaktionen, die 4D Dialogfenster aufrufen oder die 4D Anwendung beenden.

Aktionsname	Konstante (wenn vorhanden)	Bedingungen zur Aktivierung	Verfügbar mit (*)	Beschreibung
databaseSettings	ak_database_settings	Keine (lässt sich über den Befehl OBJECT SET ENABLED verwalten)(**)	Schaltflächen, Menübefehle	Zeigt (***) das Standard Dialogfenster Datenbank-Eigenschaften.
quit	ak_quit	Keine (lässt sich über den Befehl OBJECT SET ENABLED verwalten)(**)	Schaltflächen, Menübefehlen	Zeigt eine Meldung (***) "Sind Sie sicher?" und beendet die 4D Anwendung, wenn diese bestätigt wird. Ansonsten wird die Operation annulliert. Wird diese Aktion einer Schaltfläche mit einer Objektmethode zugeordnet, läuft folgende Sequenz ab: Das Dialogfenster zum Bestätigen erscheint; wird es bestätigt, führt 4D die Objektmethode aus. Anschließend wird die Anwendung beendet.
designMode	ak_return_to_design_mode	Anwendungsmodus (lässt sich über den Befehl OBJECT SET ENABLED verwalten)(**)	Schaltflächen, Menübefehlen	Bringt die Fenster und Menüleisten der 4D Designumgebung nach vorne. Läuft die Datenbank im interpretierten Modus, erscheint das aktuelle Fenster der Designumgebung. Im kompilierten Modus erscheint das Fenster der Datensätze der aktuellen Tabelle (im kompilierten Modus ist nur der Zugriff auf Datensätze möglich).
msc	ak_msc	Keine (lässt sich über den Befehl OBJECT SET ENABLED verwalten)(**)	Schaltflächen, Menübefehlen	Zeigt das Fenster Maintenance und Security Center .

(*) Siehe unten **Hinweise zu Objekten und Aktionen**.

(**) Mit der Funktion **Dynamic pop up menu** wird je nach Kontext ein Element dieser Aktion nicht automatisch ausgeblendet.

(***) Auf macOS liegen die Menübefehle, welche den Aktionen **Beenden** und **Datenbank-Eigenschaften** zugewiesen sind, automatisch im Systemmenü der Anwendung. Das vereinfacht die Verwaltung des Befehls **Beenden** auf macOS.

Aktionen Bearbeiten

Dieser Abschnitt beschreibt die Standardaktionen zum Bearbeiten. Sie sind verwendbar in:

- Editierbaren Standardbereichen
- Textbereichen mit Mehrfachstil
- 4D Write Pro Bereichen

Aktionsname	Konstante (wenn vorhanden)	Bedingungen zur Aktivierung	Verfügbar mit (*)	Beschreibung
undo	ak_undo	Eine Aktion Bearbeiten wurde ausgeführt	Schaltflächen, Menübefehlen	Annulliert die zuletzt ausgeführte Aktion (=Befehl Widerrufen im Menü Bearbeiten). Verwechseln Sie Widerrufen nicht mit Abbrechen (= annulliert alle Änderungen am Datensatz und kehrt zum Ausgabeformular zurück).
redo	ak_redo	Eine Aktion Widerrufen wurde ausgeführt	Schaltflächen, Menübefehlen	Wiederholt die zuletzt annullierte Aktion (= Befehl Erneut im Menü Bearbeiten).
cut	ak_cut	Ausgewählter Inhalt	Schaltflächen, Menübefehlen	Entfernt die Auswahl und legt sie in die Zwischenablage.
copy	ak_copy	Ausgewählter Inhalt	Schaltflächen, Menübefehlen	Setzt eine Kopie der Auswahl in die Zwischenablage.
paste	ak_paste	Zwischenablage ist nicht leer	Schaltflächen, Menübefehlen	Fügt den Inhalt aus der Zwischenablage am Einfügepunkt ein.
clear	ak_clear	Objekt mit Fokus hat editierbaren Bereich	Schaltflächen, Menübefehlen	Löscht die Auswahl. Gibt es keine Auswahl, wird der gesamte Bereich mit dem Cursor gelöscht (nur für eingebare Bereiche).
selectAll	ak_select_all	Objekt mit Fokus hat editierbaren Bereich	Schaltflächen, Menübefehlen	Wählt alle auswählbaren Elemente im Kontext aus.
showClipboard	ak_show_clipboard	Immer verfügbar	Schaltflächen, Menübefehlen	Öffnet ein neues Fenster mit dem aktuellen Inhalt der Zwischenablage.

Hinweis: Bedingung zur Aktivierung ist immer, dass der editierbare Bereich den Fokus hat (außer ZeigeZwischenablage).

(*) Siehe unten **Hinweise zu Objekten und Aktionen**.

Aktionen für Schriften, Ausdrücke und Rechtschreibprüfung

Diese Standardaktionen sind verfügbar für:

- Textbereiche mit Mehrfachstil (auch *Rich Text Bereich* genannt)
- 4D Write Pro Bereiche

Hinweis: Zusätzliche Aktionen sind nur für 4D Write Pro Bereiche verfügbar. Sie werden im Abschnitt **4D Write Pro Standardaktionen verwenden** des 4D Write Pro Handbuchs beschrieben.

Schrift

Aktionsname	Konstante (wenn vorhanden)	Bedingungen zur Aktivierung	Verfügbar mit (*)	Beschreibung
backgroundColor	<u>ak</u> <u>background</u> <u>color</u>	Keine	Menübefehlen, PopUp/DropDown Listen, hierarchischen PopUp-Menüs	Zeigt das Untermenü für Standard Hintergrundfarbe
backgroundColor?value= <color>	<u>ak</u> <u>background</u> <u>color</u>	Keine	Schaltflächen, Menübefehlen, Listeneinträgen	Setzt die Hintergrundfarbe auf <color>. Sie übergeben einen Css Wert für Farbe oder Name. z.B.: <i>backgroundColor?</i> <i>value=#FF0000</i> , <i>backgroundColor?value=red</i> , <i>backgroundColor?</i> <i>value=transparent</i>
backgroundColor/showDialog	<u>ak</u> <u>background</u> <u>color dialog</u>	Keine	Schaltflächen, Menübefehlen	Öffnet das Dialogfenster für Hintergrundfarbe für Schrift
color	<u>ak font color</u>	Keine	Menübefehlen, PopUp/DropDown Listen, hierarchischen PopUp-Menüs	Zeigt das Untermenü für Standard Schriftfarbe
color?value=<color>	<u>ak font color</u>	Keine	Schaltflächen, Menübefehlen, Listeneinträgen	Setzt die Schriftfarbe auf <color>. Sie übergeben einen Css Wert für Farbe oder Name. z.B.: <i>color?value=#FF0000</i> , <i>color?value=red</i>
color/showDialog	<u>ak font color</u> <u>dialog</u>	Keine	Schaltflächen, Menübefehlen	Zeigt das Dialogfenster des Systems für Schriftfarbe.
font/showDialog	<u>ak font show</u> <u>dialog</u>	Keine	Schaltflächen, Menübefehlen	Zeigt das Dialogfenster für Systemschriften.
fontItalic	<u>ak font italic</u>	Keine	Schaltflächen, Menübefehlen	Schaltet um auf das Attribut Kursivschrift.
fontBold	<u>ak font bold</u>	Keine	Schaltflächen, Menübefehlen	Schaltet um auf das Attribut Fettschrift.
fontLinethrough	<u>ak font</u> <u>linethrough</u>	Keine	Schaltflächen, Menübefehlen	Schaltet um auf das Attribut Schrift Durchgestrichen.
fontSize	<u>ak font size</u>	Keine	Menübefehlen, PopUp/DropDown Listen, hierarchischen PopUp- Menüs[#tab/Zeigt das Untermenü für Standard Schriftfarbe	
fontSize?value=<size>	<u>ak font size</u>	Keine	Schaltflächen, Menübefehlen, Listeneinträgen	Setzt die Schriftgröße auf <size>. Sie übergeben einen Css Wert für Länge in pt. z.B.: <i>fontSize?</i> <i>value=12pt</i>
fontUnderline	<u>ak font</u> <u>underline</u>	Keine	Schaltflächen, Menübefehlen	Schaltet um auf das Attribut Schrift Unterstrichen.
fontStyle	<u>ak font style</u>	Keine	Menübefehlen, PopUp/DropDown Listen, hierarchischen PopUp-Menüs	Zeigt das Standard Untermenü für Schriftstil.

(*) Siehe unten **Hinweise zu Objekten und Aktionen**.

Hinweis: Wird ein Stilattribut über eine Standardaktion geändert, generiert 4D das Formularereignis On After Edit.

Dynamische Ausdrücke

Aktionsname	Konstante (wenn vorhanden)	Bedingungen für Aktivierung	Verfügbar mit (*)	Beschreibung
computeExpressions	<u>ak compute expressions</u>	Keine	Schaltflächen, Menübefehlen	Aktualisiert alle dynamischen Ausdrücke im Bereich.
freezeExpressions	<u>ak freeze expressions</u>	Keine	Schaltflächen, Menübefehlen	Friert alle dynamischen Ausdrücke im Bereich ein.
visibleReferences	<u>ak show reference</u>	Keine	Schaltflächen, Menübefehlen	Zeigt alle dynamischen Ausdrücke im Bereich als Referenzen an.

Rechtschreibprüfung

Aktionen für Rechtschreibprüfung sind nur verfügbar, wenn für den Bereich die Option **Rechtschreibprüfung** markiert wurde.

Aktionsname	Konstante (falls vorhanden)	Bedingungen zur Aktivierung	Verfügbar mit (*)	Beschreibung
spell/autoCorrectionEnabled	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Aktiviert den automatischen Korrekturmodus.
spell/autoDashSubstitutionsEnabled	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Ersetzt doppelte Bindestriche (--) bei der Eingabe mit Gedankenstrichen (—) (nur macOS).
spell/autoLanguageEnabled	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Identifiziert die Sprache des Wörterbuchs anhand des Textinhalts (nur macOS).
spell/autoQuoteSubstitutionsEnabled	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Ersetzt gerade Anführungszeichen durch gebogene (nur macOS).
spell/autoSubstitutionsEnabled	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Aktiviert Ersetzen von Text
spell/enabled	-	Keine	Schaltflächen, Menübefehlen	Aktiviert Rechtschreibprüfung im Bereich (die Option Rechtschreibprüfung muss für den Bereich markiert sein).
spell/forgetIgnore	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Leert die Liste der ignorierten Wörter.
spell/grammarEnabled	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Aktiviert die Grammatikprüfung von Text (nur macOS).
spell/ignore	-	Rechtschreibprüfung ist aktiviert/Ein unbekanntes Wort ist ausgewählt oder hat den Cursor	Schaltflächen, Menübefehlen	Unbekanntes Wort bleibt bestehen und wird nicht mehr unterstrichen.
spell/learn	-	Rechtschreibprüfung ist aktiviert/Ein unbekanntes Wort ist ausgewählt oder hat den Cursor	Schaltflächen, Menübefehlen	Unbekanntes Wort wird im Wörterbuch hinzugefügt.
spell/removeSubstitution	-	Rechtschreibprüfung ist aktiviert/Ein ersetztes Wort ist ausgewählt oder hat den Cursor	Schaltflächen, Menübefehlen	Entfernt die ausgewählte Ersetzung.
spell/showDialog	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Zeigt ein Dialogfenster Rechtschreibprüfung.
spell/suggestion?index=<1<=number<=10>	-	Rechtschreibprüfung ist aktiviert/falsch geschriebenes Wort ist ausgewählt	Schaltflächen, Menübefehlen	Zahl ist Nr. des Korrekturvorschlags für das aktuelle erste falsch geschriebene Wort in der Auswahl. z.B.: <i>spell/suggestion?index=1</i> ersetzt das aktuelle falsch geschriebene Wort in der Ansicht mit Fokus durch den ersten Korrekturvorschlag
spell/unLearn	-	Rechtschreibprüfung ist aktiviert/Ein gelerntes Wort ist ausgewählt oder hat den Cursor	Schaltflächen, Menübefehlen	Entfernt das ausgewählte gelernte Wort aus der Liste der gelernten Wörter
spell/visibleSubstitutions	-	Rechtschreibprüfung ist aktiviert	Schaltflächen, Menübefehlen	Unterstreicht mögliche Ersetzungen im Text in Blau.
spell	-	Keine	Menübefehlen	Zeigt das gesamte Menü Rechtschreibhilfe an.

(*) Siehe unten [Hinweise zu Objekten und Aktionen](#)

Hinweise zu Objekten und Aktionen

- *Schaltflächen* bedeutet Standardschaltflächen, 3D Schaltflächen, Invertierte Schaltflächen, Bildschaltflächen und Unsichtbare Schaltflächen. Dazu gehören auch Optionsfelder und 3D Optionsfelder, die Aktionen mit dem Status wahr/falsch darstellen können, z.B. "fontBold" (Die Option drei Stadien wird nur mit Optionsfeldern unterstützt).
- *Den Objekten Pop-Up/Drop-down Liste* und *hierarchisches PopUp-Menü* lassen sich nur Standardaktionen zuweisen, die ein Untermenü (Liste), wie "BackgroundColor" oder "FontStyle" generieren. In diesem Fall zeigen sie eine automatische Liste mit Werten, außer den Listeneinträgen wurden eigene Standardaktionen zugewiesen (siehe nächster Punkt).
- *Listeneinträge*: Wollen Sie keine automatischen Werte zuweisen, können Sie den Einträgen eigene Standardaktionen zuordnen (über den Listeneditor oder den Befehl **SET LIST ITEM PARAMETER**) und die Liste als "Auswahlliste" für die PopUp/DropDown Listen und hierarchischen PopUp Menüs setzen. Automatische Werte werden in Echtzeit durch eigene Aktionen ersetzt. In diesem Kontext lassen sich nur Standardaktionen mit Werten in Bezug auf die Hauptaktion des Untermenüs (Liste) verwenden. Sie können z.B. eine Liste mit Einträgen für Werte der Aktion BackgroundColor definieren (backgroundColor?value="rot", backgroundColor?value="blau"...) und sie als Auswahlliste für ein hierarchisches PopUp-Menü setzen.

Überblick

4D bietet eine integrierte und anpassbare Rechtschreibhilfe an. Sie lässt sich auf Variablen und Felder vom Typ Alpha oder Text anwenden, sowie auf 4D Write Pro Dokumente.

Die Korrektur lässt sich automatisch oder per Programmierung auslösen. Korrekturaktionen werden über die Standardoberfläche oder Standardaktionen ausgeführt.

Sie können die zu verwendende Rechtschreibhilfe und Sprachwörterbücher konfigurieren. 4D bietet auch ein spezifisches Feature zum Einsatz von spezifischen Wörterbüchern.

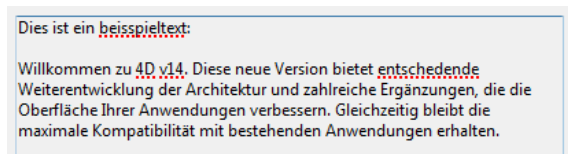
Aktivierung und Verwendung

Es gibt folgende Optionen, um die Rechtschreibprüfung in Ihren Formularen zu aktivieren:

- Sie markieren die Eigenschaft **Rechtschreibprüfung** unter der Gruppe "Eingabe" für jedes Objekt (siehe **Rechtschreibprüfung**). Die Rechtschreibprüfung wird dann automatisch beim Eingeben von Daten ausgeführt. Sie können den Korrekturvorgang auch über **Standardaktionen** verwalten.
- Sie rufen den Befehl **SPELL CHECKING** für jedes zu prüfende Objekt auf.

Prüfungsprozess

In 4D wird die Rechtschreibprüfung kontinuierlich in Textbereichen angewandt. Fehler werden direkt im Text mit einer gestrichelten Linie hervorgehoben:



Hinweis: Für das Objekt muss in der Eigenschaftsliste unter der Gruppe Eingabe die Option "Rechtschreibprüfung" markiert sein.

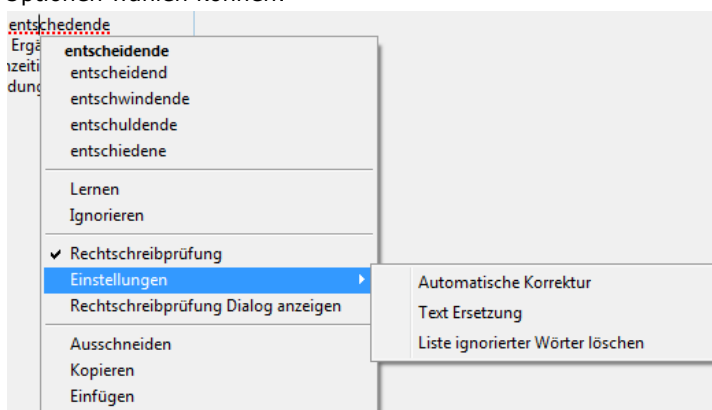
Die gestrichelten Linien haben verschiedene Farben, die die Art der Korrektur anzeigen:

- **rot** für Rechtschreibfehler
- **grün** für Grammatikfehler (nur mit der native OS X Rechtschreibprüfung in Englisch)
- **blau** für zu ersetzende Wörter (wenn die Option **Text Ersetzung** aktiviert ist, siehe unten)

Der Benutzer kann dann den Text über das Kontextmenü (rechter Mausklick auf das unterstrichene Wort) oder über das Dialogfenster Rechtschreibprüfung verbessern.

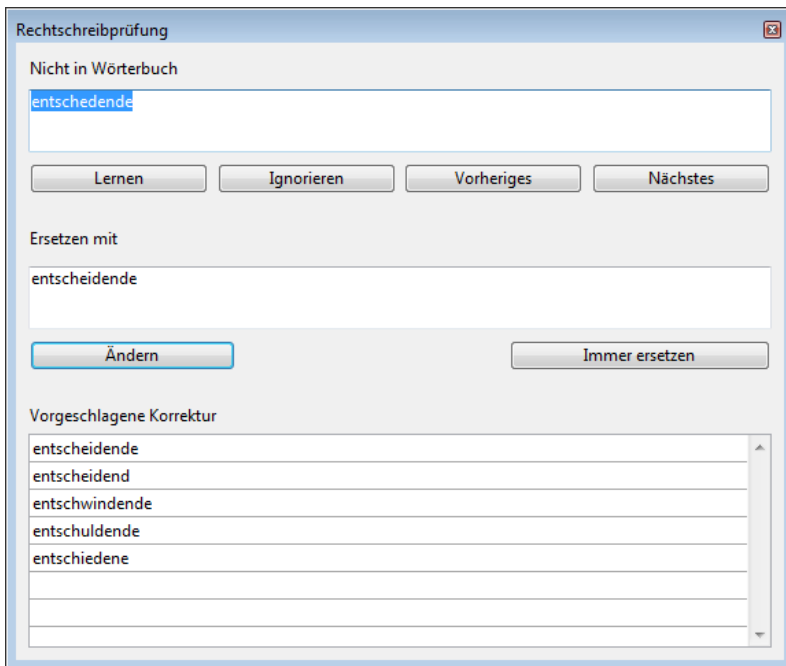
Kontextmenü oder Dialogfenster Rechtschreibprüfung

Klicken Sie mit der rechten Maustaste auf ein unbekanntes Wort, öffnet sich das Kontextmenü, wo Sie verschiedene Optionen wählen können:



Hinweis: Für das Objekt müssen in der Eigenschaftsliste unter der Gruppe **Eingabe** die Optionen "Rechtschreibprüfung" und "Kontextmenü" markiert sein.

Wählen Sie die Option **Rechtschreibprüfung Dialog anzeigen**, erscheint ein Dialog, wo Sie eine Korrektur eingeben können:



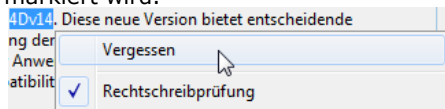
Dieses Dialogfenster ist ein Palettenfenster, das während einer Sitzung für alle Fenster der Anwendung verfügbar bleibt, bis der Benutzer es schließt.

Funktionen der Rechtschreibprüfung

Die Rechtschreibprüfung bietet nicht nur Korrekturvorschläge, sondern auch folgende Optionen und Funktionen (im Kontextmenü bzw. im Dialogfenster Rechtschreibprüfung):

- **Lernen:** Das unbekannte Wort wird im Wörterbuch hinzugefügt; es wird nicht mehr von der Rechtschreibprüfung markiert.
 - Auf OS X bleiben gelernte Wörter in `/Users/[UserName]/Library/Spelling` dauerhaft erhalten, d.h. ein in der Rechtschreibhilfe des Systems gelerntes Wort gilt auch für alle Anwendungen, die die Rechtschreibhilfe des Systems nutzen.
 - Mit Hunspell werden gelernte Wörter im Benutzer Systemverzeichnis der aktuellen Anwendungsdaten in einem eigenen Hunspell Wörterbuch abgelegt, das immer zusammen mit dem Hauptwörterbuch geladen wird (wie in bisherigen Versionen).

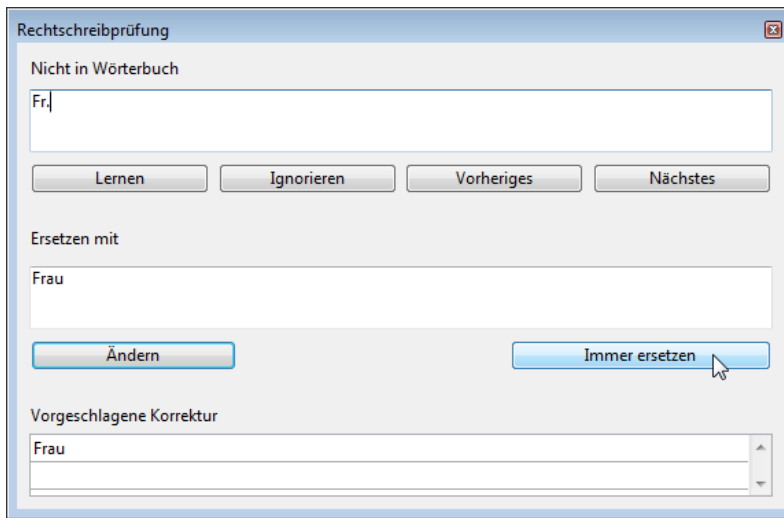
- **Vergessen** (Kontextmenü): Diese Option erscheint, wenn ein zuvor gelerntes Wort ausgewählt wird. Der Benutzer kann dieses Wort aus der Liste der gelernten Wörter entfernen, so dass es wieder als möglicher Rechtschreibfehler markiert wird.



- **Ignorieren:** Das unbekannte Wort bleibt unverändert und wird nicht mehr unterstrichen. Es wird jedoch wieder markiert, wenn es später gefunden wird. Die Rechtschreibhilfe führt für jedes Dokument eine Tabelle mit den zu ignorierenden Wörtern. Mit der Option **Liste ignorerter Wörter löschen** können Sie diese Tabelle entfernen (siehe unten).
- **Vorheriges / Nächstes** (Dialogfenster): Das unbekannte Wort wird ignoriert, bleibt aber unterstrichen und die Rechtschreibprüfung überprüft das vorige bzw. nächste unbekannte Wort im Text.
- **Rechtschreibprüfung** (Kontextmenü): Aktiviert oder deaktiviert die Rechtschreibprüfung im Bereich für den aktuellen Prozess.
- **Ändern** (Dialogfenster): Das im Text markierte Wort erscheint im ersten eingebbaren Bereich, ein Korrekturvorschlag im zweiten eingebbaren Bereich. Klicken Sie darunter auf die Schaltfläche **Ändern**, wird es im Text ersetzt und die Hilfe springt zum nächsten Fehler
- **Immer ersetzen** (Dialogfenster): Funktioniert wie **Ändern**, das ersetzte Wort wird jedoch gespeichert (siehe unten **Text Ersetzung**).

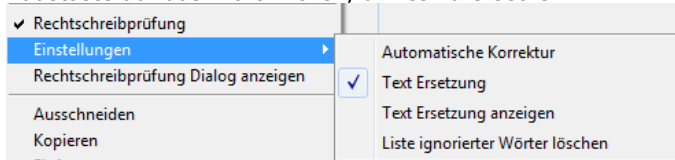
Das Untermenü **Einstellungen** > bietet folgende Funktionen:

- **Automatische Korrektur** aktiviert oder deaktiviert den Modus automatische Korrektur im Bereich für den aktuellen Prozess. In diesem Modus werden unbekannte Wörter automatisch durch das nächstgelegene bekannte Wort ersetzt - außer bei signifikanter Zweideutigkeit. Die Korrekturen werden beim Eingeben durchgeführt. Die automatische Korrektur ist standardmäßig deaktiviert.
- **Text Ersetzung** aktiviert oder deaktiviert das Ersetzen von Text, d.h. ein Wort wird durch ein anderes ersetzt. Sie können z.B. definieren, dass "Fr" durch "Frau" ersetzt wird. Die Wortpaare "Zu ersetzen"/"Ersetzen mit" erstellen Sie im Dialogfenster Rechtschreibprüfung: Geben Sie das zu ersetzende Wort im Bereich "Nicht im Wörterbuch" ein und das neue Wort im Bereich "Ersetzen mit". Klicken Sie anschließend auf die Schaltfläche **Immer ersetzen**:



Das Ersetzen wird in der ganzen Anwendung durchgeführt. Mit Hunspell führt die Rechtschreibprüfung eine globale Tabelle mit den Ersetzungen für die Anwendung, die im Benutzer Systemverzeichnis der aktuellen Anwendung gesichert wird; für die Rechtschreibhilfe in OS X wird diese Tabelle mit den System Ersetzungen zusammengelegt (Option "Symbol- und Textersetzung in den System Einstellungen). Die Operation zum Ersetzen läuft unterschiedlich, je nachdem, ob die Option **Text Ersetzung anzeigen** im Untermenü **Einstellungen** markiert ist (siehe nächster Absatz).

- **Text Ersetzung anzeigen** (diese Option erscheint nur, wenn die Option Text Ersetzung markiert ist): Dann unterstreicht die Rechtschreibprüfung mögliche Ersetzungen im Text in blau. Der Benutzer muss mit der rechten Maustaste auf das Wort klicken, um es zu ersetzen.

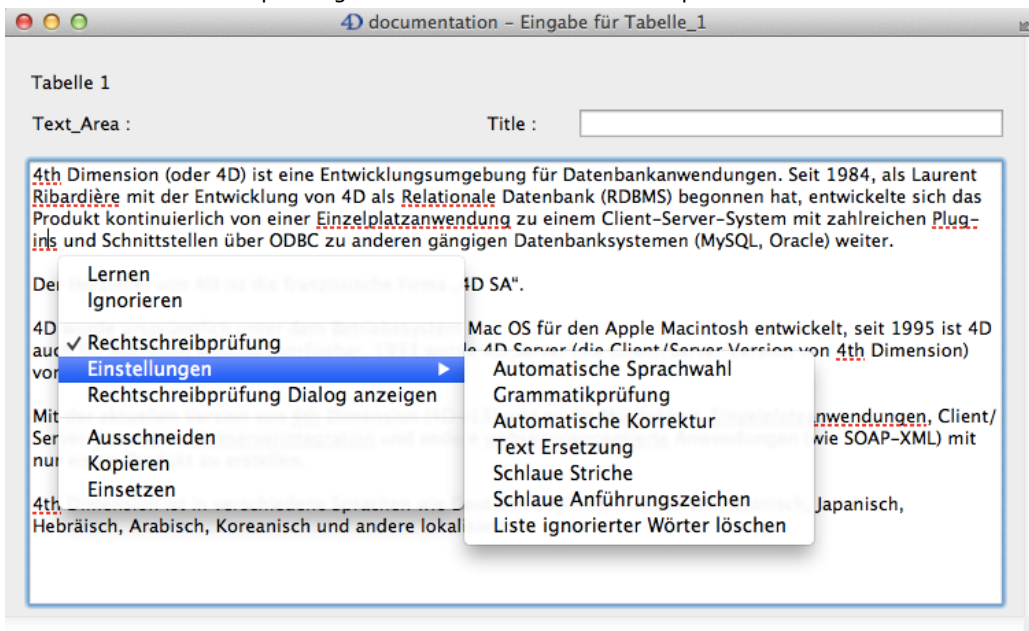


Ist diese Option nicht markiert, laufen die Ersetzungen automatisch, d.h. der Benutzer muss nicht aktiv werden.

- **Liste ignorerter Wörter löschen** löscht die Liste der Wörter, die im Dokument ignoriert werden sollen

Zusatzoptionen für native Rechtschreibhilfe auf OS X

Die native Rechtschreibprüfung in OS X bietet noch weitere Optionen:



- **Automatische Sprachwahl** identifiziert anhand des Textinhalts automatisch die passende Sprache des Wörterbuchs. Diese Rechtschreibhilfe verwendet standardmäßig die Sprache der 4D Anwendung, so wie die Hunspell Rechtschreibhilfe.
- **Grammatikprüfung** aktiviert die Grammatikprüfung des Textes. Diese Prüfung ist nur für englische Texte verfügbar.
- **Schlaue Striche** Ersetzt doppelte Bindestriche (--) während der Eingabe durch Gedankenstriche (—).
- **Schlaue Anführungszeichen** ersetzt gerade Anführungszeichen durch an die aktuelle Sprache angepasste Anführungszeichen.

Rechtschreibprüfung konfigurieren

4D Anwendungen verwenden folgende Rechtschreibhilfen:

- Windows - das *open source* "Hunspell" Wörterbuch.
- Mac - es gibt zwei Arten von Rechtschreibhilfen:
 - die native Rechtschreibprüfung des Systems. Diese wird standardmäßig eingesetzt.
 - die *open source* "Hunspell" Wörterbücher. Dazu müssen Sie den Befehl **SET DATABASE PARAMETER** mit dem Selector `Spellchecker` aufrufen.

Auf Mac hat die native Rechtschreibprüfung des Systems den Vorteil, dass sie in die Arbeitsumgebung integriert ist und zusätzliche Korrekturoptionen bietet (siehe **Zusatzoptionen für native Rechtschreibhilfe auf OS X**).

Das *open source* "Hunspell" Wörterbuch wird von vielen Programmen wie OpenOffice, Mozilla FireFox, Google Chrome, etc. verwendet. Dazu gehören eine große Anzahl zusätzlicher Sprachwörterbücher, die Sie je nach Bedarf in Ihren 4D Anwendungen installieren können. (siehe nächster Absatz **Unterstützung von Hunspell Wörterbüchern**).

Unterstützung von Hunspell Wörterbüchern

4D verwendet standardmäßig das Wörterbuch der aktuellen Sprache der Anwendung. Mit dem Befehl **SPELL SET CURRENT DICTIONARY** können Sie festlegen, dass ein anderes Wörterbuch geöffnet wird.

Bei Verwendung des Hunspell Wörterbuchs (siehe oben) können Sie zusätzliche Sprachwörterbücher hinzufügen. Sie können diese laden unter der Adresse: <https://openoffice.org>.

Jedes Hunspell Wörterbuch besteht aus einer `.aff` und einer `.dic` Datei mit demselben Namen. So besteht das Wörterbuch "fr-modern" aus den Dateien `fr-modern.aff` und `fr-modern.dic`. Zum Einsatz in einer 4D Applikation müssen Sie beide Dateien, also mit der Endung `.aff` und `.dic` installieren.

Hinweis: Wörterbuchdateien zum Herunterladen sind in der Regel komprimiert (z.B. `.oxt` Format auf OpenOffice.org) und müssen vor der Verwendung entkomprimiert werden.

Damit ein Hunspell Wörterbuch in einer 4D Applikation verwendbar ist, müssen Sie seine `.aff` und `.dic` Dateien in einem der folgenden Speicherorte auf oberster Ebene installieren:

- in der 4D Applikation: `<4D>/Resources/Spellcheck/Hunspell/`
- in der 4D Datenbank: `<Database_Files>/Resources/Hunspell`

Beide Speicherorte sind miteinander kompatibel: Der Datenbankordner wird zuerst durchlaufen und dann mit einem Ordner in der 4D Applikation vervollständigt, d.h. Sie können spezielle Wörterbücher in Ihre 4D Datenbanken einbinden. Gibt es an beiden Stellen Wörterbücher mit demselben Namen, hat das Wörterbuch in der Datenbank Priorität.

Spezialisierte Wörterbücher verwenden

4D ermöglicht, eigene Wörterlisten zu erstellen, die dann von der Rechtschreibprüfung berücksichtigt werden. So können Sie in einer Anwendung spezialisierte Wörterbücher verwenden, die Fachwörter zu einem bestimmten Beruf, Anwendungsbereich, Unternehmen, o.ä. enthalten.

Pro Hauptsprache sind mehrere Fachwörterbücher möglich. Diese Funktion arbeitet mit 4D und 4D Write Pro.

Installation

Um für eine Sprache ein Fachwörterbuch hinzuzufügen, legen Sie im Unterordner der Hauptsprache im Ordner *Spellcheck* eine Datei vom Typ Text an. Der Dateiname ist beliebig, er muss nur die Endung `.txt` haben (z.B. "Astronomie.txt").

Die Unterordner für Sprache liegen in 4D bereits an folgender Stelle:

- Unter Windows: `(4D Application folder)\Resources\Spellcheck\`
- Auf macOS: `(4D Package):Contents:Resources:Spellcheck:`

Ihre Namen richten sich nach der Hauptsprache. Sie dürfen nicht verändert werden:

- English
- French
- German
- Spanish
- Norsk

Jeder Ordner enthält eine standardmäßige Wörterliste. Sie können eigene Wörter in dieser Liste hinzufügen oder weitere Dateien hinzufügen.

Benutzerdateien müssen im Format UTF-8 gespeichert werden.

Client/Server

Die Rechtschreibdateien liegen in der remote 4D Anwendung. Wollen Sie in einer Client/Server-Lösung Fachwörterbücher einrichten, müssen Sie sicherstellen, dass diese Wörterbücher auf jedem Client-Rechner installiert sind, und zwar an der o.a. Stelle.

Inhalt der Dateien

Die Dateien der Fachwörterbücher enthalten Listen mit Wörtern, die jeweils durch Zeilenschaltung voneinander getrennt sind, wie im folgenden gezeigt:

```
Oliver
Olympia
Olympiasieger
Olympiastadion
```

Sie können einzelne Wörter oder zusammengesetzte Wörter mit Bindestrich hinzufügen, z.B. *Bildschirm* oder *Bildschirm-Schoner*. Wortsequenzen mit Leerzeichen sind nicht möglich, z.B. *Software Analyse*.

Das interne Format für Textdateien unterscheidet sich zwischen Windows und macOS, insbesondere wegen der Zeilenumbrüche. Deshalb müssen Sie ein Fachwörterbuch plattformspezifisch anlegen und zur Verfügung stellen. Um eine Datei von einer Plattform auf eine andere zu übertragen, müssen Sie zuerst die Zeilenumbrüche jeweils in das Format der Zielplattform umwandeln. Dafür können Sie z.B. einen Texteditor wie Notepad++ verwenden.

Laden und Einsetzen

Die Fachwörterbücher werden wie die Standardwörterbücher beim Starten der Anwendung geladen. Die aktuelle Sprache der Wörterbücher richtet sich nach der Anwendungssprache.

Während einer Arbeitssitzung können Sie das aktuelle Wörterbuch über den 4D Befehl **SPELL SET CURRENT DICTIONARY** ändern. Ist ein Fachwörterbuch in der gewählten Sprache vorhanden, wird es geladen. Bei Verwendung der Rechtschreibhilfe wird nicht unterschieden, ob die Wörter aus Standard- oder aus Fachwörterbüchern stammen.

Spiegeleffekt (Windows)

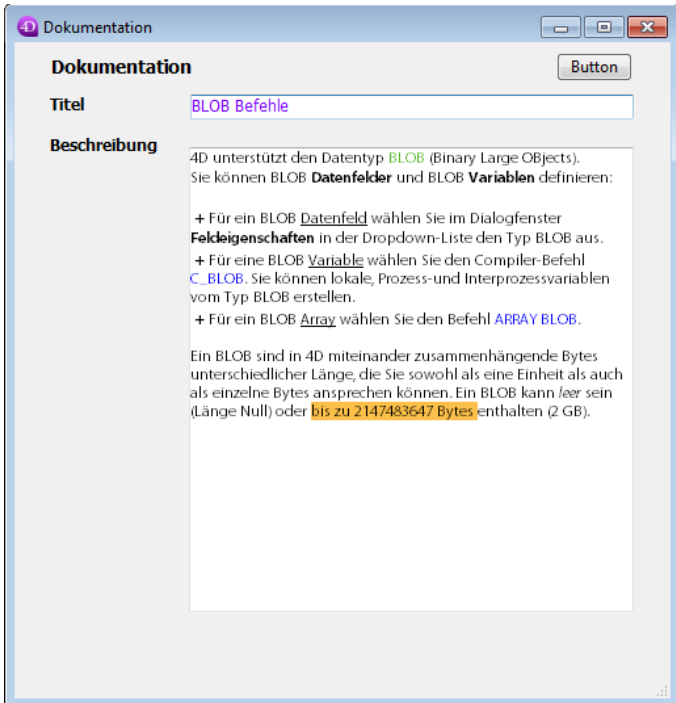
Die Option **Spiegeleffekt (Windows)** ist in der Eigenschaftenliste unter der Gruppe „Bild“ verfügbar. Sie lässt sich auf jedes aktive Objekt anwenden, das Bilder verwendet.

Ist diese Option markiert, wird das einem Objekt zugeordnete Bild invertiert dargestellt, wenn es in einer Anwendung mit dem Modus „rechts-nach-links“ erscheint. Das gilt für statische Bilder, Felder und Variablen vom Typ Bild, sowie Formularobjekte mit Bildern in Form von Bildschaltflächen, PopUp-Menüs, 3D Schaltflächen, usw. Weitere Informationen dazu finden Sie im Abschnitt **Statische Bilder verwenden**.

Text mit Stil (Rich Text)

Überblick

In 4D können Sie Textbereiche mit unterschiedlichen Stilarten verwenden, d.h. Sie können innerhalb eines Textbereichs Wörter in Fett-, Kursivschrift oder Farbe haben:



Diese Funktionalität gilt für Felder und Variablen vom Typ alphanumerisch und Text sowie Zellen von Listboxen. Sie wird für Detail- und Listenformulare unterstützt, sowohl für Anzeigen als auch für Drucken.

Hinweis: Für Eingabefilter, Schnellberichte und den Etiketteneditor können Sie keine Rich Text Bereiche verwenden.

Diese Funktionsweise lässt sich über spezifische Optionen in der Eigenschaftenliste einstellen.

Als Textattribute sind **Schrifttyp**, **Schriftgröße**, **Schriftstil** und **Farbe für Text** bzw. für **Hintergrund** verfügbar. Um Attribute in einem "Rich Text" Bereich zu ändern, gibt es folgende Möglichkeiten:

- Während der Ausführung über ein automatisches PopUp-Menü. Die Verfügbarkeit dieses Menüs lässt sich in der Eigenschaftenliste einstellen
- Über Standardaktionen, die Sie Menüs oder Schaltflächen zuweisen können
- Per Programmierung über den 4D Befehl **ST SET ATTRIBUTES**

In "Rich Text" Bereichen werden Stilattribute als HTML Tags vom Typ `` gespeichert. 4D interpretiert diese Tags beim Anzeigen des Textbereichs, d.h. der Entwickler kann Stilattribute in einem Text per Programmierung festlegen und verändern. Die von 4D unterstützten `` Attribute werden im folgenden beschrieben. Die 4D Funktion **ST Get plain text** findet Rohtext (raw text) ohne Stil Tags.

Die Beschreibung der Befehle zum Verwalten von Rich Text Bereichen und die unterstützten `` Tags finden Sie im Kapitel **Mehrfachstil Text** des Handbuchs *4D Programmiersprache*.

Eigenschaften zum Verwalten von Rich Text

Eigenschaften zum Verwalten von Rich Text sind verfügbar für eingebare Variablen, Felder und Zellen von Listboxen vom Typ alphanumerisch oder Text.

Mehrfachstil

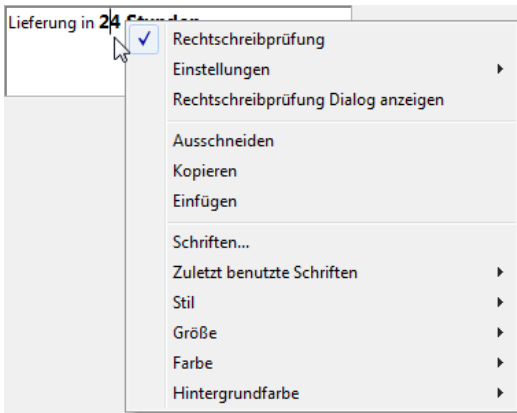
Die Option **Mehrfachstil** unter der Gruppe "Text" der Eigenschaftenliste aktiviert die Verwendung spezifischer Stilelemente im ausgewählten Bereich. Ist diese Option markiert, interpretiert 4D alle im Bereich gefundenen HTML Tags vom Typ ``.

Diese Option ist standardmäßig nicht markiert.

Kontextmenü

Diese Option erscheint nur unter der Gruppe "Eingabe", wenn die Option **Mehrfachstil** markiert ist.

Die Option **Kontextmenü** aktiviert für den Benutzer die Möglichkeit, während der Dateneingabe durch rechten Mausklick in den Bereich ein PopUp-Menü aufzurufen.



Dieses Menü bietet folgende Befehle:

- Standardbefehle zur Textbearbeitung (Ausschneiden, Kopieren, Einsetzen)
- **Schriften** zeigt das Dialogfenster mit den Systemschriften
- **Zuletzt benutzte Schriften** zeigt die Namen der zuletzt benutzten Schriften während der Sitzung. Die Liste kann bis zu 10 Schriften speichern. Darüberhinaus ersetzt die letzte Schrift die älteste. Diese Liste ist standardmäßig leer und die Option wird nicht angezeigt. Sie können diese Liste über die Befehle **SET RECENT FONTS** und **FONT LIST** verwalten
- Befehle der unterstützten Stiländerungen: Stil, Größe, Farbe und Hintergrundfarbe

Ändert der Benutzer über dieses PopUp-Menü ein Stilattribut, generiert 4D das Formularereignis On After Edit.

Hinweise:

- Sie können Stilelemente auch über den 4D Befehl **ST SET ATTRIBUTES** verändern. Beachten Sie, dass in diesem Fall kein Formularereignis erzeugt wird
- Der Stil "durchgestrichen" wird auf Mac OS nicht unterstützt. Sie können jedoch den entsprechenden Tag per Programmierung verwenden

Speichern mit Standard Stil Tags

Diese Option erscheint nur, wenn in der Gruppe "Text" die Option **Mehrfachstil** markiert wurde. Ist diese Option markiert, speichert der Bereich die Stil Tags mit dem Text, auch wenn nichts verändert wurde. In diesem Fall entsprechen die Tags dem Standardstil. Ist diese Option nicht markiert, werden nur geänderte Stil Tags gespeichert. Nachfolgend sehen Sie einen Text mit einer Stilveränderung:



Ist die Option "Mit Standard-Stil Tags speichern" nicht markiert, speichert der Bereich nur die Änderung. Der gespeicherte Inhalt ist dann:

```
What a <SPAN STYLE="font-size:13.5pt">beautiful</SPAN> day!
```

Ist die Option markiert, speichert der Bereich alle Formatierungsinformationen. Der erste generische Tag beschreibt den Standardstil, alle Variationen dazu erscheinen dann in eingebundenen Tags. Der gespeicherte Inhalt im Bereich ist dann:

```
<SPAN STYLE="font-family:'Arial';font-size:9pt;text-align:left;font-weight:normal;font-style:normal;text-decoration:none;color:#000000;background-color:#FFFFFF">What a <SPAN STYLE="font-size:13.5pt">beautiful</SPAN> day!</SPAN>
```

Standardaktionen

Bereiche mit Mehrfachstil lassen sich über folgende Standardaktionen verwalten.

- **Aktionen Bearbeiten**, wie Copy/Paste
- Aktionen für **Schrift**, wie z.B. fontBold oder fontSize
- Aktionen für **Dynamische Ausdrücke** zum Verwalten eingefügter Ausdrücke
- Aktionen für **Rechtschreibprüfung**

Sie können für Menübefehle oder Schaltflächen verwendet oder über den Befehl **INVOKE ACTION** ausgeführt werden. Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

Rich Text bearbeiten

Kopieren-Einsetzen und Drag-and-Drop

Die unterstützten Stilattribute (Schrifttyp, Schriftgröße, Schriftstil und -farbe) werden bei Drag-and-Drop oder Kopieren/Einsetzen von Text mit Stilelementen beibehalten zwischen:

- verschiedenen "Rich Text" Bereichen in 4D (Textvariablen/-felder und Listboxen)

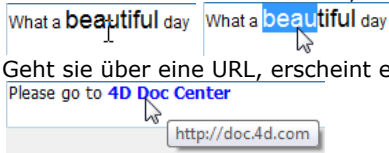
- Einem 4D Write Pro (oder 4D Write) Bereich und einem 4D "Rich Text" Bereich
- Einem externen Text mit Stilelementen und einem 4D "Rich Text" Bereich

In anderen Fällen werden die Stilelemente je nach Kontext beibehalten.

Mouseover

Es gibt zwei automatische Funktionen, wenn die Maus über einen formatierten Bereich geht:

- Geht sie über eine Textauswahl, wird der Cursor zu einem Pfeil:
- Geht sie über eine URL, erscheint ein Hilfetipp mit dem dazugehörigen Link:



Hinweis: Sie können URL Links mit dem Befehl **ST INSERT URL** einfügen.

URLs finden

URLs (Strings, die mit http://, https:// oder ftp:// beginnen) innerhalb von Feldern oder Variablen mit Mehrfachstil vom Typ Text oder Alphanumerisch werden automatisch gefunden. Klickt ein Benutzer unter Windows mit gedrückter **strg-Taste**, auf macOS mit gedrückter **Befehlstaste** in den Bereich, wird die URL direkt im standardmäßigen Web Browser ausgeführt. Unter Windows erscheinen gefundene URLs in blau und unterstrichen:



Mehrzeilige Textbereiche

Über spezifische Eigenschaften unter der Gruppe "Eingabe" lassen sich zwei Einstellungen zum Anzeigen und Drucken von Textbereichen (Felder und Variablen) in Formularen steuern:

- Anzeige von Wörtern am Zeilenende bei einzeiligen Bereichen
- Automatisches Einfügen von Zeilenumbruch in Textbereichen

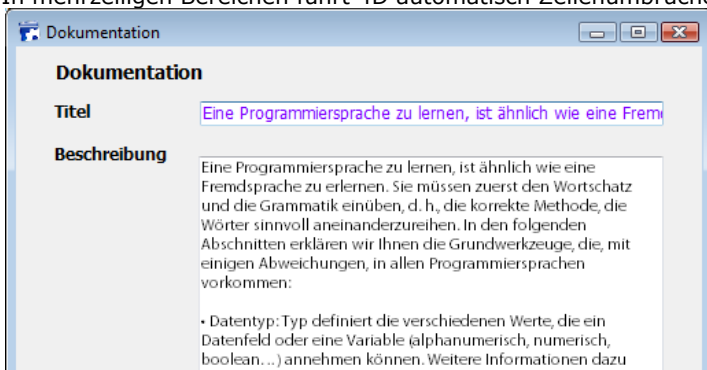
Hinweis: Der Zeilenumbruch lässt sich in Objekten vom Typ Alpha nicht verwenden.

Mehrzeilig

Diese Option ist für Variablen und Felder vom Typ alphanumerisch und Text verfügbar, sowohl eingebbar als auch nicht-eingebbar. Es gibt drei verschiedene Werte: **Ja**, **Nein**, **Automatik**.

Mehrzeilig = Automatik

In einzeiligen Bereichen werden Wörter am Zeilenende abgeschnitten und es gibt keinen Zeilenumbruch. In mehrzeiligen Bereichen führt 4D automatisch Zeilenumbrüche aus:



Mehrzeilig = Nein

In einzeiligen Bereichen werden Wörter am Zeilenende abgeschnitten und es gibt keinen Zeilenumbruch. Es gibt nie Zeilenumbrüche: Der Text wird immer in einer einzigen Zeile angezeigt. Enthalten Felder bzw. Variablen vom Typ alphanumerisch oder Text Zeilenumbrüche, wird der Text nach der ersten Zeilenschaltung entfernt, sobald der Bereich geändert wird:



Mehrzeilig = Ja

Wählen Sie diesen Wert, erscheint als zusätzliche Option **Zeilenumbruch**. Sie müssen auch hier einen Wert setzen.

Zeilenumbruch

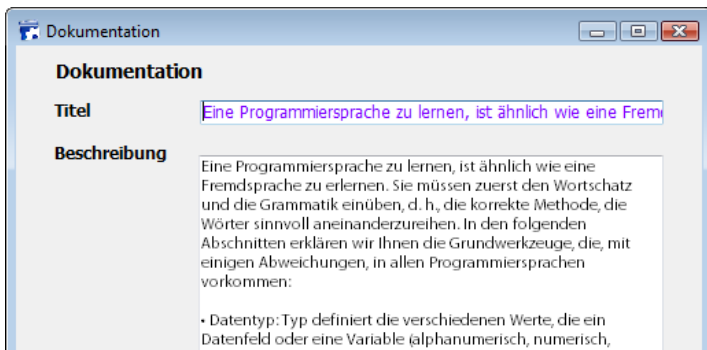
Diese Eigenschaft ist nur verfügbar, wenn die Option **Mehrzeilig** auf **Ja** gesetzt ist. Es gibt drei Werte: **Ja**, **Nein**, **Automatik**.

Hinweis: Damit diese Eigenschaften korrekt berücksichtigt werden, dürfen Textobjekte keine **Rollbalken** haben.

Zeilenumbruch = Automatik

In einzeiligen Bereichen wird der Text bis zur ersten Zeilenschaltung angezeigt oder bis zum letzten Wort, das vollständig angezeigt werden kann. 4D fügt Zeilenumbrüche ein; durch Drücken der Taste mit Pfeil nach unten können Sie im Inhalt des Bereichs scrollen.

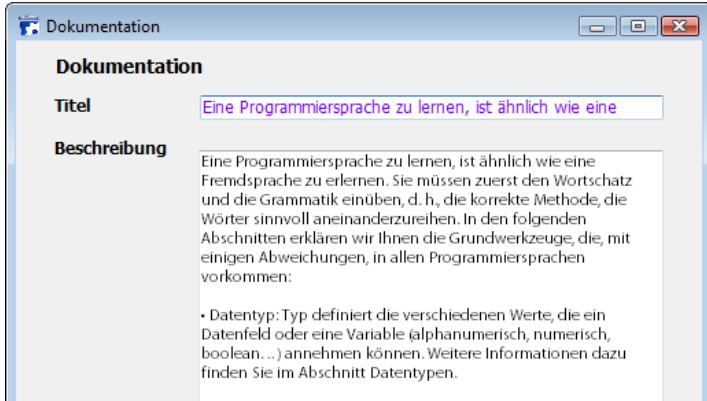
In mehrzeiligen Bereichen führt 4D automatisch Zeilenumbrüche aus:



Zeilenumbruch = Ja

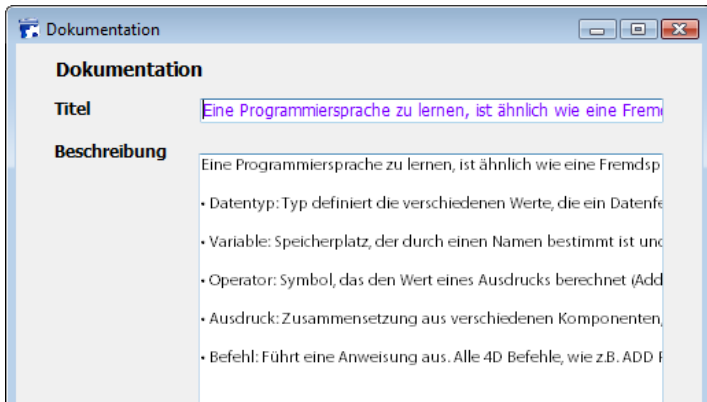
In einzeiligen Bereichen wird nur das letzte ganz anzeigbare Wort angezeigt. 4D fügt Zeilenumbrüche ein; durch Drücken der Taste mit Pfeil nach unten können Sie im Inhalt des Bereichs scrollen.

In mehrzeiligen Bereichen führt 4D automatisch Zeilenumbrüche aus:



Zeilenumbruch = Nein

4D führt keine automatischen Zeilenumbrüche aus, das als letztes anzeigbare Wort kann abgeschnitten sein. In Bereichen vom Typ Text werden Zeilenumbrüche unterstützt:



Rollbalken

Sie können Rollbalken verschiedenen Arten von Objekten zuweisen: **Get list item font** vom Typ Text oder Bild, hierarchische Listen, Listboxen und Unterformulare. Sie können diese Eigenschaft über die Eigenschaftenliste oder per Programmierung mit dem Befehl **OBJECT SET SCROLLBAR** verwalten.

Um Rollbalken hinzuzufügen, markieren Sie in der Eigenschaftenliste unter dem Thema „Darstellung“ die Option **Horizontale** bzw. **Vertikale Bildlaufleiste**.

- Für Objekte vom Typ Bild, hierarchische Liste und Listbox werden diese Eigenschaften über ein DropDown-Menü gesetzt, das auch den automatischen Modus enthält.
- Für Objekte vom Typ Text und Unterformular werden diese Eigenschaften über Kontrollkästchen gesetzt.

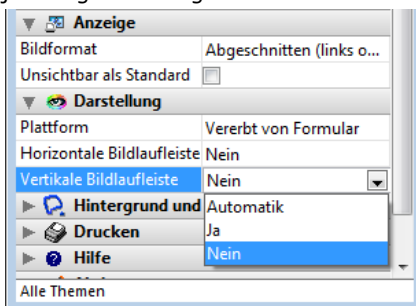
Hinweis: Hat ein Textdatenfeld oder eingebautes Objekt keinen Rollbalken, kann der Benutzer mit den Pfeiltasten scrollen.

Rollbalken für Bilder

Objekte vom Typ Bild, hierarchische Liste und Listbox unterstützen Rollbalken im automatischen Modus.

Hinweis: Objekte vom Typ Bild können Rollbalken haben, wenn für das Anzeigeformat „Abgeschnitten (nicht zentriert)“ definiert wurde. Weitere Informationen dazu finden Sie im Absatz "Bildformate" unter **GET LIST ITEM PARAMETER**.

Dann sind in der Eigenschaftenliste die Optionen **horizontale Bildlaufleiste** und **vertikale Bildlaufleiste** aktiv. Für jede Eigenschaft gibt es drei Auswahlmöglichkeiten:



- **Ja:** Der Rollbalken ist immer sichtbar, auch wenn es nicht erforderlich ist, d.h. wenn die Größe des Objekts kleiner als der Rahmen ist.
- **Nein:** Der Rollbalken ist nie sichtbar.
- **Automatisch:** Der Rollbalken erscheint bei Bedarf automatisch, d.h. wenn der Inhalt des Objekts größer als der Rahmen ist.

Tasten zum Scrollen

Tasten zum Scrollen sind für Benutzer bei Objekten vom Typ Bild mit Rollbalken automatisch verfügbar. Es gibt folgende Tastenkombinationen:

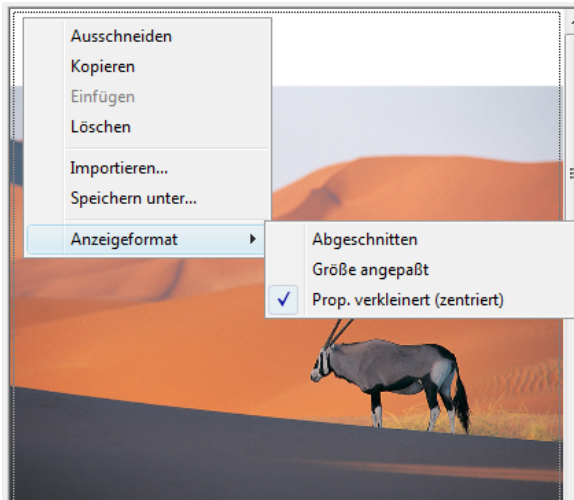
Taste(n)	Aktion
Pfeiltasten	Scrollen in Pfeilrichtung (= auf Cursor des Rollbalken klicken)
Alt + Pfeiltasten	Pixel-weises Scrollen in Pfeilrichtung
Bild Up / Bild Down	Seitenweise vertikal Scrollen (in Bezug auf Höhe des Bereichs)
Shift + Bild Up / Bild Down	Seitenweise horizontal Scrollen (in Bezug auf Breite des Bereichs)
Pos1	Oberen Bildrand anzeigen
Ende	Unteren Bildrand anzeigen
Shift + Home	Linken Bildrand zeigen
Shift + Ende	Rechten Bildrand zeigen
Mausrädchen	Vertikales Scrollen
Shift + Mausrädchen	Horizontales Scrollen

Beachten Sie, dass Scrollen im Bild das Formularereignis On Picture Scroll auslöst, siehe unter **Form event**.

Kontextmenü (Bilder)

Sie können Datenfeldern vom Typ Bild/Variable ein automatisches Kontextmenü zuordnen. Markieren Sie dazu einfach in der Eigenschaftsliste unter dem Thema "Eingabe" die Option **Kontextmenü**.

Ist diese Option markiert, hat der Bereich ein Kontextmenü, wenn das Formular in der Design- oder Anwendungsumgebung ausgeführt wird. Der Benutzer kann durch rechten Mausklick auf Befehle zum Bearbeiten und Anzeigen zugreifen:



Dieses Menü enthält zusätzlich zu den Standardbefehlen Ausschneiden, Kopieren, Einfügen und Löschen noch zwei weitere Befehle: Mit Importieren können Sie ein Bild aus einer Datei importieren, mit Speichern unter auf der Festplatte speichern. Diese beiden Befehle sind hilfreich zu nativen Bildverwaltung. Sie können damit Bilder in jedem von 4D unterstützten native Format öffnen und sichern.

Über dieses Menü können Sie auch das Anzeigeformat des Bildes verändern. Es gibt die Formate **Abgeschnitten**, **Größe angepaßt** und **Prop. verkleinert (zentriert)**. Die Änderung der Anzeige ist temporär, sie wird nicht mit dem Datensatz gesichert. Weitere Informationen dazu finden Sie im Abschnitt **GET LIST ITEM PARAMETER**.

Sind Bildfeld oder -variable nicht eingebbar, sind nur die Befehle **Kopieren**, **Sichern unter...** und die Formatierungsbefehle verfügbar.

Hinweis: Sie können die Option **Kontextmenü** auch für Felder oder Variablen vom Typ Text setzen. Sie richtet sich aber nach der Eigenschaft **Mehrfachstil** (siehe **Text mit Stil (Rich Text)**).

Drucke variable Größe

Funktionsweise

Die Option **Drucke variable Größe** ist für folgende Objekte verfügbar:

- Felder und Variablen vom Typ Bild
- Felder und Variablen vom Typ Text
- 4D Write Pro Areas (weitere Informationen dazu siehe Abschnitt **4D Write Pro Bereich verwenden** des Handbuchs *4D Write Pro*).

Diese Option liegt in der Eigenschaftsliste unter der Gruppe "Drucken":



Sie ist auch über die Befehle **OBJECT SET PRINT VARIABLE FRAME** und **OBJECT GET PRINT VARIABLE FRAME** verfügbar.

Hinweis: Unterformulare haben eine ähnliche Option. Weitere Informationen dazu finden Sie unter **Drucken** im Abschnitt **Unterformulare als Liste**.

Diese Eigenschaft verwaltet den Druckmodus für Objekte, deren Größe je nach Inhalt in den Datensätzen variieren kann. Diese Objekte lassen sich entweder mit fester oder variabler Größe drucken. Ein Objekt mit fester Größe wird in den Ausmaßen gedruckt, wie es im Formular erstellt wurde. Ein Objekt mit variabler Größe wird beim Drucken so erweitert, dass der gesamte Inhalt des Objekts eingeschlossen ist.

Dabei wird beim Drucken nur die Höhe automatisch an den Inhalt des Objekts angepasst. Die Breite ist davon nicht betroffen, sie wird von Objekteigenschaften vorgegeben.

Sie können nicht mehrere Objekte mit variabler Größe beim Drucken nebeneinander in ein Formular setzen. Sie können Objekte mit fester Größe neben jede Seite dieses variablen Objekts legen. Es muss aber mindestens eine Zeile länger als das daneben gelegte Objekt sein und alle Objekte müssen oben ausgerichtet sein. Sonst wird der Inhalt der anderen Felder für jeden horizontalen Abschnitt des Objekts mit variabler Größe beim Drucken wiederholt.

Bei Ausgabeformularen können Sie nur im Detailbereich Objekte mit variabler Größe setzen.

Hinweis: Die Funktionen **Print object** und **Print form** unterstützen **nicht** diese Option.

Bilder

Bilder lassen sich mit fester oder variabler Größe drucken, sofern das Anzeigeformat dies zulässt. Das ist für folgende Formate möglich:

- Abgeschnitten (Zentriert)
- Auf Hintergrund
- Abgeschnitten (Nicht-zentriert)

Weitere Informationen dazu finden Sie im Abschnitt **Anzeigeformate**.

- Markieren Sie die Option **Drucke variable Größe**, wird das Bild mit einer an die Größe angepassten Höhe gedruckt. Sie wird automatisch beim Drucken erweitert, damit das ganze Bild erscheint.
- Markieren Sie diese Option nicht, wird das Bild mit der festen Höhe gedruckt, die im Formular vorgegeben ist.

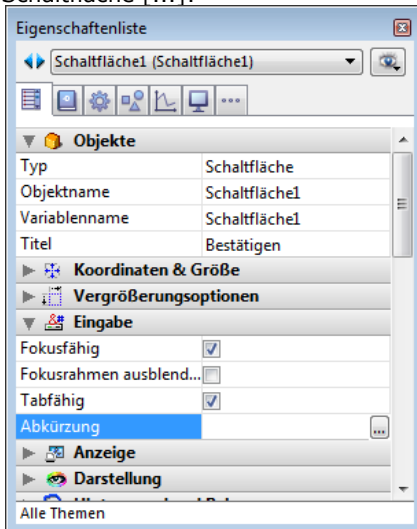
Text

- Markieren Sie die Option **Drucke variable Größe**, wird der Text mit einer an die Größe angepassten Höhe gedruckt. Das Textfeld wird beim Drucken automatisch erweitert, damit der ganze Textinhalt erscheint.
- Markieren Sie diese Option nicht, wird der Text mit der festen Höhe gedruckt, die im Formular vorgegeben ist.

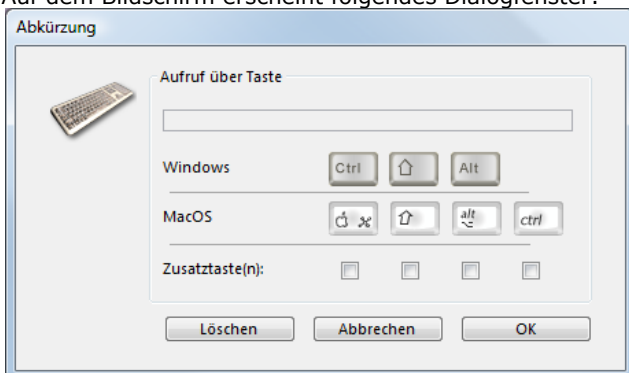
Tastaturkürzel

Sie können **Schaltflächen** und **Kontrollkästchen** zuweisen. Der Benutzer kann dann die Schaltfläche oder das Kontrollkästchen über die Tastatur statt mit der Maus aktivieren.

Gehen Sie dazu in der Eigenschaftensliste zur Gruppe „Eingabe“ und klicken neben der Option **Abkürzung** auf die Schaltfläche [...]:



Auf dem Bildschirm erscheint folgendes Dialogfenster:



Geben Sie das Tastaturkürzel ein. Wollen Sie zum Beispiel die Tastenkombination **Strg-Taste + h** verwenden, drücken Sie die **Strg-Taste** und dann die **h-Taste**. Der Buchstabe erscheint dann im Bereich „Aufruf über Taste“, das Kontrollkästchen unter der **Strg-Taste** bekommt ein Häkchen.

Sie müssen keine Sondertasten verwenden, d.h. Sie können auch jede Taste allein als Abkürzung verwenden. Verwenden Sie Tastenkürzel mit Bedacht, damit es nicht zu Interferenzen mit vorgegebenen Kombinationen, z.B. für Menübefehle, kommt.

Bei Bedarf können Sie die Auswahl der Sondertasten durch Aktivieren oder Deaktivieren der Kontrollkästchen ändern. Um neu zu beginnen, können Sie auf die Schaltfläche **Löschen** klicken. Wenn Sie fertig sind, klicken Sie auf die Schaltfläche **OK**.

Die Eigenschaftensliste zeigt die Tastenkombination an, die dem Objekt zugewiesen wurde. Wollen Sie die Abkürzung später wieder ändern, öffnen Sie einfach das Dialogfenster *Abkürzung* und geben eine andere Tastenkombination ein.

Hinweis: Sie können auch einen eigenen Menübefehl mit Tastenkürzel versehen. Bei einem Konflikt zwischen zwei Tastenkürzeln hat das aktive Objekt Vorrang. Weitere Informationen dazu finden Sie im Abschnitt **Menüeigenschaften einrichten**.

Geometrie des Fensters beibehalten

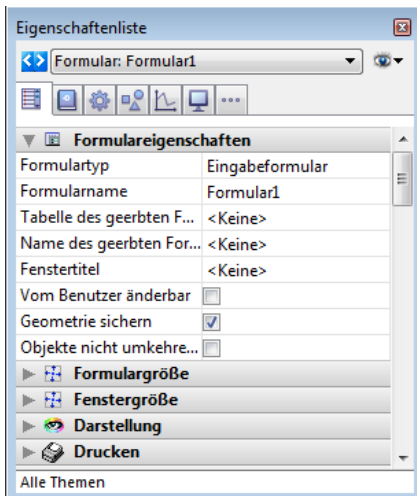
4D enthält eine automatische Funktionalität, die die spezifische Darstellung von Fenstern, d.h. ihre Geometrie, beim Schließen mitsichert. Auf diese Weise finden Benutzer ihre Arbeitsumgebung so vor, wie sie diese verlassen haben. Dieser Automatismus betrifft die Koordinaten des Fensters und die Position der darin enthaltenen Objekte, sowie den aktuellen Status bestimmter Objekte, wie z.B. **Registerkarten**.

Hinweise:

- Die automatischen Funktionalitäten werden nur unterstützt, wenn Formulare mit derselben Größe wieder geöffnet werden, in der sie geschlossen wurden. Sie beruhen folglich in erster Linie auf der Funktion **Open form window** mit dem Parameter *.
- Bitte beachten Sie beim Sichern der Formulargeometrie: Verwenden Sie vererbte Formulare, müssen die Objektnamen auch zwischen dem Hauptformular und dem vererbten Formular einmalig sein.

Geometrie des Formulars sichern

Die Option **Geometrie sichern** liegt in der Eigenschaftensliste des Formulars unter der Gruppe **Formulareigenschaften**:



Ist diese Option markiert, sichert 4D beim Schließen des Fensters automatisch verschiedene Parameter des Formulars, unabhängig, wie sie während der Arbeitssitzung verändert wurden:

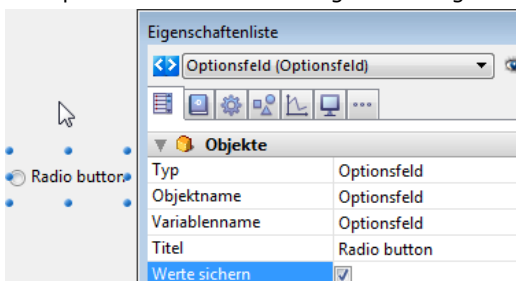
- Die aktuelle Seite
- Position, Größe und Sichtbarkeit jedes Formularobjekts (inkl. Größe und Sichtbarkeit der Spalten von Listboxen)

Hinweis: Diese Option berücksichtigt keine Objekte, die über den Befehl **OBJECT DUPLICATE** erstellt wurden. Damit ein Benutzer die Umgebung solcher Objekte wiederherstellen kann, muss der Entwickler die Abfolge Erstellen, Definition und Positionierung dieser Objekte wiederholen.

Ist diese Option markiert, ist für einige Objekte auch die Option **Wert sichern** verfügbar (siehe nächster Abschnitt).

Option Werte sichern

Die Option **Werte sichern** liegt in der Eigenschaftensliste unter der Gruppe "Objekte":



Sie ist verfügbar, wenn:

- Die Option **Geometrie des Formulars sichern** für das Formular markiert ist
- Das ausgewählte Objekt zur generellen Geometrie des Formulars beiträgt.
Die Option ist z.B. für Optionsfelder verfügbar, da ihr Wert zum Ein- oder Ausblenden zusätzlicher Bereiche im Fenster dient (siehe nachfolgende Beispiele).

Hier die Liste der Objekte, deren Wert sich sichern lässt:

Objekt	Gesicherter Wert
Optionsfeld	Wert der zugeordneten Variable (1, 0, Wahr oder Falsch für Schaltflächen je nach Typ)
3D Optionsfeld	Wert der zugeordneten Variable (0, 1)
Kontrollkästchen	Wert der zugeordneten Variable (0, 1, 2)
3D Kontrollkästchen	Wert der zugeordneten Variable (0, 1)
Registerkarte	Nummer der gewählten Registerkarte
PopUp/DropDown Listet	Nummer der gewählten Zeile
Bildmenü	Nummer der gewählten Zeile

Information speichern und verwenden

4D behält beim Schließen die Koordinaten des Fensters und unter Windows ihren maximierten Status bei, wenn das Fenster mit der Anweisung **Open form window (formName; *)** generiert wurde.

Diese Information, sowie optionale Sicherungsoptionen (Geometrie und Wert) werden jetzt im json Format beim Schließen des Fensters im aktuellen Benutzerordner des Rechners gesichert. So kann jeder Benutzer, der sich mit seinem eigenen Rechner anmeldet, seine eigene Umgebung beibehalten, selbst wenn das Account "Standardbenutzer" verwendet wird.

Diese Information wird nur verwendet, wenn das Formular mit denselben Dimensionen und nach Schließen wieder geöffnet wird. Das ist der Fall, wenn entweder die Anweisung **Open form window(*)** verwendet wurde, oder der Entwickler ein eigenes System zum Sichern der Koordination eingerichtet hat.

Die gesicherte Information wird in folgender Reihenfolge wiederhergestellt und angewandt:

- Größe und Position des Fensters werden wiederhergestellt, wenn die Funktion **Open form window** ausgeführt wird
- Werte von Variablen werden wiederhergestellt, wenn das Formular vor Aufrufen des Ereignisses On Load geladen wird.
- Die aktuelle Seite wird vor Aufrufen des Ereignisses On Load geladen
- Position, Größe und Sichtbarkeit jedes Objekts werden direkt nach dem Ereignis On Load wiederhergestellt.

Die Eigenschaften von Objekten in Unterformularen werden auf dieselbe Weise gesichert und automatisch wieder angewandt.

Warnung: Über die Optionen "Geometrie sichern" und "Werte sichern" gespeicherte Angaben werden zurückgesetzt, wenn die Objekte eines Formulars im Designmodus geändert werden (Größe ändern, Objekt bewegen, hinzufügen, löschen oder umbenennen). Deshalb raten wir dringend, diese Oberflächenfunktionalität **NICHT** zum Sichern dauerhafter Werte wie Benutzereinstellungen zu verwenden.

Beispiele

Trenner speichern

Die relative Position der Trenner im Formular soll gesichert werden. Dafür müssen Sie nur die Option Geometrie sichern markieren. Beim ersten Öffnen sieht das Formular so aus:



Der Benutzer vergrößert das Fenster und bewegt die Trenner. Die Objekte werden gemäß ihren eigenen Eigenschaften angepasst. Dann schließt der Benutzer das Fenster. Wird das Formular wieder geöffnet, bewahren die Objekte ihr neues Aussehen:

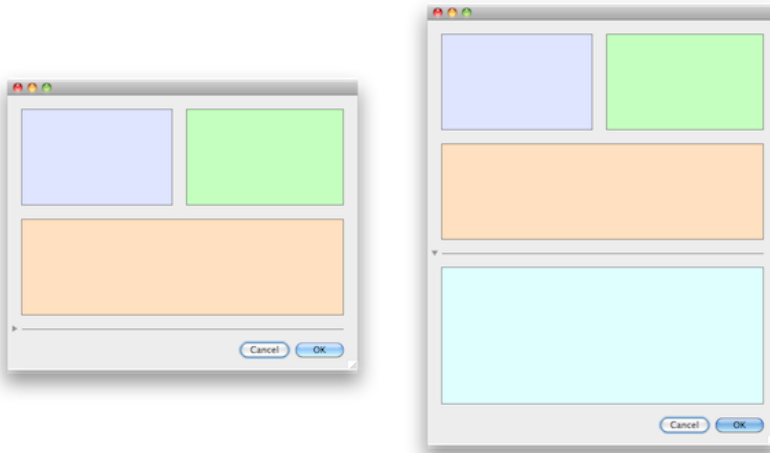


Aufklappbereiche speichern

Sie haben in einem Formular einen oder mehrere Aufklappbereiche gesetzt, die über 3D Optionsfelder verwaltet werden. Jedes Optionsfeld hat ein Dreieck, das nach rechts zeigt, wenn der Bereich zugeklappt ist und nach unten, wenn er aufgeklappt ist. Es gibt mehrere Wege, solche Bereiche einzurichten (Bewegen oder Sichtbarkeit von Objekten, über verschiedene Formularseiten, etc.). In allen Fällen kann auch die Größe des Fensters variieren.

Damit der Status der Aufklappbereiche zwischen zwei Sitzungen beibehalten wird, müssen Sie folgendes ausführen:

- Die Option **Geometrie sichern** für das Formular markieren, damit die aktuelle Seite, sowie Position und Status aufgeklappt der Objekte beibehalten wird.
- Die Option **Wert sichern** für das Objekt 3D Optionsfeld markieren, damit der Wert der zugewiesenen Variable beibehalten wird (0 oder 1 für Status auf- oder zugeklappt).



Registerkarten speichern










Sie haben in einem Formular Registerkarten mit der Standardaktion "Gehe zu Seite" gesetzt.



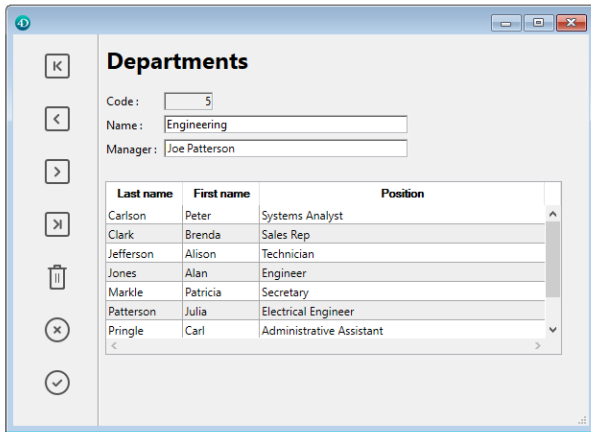
Damit der Speichermechanismus korrekt arbeitet, müssen Sie die Option **Geometrie sichern** für das Formular und **Wert sichern** für das Objekt Registerkarte markieren



Listboxen

-  Überblick
-  Eigenschaften für Listboxen
-  Eigenschaften für Spalten der Listbox
-  Eigenschaften für Spaltentitel der Listbox
-  Eigenschaften für Fußteile der Listbox
-  Hierarchische Listboxen erstellen und verwenden
-  Standardaktionen verwenden
-  Anzeige der Felder in Listboxen
-  Ergebnis einer SQL-Anfrage in Listbox anzeigen

Listboxen sind komplexe aktive Objekte, in denen sich Daten als synchronisierte Spalten anzeigen und eingeben lassen. Sie sind ähnlich wie "gruppierte rollbare Bereiche", bieten jedoch zusätzliche Möglichkeiten, wie z.B. Werteingabe, Spalten sortieren oder verschieben, angepasste Darstellung, etc. Ein Objekt vom Typ Listbox lässt sich vollständig im 4D Formulareditor einrichten oder über Programmierung verwalten.



Dieser Abschnitt beschreibt die Funktionalitäten zum Erstellen und Einrichten von Listboxen im Formulareditor. Weitere Informationen zur programmierten Verwaltung finden Sie im Kapitel **Listbox** im Handbuch *4D Programmiersprache*.

Typ Auswahl, Array und Collection oder Entity-Selection

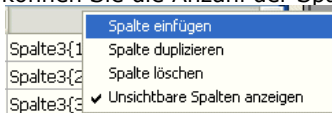
Es gibt mehrere Arten von Listboxen: **Listbox vom Typ Auswahl**, **Listbox vom Typ Array** und **Listbox vom Typ Collection oder Entity-Selection**.

- **Listbox vom Typ Auswahl:** Die Anzahl der Zeilen richtet sich nach der aktuellen bzw. der temporären Auswahl. Jede Spalte wird einem Datenfeld, z.B. [Angestellte]Nachname) oder einem 4D Ausdruck zugewiesen. Der Ausdruck kann auf einem oder mehreren Datenfeldern, z.B. [Angestellte]Vorname+" "[Angestellte]Nachname) oder auf einer Formel, z.B. String(Milliseconds) basieren. Auch eine Projektmethode, eine Variable oder ein Array-Eintrag sind möglich.
Basiert die Listbox auf der aktuellen Auswahl, wird jede Veränderung in der Datenbank automatisch in der Listbox wiedergegeben, und umgekehrt. Von daher ist die aktuelle Auswahl an beiden Stellen immer gleich.
- **Listbox vom Typ Array:** Die Anzahl der Zeilen richtet sich nach der Anzahl der Array-Elemente. Jede Spalte der Listbox ist mit einem 4D Array verbunden. 4D weist standardmäßig jeder Spaltenvariablen den Namen SpalteX zu, und so auch jedem verbundenen Array. Sie können den Namen in den Spalteneigenschaften verändern. Bei diesem Listentyp werden die eingegebenen oder angezeigten Werte über die 4D Programmiersprache gesteuert. Sie können einer Spalte auch eine Auswahlliste zuordnen, um die Dateneingaben zu steuern. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Spalten der Listbox**.
- **Listbox vom Typ Collection oder Entity-Selection:** Die Anzahl der Zeilen richtet sich nach der Anzahl der Collection-Elemente oder Entities. In den Spalteneigenschaften ist jeder Spalte ein 4D Ausdruck, der in der Regel auf einen Eigenschaftspfad verweist (siehe **Eigenschaften für Spalten der Listbox**). Collection-Elemente oder Entities werden mit dem Befehl **This** zurückgegeben (zum Beispiel *This.firstName*, um den Wert der Eigenschaft "name" für jedes Element/Entity anzuzeigen).
Hinweis: Collections und Entity-Selections sind als Datenquelle der Listbox sehr ähnlich. Der Unterschied ist, dass Listboxen vom Typ Entity-Selections zusätzliche Features bei der Eingabe von Benutzerdaten verwenden können (siehe **Eingabe verwalten**), da Zeilen direkt mit den Daten aus dem Datastore verbunden sind.

Sie können in einem Objekt Listbox nicht verschiedene Listboxtypen miteinander kombinieren.

Hauptmerkmale

Eine Listbox enthält eine oder mehrere Spalten, deren Inhalt automatisch aufeinander abgestimmt wird. Beim Erstellen enthält eine Listbox standardmäßig eine Spalte. Über das Kontextmenü der Spalte oder die Eigenschaften der Listbox können Sie die Anzahl der Spalten verändern, d.h. Spalten hinzufügen, kopieren oder entfernen.



Die Anzahl der Spalten ist theoretisch unbegrenzt. Die Begrenzung liegt allein in den Ressourcen des Rechners.

Eine Listbox besteht aus vier Teilen: das **Objekt** ansich, **Spalten**, **Spaltenkopfteile** und **Spaltenfußteile**. Diese Teile lassen sich im Formulareditor getrennt auswählen. Jeder Teil hat einen eigenen Objekt- und Variablennamen sowie spezifische Eigenschaften. So wird die Anzahl der Spalten, die wechselnde Farbe pro Spalte in den Objekteigenschaften der Listbox definiert, die Spaltenbreite in den Spalteneigenschaften, die Schriftart für den Kopfteil in den Eigenschaften des Kopfteils.

Hinweis: Bei Listboxen vom Typ Auswahl haben Spalten keinen Variablennamen.

Sie können eine Listbox vom Typ Array entweder im **Standardmodus** oder im **hierarchischen Modus** anzeigen. Listboxen im hierarchischen Modus verwenden spezifische Mechanismen, die unten beschrieben werden.

Mit Listboxen arbeiten

Während der Ausführung können Sie in Listboxen Daten als Liste anzeigen und eingeben. Um eine Zelle editierbar zu machen (wenn die Dateneingabe für die Spalte erlaubt ist), doppelklicken Sie auf den Wert:

Vorname	Nachname
Michael	Meier
Alexander	Baumann

Hinweis: Weitere Informationen dazu finden Sie im Abschnitt **Eingabe verwalten** im Handbuch 4D *Programmiersprache*.

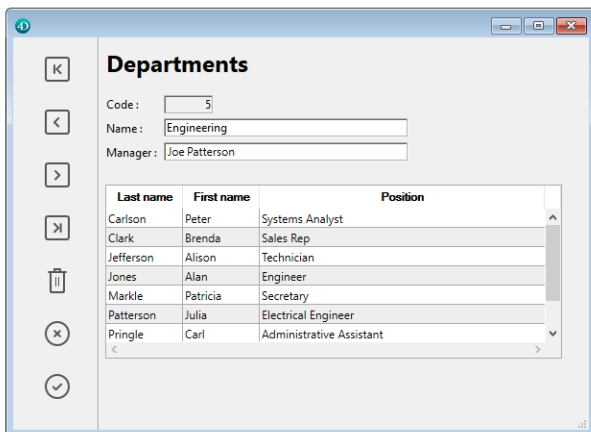
Sie können in einer Zelle der Listbox Text über mehrere Zeilen eingeben und anzeigen. Um eine Zeilenschaltung hinzuzufügen, wählen Sie:

- Unter Windows **Strg+Zeilenschaltung**
- Auf macOS **Wahltaste+Zeilenschaltung**

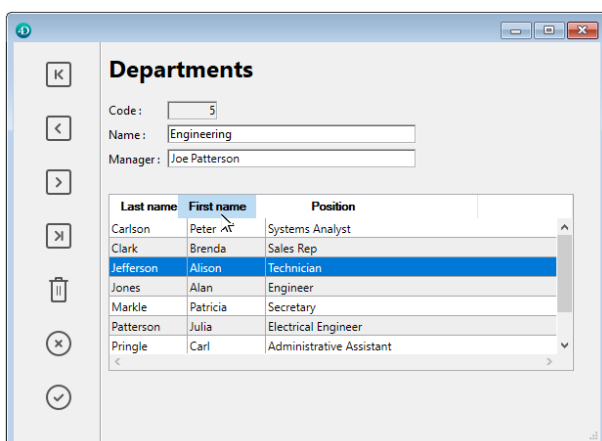
Beachten Sie, dass die Zeilenhöhe nicht automatisch angepasst wird. Das gilt nicht für Listboxen vom Typ Auswahl oder Collection, sondern nur für Listboxen vom Typ Array, wenn die Option **Automatische Zeilenhöhe** markiert ist. Um Spaltenwerte zu sortieren, klicken Sie auf den Spaltentitel. Die Sortierung ist alphanumerisch, je nach Klick entweder aufsteigend oder absteigend. Alle Spalten werden automatisch miteinander synchronisiert.

Hinweis: Weitere Informationen dazu finden Sie im Abschnitt **Sortieren verwalten** im Handbuch 4D *Programmiersprache*.

Sie können auch jede Spalte in der Größe verändern:



Der Benutzer kann die Reihenfolge der Spalten und für Listboxen vom Typ Array auch die Reihenfolge der Zeilen mit der Maus verschieben, wenn diese Aktion zugelassen ist:



Der Benutzer kann auch eine oder mehrere Zeilen über standardmäßige Tastenkombinationen wählen: **Umschalttaste+Klick** für eine fortlaufende Auswahl; **Strg-Taste+Klick** unter Windows, **Befehlstaste+Klick** auf macOS für eine unterbrochene Auswahl.

All diese Merkmale können Sie in den Eigenschaften der Listbox, der Spalte und des Kopfteils einstellen. Diese werden im folgenden beschrieben.

Hinweis: Die Beschreibung der spezifischen Merkmale für hierarchische Listboxen finden Sie unter **Hierarchische Listboxen erstellen und verwenden**.

Listboxen drucken

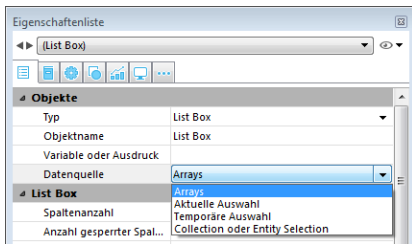
Listboxen in Formularen lassen sich im Modus "Vorschau" (Bereich Listbox als Bild drucken) oder im Modus "Erweitert" (dynamisches Drucken in variabler Größe) drucken. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Listboxen drucken**.

Eigenschaften für Listboxen

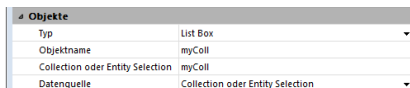
Wählen Sie ein Objekt vom Typ Listbox im Formulareditor, zeigt die Eigenschaftensliste mehrere spezifische Gruppen und Eigenschaften. Weitere Eigenschaften sind verfügbar, wenn Sie in der Listbox eine Spalte (siehe [Eigenschaften für Spalten der Listbox](#)), einen Spaltentitel (siehe [Eigenschaften für Spaltentitel der Listbox](#)) oder einen Spaltenfußteil ([Eigenschaften für Fußteile der Listbox](#)) auswählen.

Gruppe Objekte

Diese Gruppe enthält die Eigenschaft **Datenquelle**, über die Sie die Art der Listbox festlegen können:



- Wählen Sie die Option **Arrays**, um Array Elemente als Zeilen der Listbox zu verwenden. Diese Option ist erforderlich, um das Ergebnis einer SQL-Anfrage in der Listbox einzufügen. Weitere Informationen dazu finden Sie im Abschnitt [Ergebnis einer SQL-Anfrage in Listbox anzeigen](#).
- Wählen Sie die Option **Aktuelle Auswahl**, um Ausdrücke, Datenfelder oder Methoden zu verwenden, deren Werte für jeden Datensatz der aktuellen Auswahl einer Tabelle gewertet werden.
- Wählen Sie die Option **Temporäre Auswahl**, um Ausdrücke, Datenfelder oder Methoden zu verwenden, deren Werte für jeden Datensatz der temporären Auswahl einer Tabelle gewertet werden.
- Wählen Sie die Option **Collection oder Entity-Selection**, um Collection-Elemente oder Entities zum Definieren von Zeileninhalt der Listbox zu verwenden. Wählen Sie diese Option als Datenquelle, wird die Eigenschaft *Variable oder Ausdruck* ersetzt durch die Eigenschaft *Collection oder Entity-Selection*. Hier geben Sie einen Ausdruck ein, der eine Collection oder eine Entity-Selection zurückgibt. Das ist in der Regel der Name einer Variablen, eines Collection-Elements oder einer Eigenschaft, die eine Collection oder eine Entity-Selection enthält. Namen der Collection Variable ein, die Sie mit der Listbox verwenden wollen:



Die Collection oder Entity-Selection muss beim Laden des Formulars verfügbar sein. Jedes Element der Collection oder Entity der Entity-Selection wird einer Zeile der Listbox zugeordnet und durch den Befehl **This** als Objekt verfügbar gemacht:

- Verwenden Sie eine Collection von Objekten, rufen Sie **This** im Ausdruck der Datenquelle auf, um auf jeden Eigenschaftswert zuzugreifen, z.B. **This.<propertyPath>**
- Verwenden Sie eine Collection von skalaren Werten, rufen Sie **This.value** im Ausdruck der Datenquelle auf, um auf jeden Wert zuzugreifen
- Verwenden Sie eine Entity-Selection, rufen Sie **This** im Ausdruck der Datenquelle auf, um auf jeden Attributswert zuzugreifen, z.B. **This.<attributePath>**

Weitere Informationen zu Entity-Selections finden Sie im Kapitel [ORDA](#).

Gruppe Datenquelle

Dieses Thema erscheint nur für Listboxen vom Typ Auswahl und Collection/Entity-Collection.

Listboxen vom Typ Auswahl

Es enthält die Eigenschaft **Haupttabelle** für Listboxen, die auf der aktuellen Auswahl basieren und die Eigenschaft **Temporäre Auswahl** für Listboxen, die auf einer temporären Auswahl basieren.

- **Haupttabelle:** Damit definieren Sie die Tabelle, deren aktuelle Auswahl Sie benutzen wollen. Diese Tabelle und die dazugehörige aktuelle Auswahl sind die Referenz für die Datenfelder, die den Spalten in der Listbox zugeordnet werden (Feldreferenzen oder Ausdrücke mit Feldern). Selbst wenn einige Spalten Felder aus anderen Tabellen enthalten, wird die Anzahl der angezeigten Zeilen durch die Haupttabelle bestimmt. Das PopUp-Menü für diese Eigenschaft zeigt alle Tabellen der Datenbank an, egal ob das Formular mit einer Tabelle (Tabellenformular) oder nicht (Projektformular) verknüpft ist. Die Eigenschaft zeigt standardmäßig die erste Tabelle der Datenbank. Weitere Informationen dazu finden Sie im Abschnitt [Anzeige der Felder in Listboxen](#).
- **Temporäre Auswahl:** Hier geben Sie den Namen einer gültigen temporären Auswahl ein. Sie kann vom Typ Prozess oder Interprozess sein. Der Inhalt der Listbox basiert dann auf dieser Auswahl. Die gewählte temporäre Auswahl muss vorhanden und gültig sein, wenn die Listbox angezeigt wird. Andernfalls bleibt die Listbox leer. Sie

bleibt ebenfalls leer, wenn Sie den Namensbereich leer lassen.

Hinweis: Temporäre Auswahlen sind geordnete Datensatzlisten. Sie dienen dazu, die Reihenfolge und den aktuellen Datensatz einer Auswahl im Speicher zu halten. Weitere Informationen dazu finden Sie im Kapitel **Temporäre Auswahl** des Handbuchs **4D Programmiersprache**.

Listboxen vom Typ Collection oder Entity-Selection

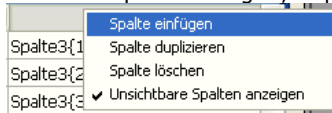
Für Listboxen vom Typ Collection oder Entity-Selection gibt es drei Eigenschaften.

- **Aktueller Eintrag:** Gibt eine Variable oder Ausdruck an, die dem vom Benutzer ausgewählten Collection-Element bzw. der Entity zugewiesen wird. Sie müssen eine Objektvariable oder einen zuweisbaren Ausdruck verwenden, der Objekte akzeptiert. Wenn der Benutzer nichts auswählt, gibt die Variable **Null** zurück.
- **Aktuelle Position des Eintrags:** Gibt eine Variable oder Ausdruck an, der eine Lange Ganzzahl zugewiesen wird, die die Position des vom Benutzer ausgewählten Collection-Element bzw. der Entity angibt.
 - Wenn kein Element bzw. Entity ausgewählt ist, erhält die Variable oder der Ausdruck Null.
 - Wenn ein einzelnes Element bzw. Entity ausgewählt ist, erhält die Variable oder der Ausdruck seine Position.
 - Wenn mehrere Elemente bzw. Entities ausgewählt sind, erhält die Variable die Position des zuletzt ausgewählten Elements bzw. Entity.
- **Ausgewählte Einträge:** Gibt eine Variable oder Ausdruck an, die den Elementen oder Entities zugewiesen wird, die der Benutzer ausgewählt hat.
 - Für eine Listbox vom Typ Collection müssen Sie eine Collection-Variable oder einen zuweisbaren Ausdruck verwenden, der Collections akzeptiert.
 - Für eine Listbox vom Typ Entity-Selection wird ein Objekt Entity-Selection erstellt. Sie müssen eine Objekt-Variable oder einen zuweisbaren Ausdruck verwenden, der Objekte akzeptiert.

Gruppe Listbox

Diese Gruppe enthält alle Grundeigenschaften des Objekts:

- **Spaltenanzahl:** Anzahl der Spalten in der Listbox (standardmäßig 1). Klicken Sie auf den Spaltentitel, erscheint das Kontextmenü, über das Sie die Anzahl ändern können. Sie klicken auf eine Spalte bzw. einen Spaltentitel und wählen zwischen Spalte einfügen, duplizieren oder löschen:



- **Anzahl gesperrte Spalten:** Anzahl, die dauerhaft am linken Rand der Listbox sichtbar bleiben soll, auch wenn der Benutzer horizontal durch die Spalten scrollt. Eine gesperrte Spalte lässt sich in der Größe anpassen, ist eingebbar, etc., genauso wie reguläre Spalten. Gesperrt ist lediglich ihre Position in der Listbox, d.h. sie bleibt beim Scrollen erhalten. Das ist z.B. hilfreich, um Zeilenbezeichnungen in umfangreichen Arrays anzuzeigen.

Mit dieser Eigenschaft können Sie einen gesperrten Bereich setzen. Wird eine gesperrte Spalte per Programmierung gelöscht, verringert sich die Anzahl der gesperrten Spalten um 1. Wird eine Spalte per Programmierung in den gesperrten Bereich eingefügt, wird sie automatisch gesperrt.

Hinweis: *Einen Vergleich zwischen gesperrten und statischen Spalten finden Sie im unteren Absatz "Gesperrte und statische Spalten".*

- **Anzahl statische Spalten:** Anzahl, die sich nicht während der Ausführung bewegen lässt. Der Wert gibt die Anzahl der festen Spalten, beginnend mit der ersten Spalte der Listbox an. Um das Bewegen von Spalten auszuschließen, muss dieser Wert gleich der Gesamtanzahl der Spalten sein. Beachten Sie, dass auch ausgeblendete Spalten mitgezählt werden.

Ist die Option **hierarchische Listbox** nicht aktiviert, hat diese Eigenschaft standardmäßig den Wert 0 (Null), d.h. alle Spalten lassen sich bewegen.

Ist der hierarchische Modus aktiviert (nur Listboxen vom Typ Array), hat diese Eigenschaft immer mindestens den Wert 1.

Hinweis: *Einen Vergleich zwischen gesperrten und statischen Spalten finden Sie im unteren Absatz "Gesperrte und statische Spalten".*

- **Auswahlmenge:** Diese Eigenschaft wurde nur für Listboxen vom Typ *Auswahl* hinzugefügt. Sie definiert die Menge zum Verwalten markierter Datensätze in der Listbox. Ist die Datenquelle **Array** ausgewählt, wird ein Array vom Typ Boolean mit demselben Namen wie die verwendete Listbox verwendet.

4D erstellt eine Standardmenge mit Namen *ListboxMengeN*, wobei N bei 0 startet und gemäß der Anzahl Listboxen im Formular erhöht wird. Bei Bedarf können Sie diese Standardmenge ändern, es kann eine lokale, Prozess- oder Interprozessmenge sein. Wir empfehlen die Verwendung einer lokalen Menge, z.B. *\$LBMenge*, um den Netzwerkverkehr gering zu halten. Sie wird dann automatisch von 4D verwaltet. Wählt der Benutzer eine oder mehrere Zeilen in der Listbox, wird die Menge sofort aktualisiert. Um eine oder mehrere Zeilen per Programmierung auszuwählen, verwenden Sie die Befehle im Kapitel **Mengen** des Handbuchs *4D Programmiersprache*.

Hinweise:

- Der Status **Markiert** in Zeilen der Listbox ist vollkommen unabhängig vom Status **Markiert** in Datensätzen einer Tabelle.

- Enthält die Eigenschaft **Auswahlmenge** keinen Namen, ist in der Listbox keine Auswahl möglich.

- **Zeilenkontrolle Array** (nur Listboxen vom Typ Array): Damit steuern Sie die Eigenschaften "ausgeblendet", "deaktiviert" und "auswählbar" für jede Zeile der Listbox. Das Array vom Typ Lange Ganzzahl muss die gleiche Anzahl Elemente haben wie die Listbox. Jedes Element des Array gibt an, ob die entsprechende Zeile:

- ein- oder ausgeblendet ist (sichtbar ist Standard)
- aktiviert oder deaktiviert ist (aktiviert ist Standard)
- auswählbar oder nicht auswählbar ist (auswählbar ist Standard)

Sie können die Eigenschaften für jede Zeile setzen, indem Sie dem entsprechenden Element des Array Konstanten zuweisen. Weitere Informationen dazu finden Sie im Abschnitt **Zeilenanzeige verwalten**.

Hinweis zur Kompatibilität: *Diese Eigenschaft ersetzt "Ausgeblendete Zeilen Array", die in bisherigen 4D Releases verfügbar war. Zur Wahrung der Kompatibilität wird weiterhin ein Boolean Array als Zeilenkontrolle unterstützt. In diesem Fall wird das Array wie bisher verwendet: Jedes Element des Array zeigt den Status*

eingebildet/ausgebildet der entsprechenden Zeile in der Listbox. **Wahr** bedeutet ausgeblendet, **Falsch** bedeutet eingebildet.

Die Eigenschaft **Zeilenkontrolle Array** lässt sich über die Befehle **LISTBOX SET ARRAY** und **LISTBOX Get array** setzen oder erhalten. Das Array lässt sich auch mit dem Befehl **LISTBOX GET ARRAYS** zurückgeben.

- **Auswahlmodus:** Hier definieren Sie den Auswahlmodus für die Zeilen der Listbox. Es gibt drei Optionen:
 - **Nichts:** Sie können weder eine Zeile auswählen, noch Daten eingeben (außer, die Option "Einzelklick Editieren" ist markiert, siehe unter **Gruppe Eingabe**). Auswahl und Eingabe lassen sich nur per Programmierung verwalten. Klick oder Doppelklick in die Liste haben keine Auswirkung, selbst wenn die Option **Eingebbar** markiert ist. Die Ereignisse **On Clicked** und **On Double Clicked** lassen sich jedoch erzeugen. Mit dieser Option hat der Entwickler die volle Kontrolle, und zwar mit der Eigenschaft Auswahlmenge über Auswählen, mit dem 4D Befehl **EDIT ITEM** über die Dateneingabe. Die Ereignisse **On Selection Change** und **On Before Data Entry** werden nicht generiert. Das Ereignis **On After Edit** lässt sich dagegen erzeugen, wenn der Benutzer Daten über den 4D Befehl **EDIT ITEM** eingibt.
 - **Einfach:** Sie können nur eine Zeile auf einmal auswählen
 - **Mehrfach:** Sie können mehrere Zeilen auf einmal auswählen. Die standardmäßigen Tastenkombinationen lauten: **Umschalttaste+Klick** für eine fortlaufende Auswahl; unter Windows **Strg-Taste+Klick**, auf Mac OS **Befehlstaste+Klick** für eine unterbrochene Auswahl.
- **Doppelklick auf Zeile** (nur Listboxen vom Typ Auswahl): Bestimmt, welche Aktion durchgeführt wird, wenn der Benutzer in eine Zeile der Listbox doppelklickt. Es gibt folgende Optionen:
 - **Nichts** (Standard): Doppelklick in eine Zeile löst keine automatisch Aktion aus.
 - **Datensatz ändern:** Doppelklick in eine Zeile zeigt den entsprechenden Datensatz im Detail-Formular für die Listbox (siehe nächster Abschnitt "Eingabeformular"). Er wird im Lese-/Schreibmodus geöffnet, so dass er sich ändern lässt.
 - **Datensatz anzeigen:** ist identisch zur vorigen Aktion. Der Datensatz wird jedoch in Nur-Lesen Modus geöffnet, so dass er sich nicht ändern lässt.

Hinweis: Doppelklick in eine leere Zeile wird ignoriert.

Unabhängig von der gewählten Aktion wird hier das Formularereignis **On Double Clicked** erzeugt.

Für die beiden letzten Aktionen wird auch das Formularereignis **On Open Detail** generiert. Das Ereignis **On Close Detail** wird generiert, wenn ein Datensatz im Detail-Formular für die Listbox angezeigt wird und geschlossen werden soll - unabhängig, ob er geändert wurde.

- Eingabeformular (nur Listboxen vom Typ Auswahl): Definiert, welches Formular zum Ändern oder Anzeigen einzelner Datensätze der Listbox verwendet wird.

Das definierte Formular erscheint:

- wenn die Standardaktionen **Unterdatensatz hinzufügen** und **Unterdatensatz bearbeiten** für die Listbox angewandt werden (siehe **Standardaktionen verwenden**)
- Bei Doppelklick auf die Zeile, wenn für "Datensatz ändern" und "Datensatz anzeigen" die Eigenschaft **Doppelklick auf Zeile** gesetzt ist (siehe voriger Abschnitt "Doppelklick auf Zeile")

Gesperrte Spalten und statische Spalten

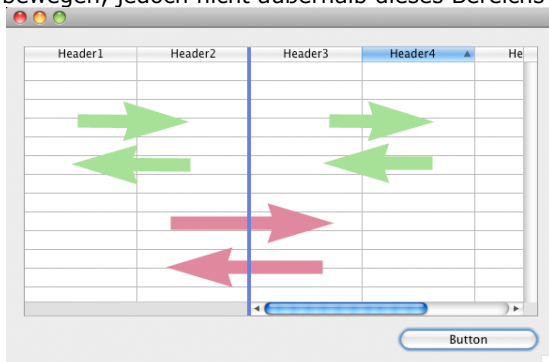
Gesperrte Spalten und statische Spalten sind zwei getrennte und von einander unabhängige Funktionalitäten in Listboxen:

- Gesperrte Spalten bleiben immer am linken Rand der Listbox sichtbar, sie verändern ihre Position nicht beim horizontalen Scrollen.
- Statische Spalten lassen sich innerhalb der Listbox nicht per Drag and Drop bewegen.

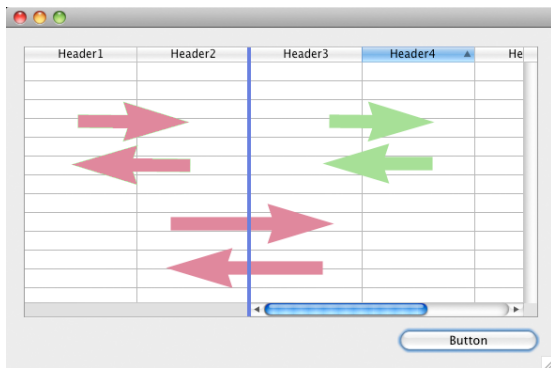
Hinweis: Sie können statische Spalten per Programmierung setzen. Weitere Informationen dazu finden Sie im Kapitel **Listbox** des Handbuchs *4D Programmiersprache*.

Die Interaktion dieser Eigenschaften ist wie folgt:

- Setzen Sie Spalten, die nur statisch sind, lassen sie sich nicht bewegen.
- Setzen Sie Spalten, die gesperrt, aber nicht statisch sind, lassen sie sich innerhalb des gesperrten Bereichs frei bewegen, jedoch nicht außerhalb dieses Bereichs setzen.



- Definieren Sie dieselbe Anzahl gesperrte Spalten wie statische Spalten, können Sie diese Spalten innerhalb des gesperrten Bereichs nicht bewegen.



- Sie können je nach Ihren Bedürfnissen eine Kombination aus gesperrten und statischen Spalten setzen. Setzen Sie z.B. drei gesperrte Spalten und eine statische Spalte, kann der Benutzer die beiden rechts im gesperrten Bereich gelegenen Spalten verschieben, da nur die erste Spalte statisch ist.

Gruppen Kopfteile und Fußteile

Diese Gruppen verwalten die Anzeige und Höhe von Kopf- und Fußteilen für Spalten von Listboxen. Beachten Sie, dass Kopf- und Fußteile nicht eingebbar sind. Ihr Inhalt wird immer berechnet.

- **Kopfteil / Fußteil anzeigen:** Dient zum Anzeigen oder Ausblenden von Kopf- und Fußteilen. Die anderen Optionen sind nur verfügbar, wenn diese Option markiert ist. Wird der Bereich in der Listbox angezeigt, können Sie darauf klicken, um ihn auszuwählen und dann die anderen Optionen in der Eigenschaftsliste nutzen. (siehe **Eigenschaften für Spaltentitel der Listbox** und **Eigenschaften für Fußteile der Listbox**). Sie können pro Spalte ein Kopfteil und ein Fußteil haben; jedes Teil lässt sich separat konfigurieren.
- **Standardmäßig unsichtbar:** Wie bei allen Formularobjekten ermöglicht diese Option die dynamische Anzeige über den Befehl **OBJECT SET VISIBLE**. Der Befehl **OBJECT SET VISIBLE** hat keine Auswirkung, wenn die Option zum Anzeigen des Kopf- bzw. Fußteilbereichs in der Eigenschaftsliste nicht markiert ist.
- **Höhe:** Dient zum Setzen der Zeilenhöhe für ein Kopf- oder Fußteil in der Listbox. Sie können die Einheit (**Zeilen** oder **Pixel**) für den Wert der Höhe wählen. Weitere Informationen dazu finden Sie im nachfolgenden Abschnitt **Höhe in Pixel oder Zeilen**.
Standardmäßig ist die Höhe von Kopf- bzw. Fußteilen **1 Zeile**.
Warnung: Die Mindesthöhe in Pixel für Kopfteile richtet sich nach dem Betriebssystem. Übergeben Sie einen zu kleinen Wert, wird er durch den im System festgelegten Mindestwert ersetzt. Für Fußteile und Zeilen gibt es keine Mindesthöhe.
Hinweis zur Kompatibilität: Unter Windows 7 und Windows Vista ist die Mindesthöhe für Kopfteile 24 Pixel. Haben Kopfteile in Ihren konvertierten Datenbanken eine geringere Höhe, werden diese automatisch angepasst. In solchen Fällen müssen Sie Ihre Formulare evtl. entsprechend anpassen.

Sie können die Zeilenhöhe von Kopf- bzw. Fußteil über die Befehle **LISTBOX SET HEADERS HEIGHT** und **LISTBOX SET FOOTERS HEIGHT** auch dynamisch setzen.

Höhe in Pixel oder Zeilen

Sie können die Höhe für Kopf-, Fußteile und Zeilen entweder in **Pixel** oder in **Zeilen** setzen. In der Eigenschaftsliste setzen Sie die Einheit über die Schaltfläche rechts neben dem Eintrag "Höhe".



Sie können auch direkt im Wertebereich ein "L" (für "line"= Zeile) oder ein "P" (für Pixel) eingeben, z.B. "17 P". Die Bezeichnung der Schaltfläche wird dann entsprechend angepasst.

Sie können in derselben Listbox auch beide Einheiten zusammen verwenden:

- Bei "Pixel" wird der Wert der Höhe direkt auf die betreffende Zeile angewandt, unabhängig von der Schriftgröße für die Spalten. Ist sie zu groß, wird der Text abgeschnitten. Außerdem werden Bilder je nach Format abgeschnitten oder angepasst.
- Bei "Zeile" wird die Höhe unter Berücksichtigung der Schriftgröße der betreffenden Zeile berechnet.
 - Sind mehrere Werte definiert, verwendet 4D den größten. Enthält eine Zeile z.B. "Verdana 18", "Geneva 12" und "Arial 9", verwendet 4D für die Zeilenhöhe "Verdana 18" (z.B. 25 Pixel). Diese Höhe wird dann mit der Anzahl der definierten Zeilen multipliziert.
 - Diese Berechnung berücksichtigt nicht die Größe von Bildern oder den Schriften zugewiesene Stilarten.
 - Auf OS X kann die Zeilenhöhe u.U. inkorrekt sein, wenn der Benutzer Zeichen eingibt, die in der gewählten Schrift nicht verfügbar sind. Ist das der Fall, wird eine Ersatzschrift verwendet, was zu einer Variation der Größe führen kann.
- **Einheit wechseln:** Wechseln Sie von Zeile in Pixel oder umgekehrt, wandelt 4D die Einheit automatisch um und zeigt das Ergebnis in der Eigenschaftsliste an. So wird z.B. bei der Schrift "Lucida grande 24" die Höhe "1 Zeile" in "30 Pixel" umgewandelt, die Höhe "60 Pixel" in "2 Zeilen".
Beachten Sie, dass das Vor- und Zurückverwandeln zu einem anderen Endergebnis als dem Startwert führen kann, da 4D automatische Berechnungen durchführt. Dies veranschaulicht das folgende Beispiel:
(Schrift Arial 18): 52 Pixel -> 2 Zeilen -> 40 Pixel
(Schrift Arial 12): 3 Pixel -> 0,4 Zeile gerundet auf 1 Zeile -> 19 Pixel

Rasterlinien

Diese Gruppe enthält die Eigenschaften für Rasterlinien im Objekt *Listbox*.

- **Horizontale Linien:** Blendet die horizontalen Linien der Listbox ein oder aus. Sie sind standardmäßig eingeblendet.
- **Horizontale Linienfarbe:** Definiert die Farbe der horizontalen Linien der Listbox. Standardmäßig ist grau vorgegeben.
- **Vertikale Linien:** Blendet die vertikalen Linien der Listbox ein oder aus. Sie sind standardmäßig eingeblendet.
- **Vertikale Linienfarbe:** Definiert die Farbe der vertikalen Linien der Listbox. Standardmäßig ist grau vorgegeben.

Hierarchie

Nur Listboxen vom **Typ Array** können hierarchisch sein. Diese Option konfiguriert die hierarchische Anzeige der Listbox. Beachten Sie, dass diese Eigenschaften automatisch geändert werden, wenn Sie die Hierarchie über das PopUp-Menü des Objekts **Listbox** definieren (siehe **Hierarchische Listboxen erstellen und verwenden**).

- **Hierarchische Listbox:** Damit definieren Sie in der Eigenschaftsliste, dass die Listbox in hierarchischer Form angezeigt wird.
- **Variable 1 ... 10:** Diese zusätzlichen Optionen erscheinen, wenn **Hierarchische Listbox** markiert ist. Immer wenn ein Wert in ein Feld eingegeben wird, wird eine neue Zeile hinzugefügt. Es lassen sich bis zu 10 Variablen definieren. Diese Variablen setzen die hierarchischen Ebenen, die in der ersten Spalte angezeigt werden sollen.

Hierarchie	
Hierarchische Listbox	<input checked="" type="checkbox"/>
Variable 1	arr1
Variable 2	arr2
Variable 3	arr3

Die erste Variable entspricht immer dem Variablennamen für die erste Spalte der Listbox. Die beiden Werte werden automatisch verknüpft. Die erste Variable ist immer sichtbar und eingebbar, z.B. Land.

Die zweite Variable ist auch immer sichtbar und eingebbar; sie definiert die zweite hierarchische Ebene, z.B. Bundesland.

Ab dem dritten Feld hängt jede Variable von der vorangegangenen ab, z.B. Landkreis, Stadt, usw. Es sind bis zu zehn hierarchische Ebenen möglich.

Entfernen Sie einen Wert, steigt die gesamte Hierarchie eine Ebene höher.

Die letzte Variable kann nie hierarchisch sein, selbst wenn auf dieser Ebene mehrere gleichwertige Werte existieren. Nehmen wir z.B. eine Konfiguration, wo Arr1 die Werte A A B B B, Arr2 die Werte 1 1 1 2 2 2 und Arr3 die Werte X X Y Y Z enthält. In diesem Fall können A, B, 1 und 2 in zugeklappter Form erscheinen, X und Y dagegen nicht:

```
+ A
  + 1
    X
    X
    Y
  + B
    + 2
      Y
      Y
      Z
```

Dieses Prinzip gilt nicht, wenn nur eine Variable in der Hierarchie angegeben ist: In diesem Fall werden identische Werte gruppiert.

Hinweis: Definieren Sie eine Hierarchie, die auf den ersten Spalten einer vorhandenen Liste basiert, müssen Sie diese Spalten mit Ausnahme der ersten entfernen oder ausblenden, da sie sonst in der Listbox doppelt erscheinen. Definieren Sie die Hierarchie über das PopUp-Menü des Editors, werden die nicht benötigten Spalten automatisch aus der Listbox entfernt.

Gruppe Koordinaten & Größe

Diese Gruppe enthält alle Eigenschaften für Koordinaten, Breite und Höhe der Listbox.

Die Eigenschaft **Zeilenhöhe** gilt für die Höhe von Zeilen. Die Höhe von Kopf- und Fußteilen definieren Sie unter der Gruppe "Kopfteile" bzw. "Fußteile".

Sie können die Höhe in Pixel oder Zeilen setzen. Weitere Informationen dazu finden Sie im oberen Abschnitt **Höhe in Pixel oder Zeilen**. Standardmäßig richtet sie sich nach Plattform und Schriftgröße.

Eigenschaft Zeilenhöhe Array

Nur 4D View Pro: Für dieses Feature wird eine 4D View Pro Lizenz benötigt. Weitere Informationen dazu finden Sie im **4D View Pro Handbuch**.

Hinweis: Für Listboxen, die auf Arrays basieren, ist diese Eigenschaft nur verfügbar, wenn die Option **Automatische Zeilenhöhe** nicht markiert ist (siehe nächster Absatz)

Diese Eigenschaft gibt den Namen des Array für Zeilenhöhe an, das Sie der Listbox zuordnen wollen; es muss vom Typ Zahl sein (standardmäßig Länge Ganzzahl)

Ist ein Array für Zeilenhöhe definiert, wird jedes Element mit einem Wert unterschiedlich zu 0 (Null) berücksichtigt, um die Höhe der entsprechenden Zeile in der Listbox zu definieren. Dabei gilt die aktuelle Einheit für Zeilenhöhe.

Sie können beispielsweise schreiben:

```
ARRAY LONGINT(RowHeights;20)
RowHeights{5};=3
```

Gilt als Einheit "Zeilen", ist die fünfte Zeile der Listbox drei Zeilen hoch, während alle anderen Zeilen die Standardhöhe beibehalten.

Das Array Zeilenhöhe lässt sich auch über den Befehl **LISTBOX SET ROW HEIGHT** verwalten.

Hinweis: Diese Eigenschaft wird für hierarchische Listboxen nicht berücksichtigt.

Automatische Zeilenhöhe

Nur 4D View Pro: Für dieses Feature wird eine 4D View Pro Lizenz benötigt. Weitere Informationen dazu finden Sie im **4D View Pro Handbuch**.

Die Eigenschaft **Automatische Zeilenhöhe** ist nur für auf Arrays basierende und nicht hierarchische Listboxen verfügbar. Sie ist standardmäßig nicht markiert. Wird sie markiert, erscheinen jetzt die Optionen **Min Zeilenhöhe** und **Max Zeilenhöhe** anstelle von *Zeilenhöhe* und *Zeilenhöhe Array* (siehe oben).

Diese Eigenschaft lässt sich für die Listbox oder für jede Spalte setzen (siehe Abschnitt **Eigenschaften für Spalten der Listbox**):

- Wird die Eigenschaft auf Listboxebene gesetzt, werden zum Berechnen der Zeilenhöhe alle Spalten berücksichtigt.
- Wird die Eigenschaft auf Spaltenebene gesetzt, werden zum Berechnen der Zeilenhöhe nur die Spalten mit der markierten Option berücksichtigt.

Hinweis: Wird die Eigenschaft für beide Ebenen gesetzt, wird die Listboxebene ignoriert. Nur die betroffenen Spalten werden berücksichtigt.

Ist diese Eigenschaft markiert, wird die Höhe jeder Zeile automatisch berechnet, so dass der Zelleninhalt ganz hineinpasst und nicht abgeschnitten wird - außer die Option **Zeilenumbruch** ist deaktiviert (siehe unten).

- Die Berechnung der Zeilenhöhe berücksichtigt:
 - alle Arten von Inhalten (Text, Zahlen, Datum, Zeit, Bilder *, Objekte),
 - alle Steuerungstypen (Eingaben, Optionsfelder, Listen, DropDowns),
 - Schrifttypen, Schriftstile und Schriftgrößen,
 - Die Option **Zeilenumbruch**: Ist sie deaktiviert, richtet sich die Höhe nach der Anzahl der Absätze (Zeilen werden abgeschnitten); ist sie aktiviert, basiert die Höhe auf der Anzahl der Zeilen (nicht abgeschnitten).

() Die Berechnung hängt vom Bildformat ab.*

- Die Berechnung der Zeilenhöhe ignoriert:
 - Ausgeblendeten Inhalt in Spalten
 - Die Eigenschaften **Zeilenhöhe** und **Zeilenhöhe Array** (falls vorhanden), entweder in der Eigenschaftsliste oder per Programmierung gesetzt.

Hinweis: Da in Echtzeit weitere Berechnungen erforderlich sind, kann die Option automatische Zeilenhöhe das Scrollen in Ihrer Listbox beeinträchtigen, insbesondere bei einer großen Anzahl von Zeilen.

Min Zeilenhöhe und **Max Zeilenhöhe** sind zusätzliche Eigenschaften. Sie erscheinen, wenn die Option **Automatische Zeilenhöhe** markiert ist. Ihr Wert kann in **Pixel** oder **Zeilen** definiert werden. Der Wert von "Min" muss kleiner oder gleich dem Wert von "Max" sein (bei Definition mit derselben Einheit), sonst wird in Echtzeit der Maximumwert auf den Mindestwert gesetzt.

Hinweise:

- Die Eigenschaften **Min Zeilenhöhe** und **Max Zeilenhöhe** lassen sich nur auf Listboxebene und nicht für einzelne Spalten setzen.
- Diese Eigenschaften lassen sich auch über die Befehle **LISTBOX Get auto row height** und **LISTBOX SET AUTO ROW HEIGHT** verwalten.

Gruppe Vergrößerungsoptionen

Diese Gruppe enthält Optionen zum Einrichten der Listbox beim Anpassen der Größe. Sie gelten sowohl bei Ausführen durch Benutzer als auch per Programmierung, z.B. über den Befehl **OBJECT SET COORDINATES**.

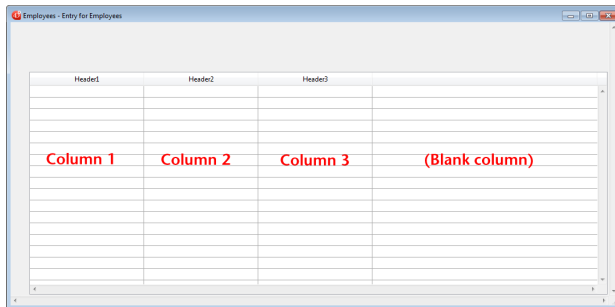
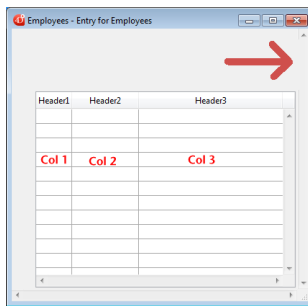
- **Horizontale Größe / Vertikale Größe:** Diese Standardeigenschaften zum Anzeigen von Objekten werden im Abschnitt **Größe anpassen** beschrieben.
- **Spaltenbreite Automatisch:** Ist diese Eigenschaft aktiviert, wachsen die Spalten automatisch mit, wenn die Größe der Listbox verändert wird, und zwar innerhalb der definierten Mindest- und Maximumwerte für die Breite (siehe unten).

Ist diese Eigenschaft nicht markiert (Standard in Anwendungen, die von einer 4D Version vor v16 konvertiert wurden), wird nur die ganz rechts liegende Spalte der Listbox angepasst, selbst wenn die Breite den definierten Maximumwert überschreitet.

Funktionsweise

- Bei zunehmender Breite der Listbox werden die Spalten der Reihe nach von rechts nach links verbreitert, bis die maximale Breite erreicht ist.
Hinweis: Alle Spalten mit markierter Eigenschaft **Vergrößerbar** werden angepasst. Diese Eigenschaft lässt sich für jede Spalte einzeln setzen.
- Dasselbe gilt bei abnehmender Breite der Listbox, nur dass das Verkleinern von links nach rechts abläuft. Hat jede Spalte die Mindestbreite erreicht, wird der horizontale Rollbalken wieder aktiv.
- Spalten werden nur angepasst, wenn der horizontale Rollbalken nicht "aktiv" ist, z.B. alle Spalten sind in der aktuell angezeigten Listbox voll sichtbar.
Hinweis: Ist der horizontale Rollbalken ausgeblendet, verändert das nicht seinen Status, d.h. er bleibt auch aktiv, wenn er nicht sichtbar ist.

- Haben alle Spalten ihre maximale Größe erreicht, werden sie nicht weiter vergrößert. Stattdessen wird am rechten Rand eine leere Proforma Spalte hinzugefügt, um den verbleibenden Bereich zu füllen.



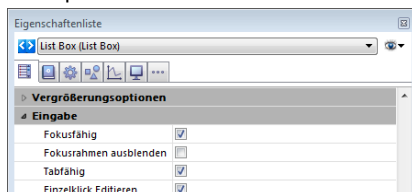
Hinweise:

- Gibt es eine Proforma Spalte (leer), wird dieser Bereich beim Verringern der Listboxbreite zuerst reduziert.
- Das Aussehen der Proforma Spalte richtet sich nach den vorhandenen Spalten. Sie enthält auch Kopf- bzw. Fußteil, wenn diese in den Spalten der Listbox vorhanden sind und wendet die gleichen Hintergrundfarben an.
- Proforma Kopf- und Fußteil lassen sich anklicken, das hat aber keine Auswirkung auf die anderen Spalten (z.B. keine Sortierung); dagegen werden die jeweiligen Ereignisse On Clicked, On Header Click und On Footer Click generiert.
- Wird in der Proforma Spalte auf eine Zelle geklickt, gibt der Befehl **LISTBOX GET CELL POSITION** für die Spaltennummer "X+1" zurück (X ist die Anzahl der vorhandenen Spalten).

Gruppe Eingabe

Diese Gruppe enthält Standardeigenschaften zur Dateneingabe in der Listbox.

Die Option **Einzelklick Editieren** ist speziell, sie aktiviert den direkten Übergang in den Bearbeitungsmodus.



Ist diese Option markiert, wechseln Zellen von Listboxen nach einfachem Klick durch den Benutzer in den Bearbeitungsmodus, unabhängig, ob vorher dieser Bereich der Listbox ausgewählt wurde. Beachten Sie, dass Zellen auch editierbar sind, selbst wenn der Auswahlmodus für Listbox auf "Nichts" gesetzt ist. Weitere Informationen dazu finden Sie im Abschnitt **Gruppe Listbox**.

Ist diese Option nicht markiert, müssen Benutzer zuerst die entsprechende Zeile auswählen und dann in die Zelle klicken, um den Inhalt zu bearbeiten. Dies ist das standardmäßige Verhalten in 4D vor v15 R3. Zur Wahrung der Kompatibilität ist die Option standardmäßig deaktiviert.

Gruppe Anzeige

Diese Gruppe enthält zwei Eigenschaften zur globalen Anzeige der Listbox:

- **Unsichtbar als Standard** verwaltet generisch die Anzeige des Objekts beim Laden des Formulars (siehe Abschnitt **Unsichtbar als Standard**).
- **Abkürzen mit Auslassungspunkten** steuert die Anzeige der Werte, wenn die Spalte der Listbox für den ganzen Inhalt zu schmal ist. Diese Option ist für verschiedene Arten von Inhalt in Spalten verfügbar. Davon ausgenommen sind Bilder und Objekte.
 - Ist die Option **markiert** (Standard) und ist der Inhalt größer als die Breite der Spalte, wird er abgeschnitten und es erscheinen Auslassungspunkte:

Cities	Popu...
Charlotte	792862
New York	8406...
Philadelp...	1553...
San Fran...	837442
San Jose	288054

Hinweis: Die Position der Auslassungspunkte richtet sich nach dem Betriebssystem. Im obigen Beispiel

Markieren Sie diese Option, erscheint das Objekt Listbox ohne Leerzeilen am Ende:

Days	Values
Monday	5
Tuesday	6
Wednesday	41
Thursday	66
Friday	12
Saturday	2
Sunday	88

Gruppe Text

In dieser Gruppe definieren Sie die Standardeigenschaften für Text in der Listbox: Stilvorlage, Schrifttyp, Schriftstil, etc. sowie einige spezifische Eigenschaften zum Anzeigen von Text in jeder Zelle der Listbox:

- **Zeilenstil Array** (Listboxtyp Array) / **Zeilenstil** (Listboxtyp Auswahl und Collection/Entity-Selection): Hier weisen Sie jeder Zeile der Listbox einen eigenen Schriftstil zu.
 - Für Listboxen vom Typ Array müssen Sie den Namen eines Array vom Typ Lange Ganzzahl eingeben. Jedes Element im Array entspricht einer Zeile in der Listbox. Das Array muss dieselbe Größe haben wie die den Spalten zugeordneten Arrays. Zum Füllen des Array über eine Methode wählen Sie eine Konstante unter dem Thema **Schriftstile** im Handbuch *4D Programmiersprache*. Sie können auch mehrere Stilarten miteinander kombinieren. Um den Stil, der in den Eigenschaften der Listbox definiert wurde, der Zeile zuzuweisen, übergeben Sie den Wert -255 im Element des dazugehörigen Array.
 - Für Listboxen vom Typ Auswahl und Collection/Entity-Selection müssen Sie einen Ausdruck oder eine Variable (außer vom Typ Array) eingeben. Der Ausdruck bzw. die Variable wird für jede angezeigte Zeile ausgewertet. Einen Ausdruck können Sie über den Formeleditor definieren. Klicken Sie dazu auf die Schaltfläche [...]. Sie erscheint, wenn Sie den Bereich auswählen. Sie können auch die Konstanten unter dem Thema **Schriftstile** im Handbuch *4D Programmiersprache* verwenden.

Hinweis: Für Listboxen vom Typ Collection oder Entity-Selection lässt sich diese Eigenschaft auch mit der Option **Meta Info Ausdruck** setzen (siehe unten).

Das folgende Beispiel verwendet einen Variablennamen: Geben Sie *FirmaStil* im Bereich **Zeilenstil** ein und schreiben Sie folgende Methode:

```
FirmaStil:=Choose([Firma]ID;Bold;Plain;Italic;Underline)
```

- **Zeilen Schriftfarbe Array** (Listboxtyp Array) / **Zeilenschriftfarbe** (Listboxtyp Auswahl und Collection/Entity-Selection): Hier weisen Sie jeder Zeile der Listbox eine eigene Schriftfarbe zu. Sie müssen einen RGB-Farbwert verwenden. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter dem Befehl **OBJECT SET RGB COLORS**.
 - Für Listboxen vom Typ Array müssen Sie den Namen eines Array vom Typ Lange Ganzzahl eingeben. Jedes Element im Array entspricht einer Zeile in der Listbox. Das Array muss dieselbe Größe haben wie die den Spalten zugeordneten Arrays. Sie können eine Konstante unter dem Thema **SET RGB COLORS** der *4D Programmiersprache* verwenden. Um die Schriftfarbe, die in den Eigenschaften der Listbox definiert wurde, einer Zeile zuzuweisen, übergeben Sie den Wert -255 im Element des dazugehörigen Array.
 - Für Listboxen vom Typ Auswahl und Collection/Entity-Selection müssen Sie einen Ausdruck oder eine Variable (außer vom Typ Array) eingeben. Der Ausdruck bzw. die Variable wird für jede angezeigte Zeile gezeigt. Einen Ausdruck können Sie über den Formeleditor definieren. Klicken Sie dazu auf die Schaltfläche [...]. Sie erscheint, wenn Sie den Bereich auswählen. Sie können auch die Konstanten unter dem Thema **SET RGB COLORS** im Handbuch *4D Programmiersprache* verwenden.

Hinweis: Für Listboxen vom Typ Collection oder Entity-Selection können Sie diese Eigenschaft auch mit der Option **Meta Info Ausdruck** setzen (siehe unten).

Das folgende Beispiel verwendet einen Variablennamen: Geben Sie *FirmaFarbe* im Bereich **Zeilenschriftfarbe** ein und schreiben Sie folgende Methode:

```
FirmaFarbe:=Choose([Firma]ID;Default background color;Default light shadow color;Default foreground color;Default dark shadow color)
```

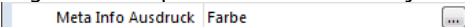
- **Horizontale Ausrichtung:** Der Inhalt jeder Zelle lässt sich horizontal **Links**, **Rechts** oder **Zentriert** ausrichten. Die Option **Standard** setzt die Ausrichtung gemäß dem jeweils gefundenen Datentyp: Text und Bilder werden linksbündig, numerische Daten werden rechtsbündig ausgerichtet.
- **Vertikale Ausrichtung:** Der Inhalt jeder Zelle lässt sich vertikal **Oben**, **Zentriert** oder **Unten** ausrichten. Die Option **Standard** setzt die Ausrichtung gemäß dem jeweils gefundenen Datentyp: **Unten** für alle Daten bis auf Bilder, **Oben** für Bilder.

- **Meta Info Ausdruck:** (Nur Listbox vom Typ Collection oder Entity-Selection) Gibt einen Ausdruck oder eine Variable an, die für jede angezeigte Zeile bewertet wird. Damit lässt sich ein ganzer Satz von Textattributen für Zeilen definieren. Sie müssen eine Objekt-Variable oder einen Ausdruck übergeben, der ein Objekt zurückgibt. Es gibt folgende Eigenschaften:

Eigenschaftsname	Typ	Beschreibung
stroke	String	Schriftfarbe. Jede CSS Farbe (z.B.: "#FF00FF"), "automatic", "transparent"
fill	String	Hintergrundfarbe. Jede CSS Farbe (z.B.: "#F00FFF"), "automatic", "transparent"
fontStyle	String	"normal", "italic"
fontWeight	String	"normal", "bold"
textDecoration	String	"normal", "underline"
unselectable	Boolean	Macht die entsprechende Zeile nicht auswählbar. Dann lässt sie z.B. nicht mehr markieren und eingebare Bereiche sind nicht mehr eingebbar, außer die Option "Einzelklick Editieren" ist ebenfalls markiert. Steuerung wie Kontrollkästchen und Listen bleiben funktional. Diese Einstellung wird ignoriert, wenn als Auswahlmodus für Listbox "Nichts" gewählt ist. Standardwert: Falsch.
disabled	boolean	Deaktiviert die entsprechende Zeile. Eingebare Bereiche sind damit nicht mehr eingebbar. Text und Steuerung (Kontrollkästchen, Listen, etc.) erscheinen gedimmt oder in Grau. Standardwert: Falsch.
cell.<columnName>	Objekt	Damit lässt sich die Eigenschaft auf eine einzelne Spalte anwenden. In <columnName> übergeben Sie den Objektnamen der Listboxspalte. Hinweis: Die Eigenschaften "unselectable" und "disabled" lassen sich nur auf Zeilenebene definieren. Bei Übergeben im Objekt "cell" werden sie ignoriert.

Hinweis: Mit dieser Eigenschaft gesetzte Stileinstellungen überschreiben andere über Ausdrücke definierte Stileinstellungen, wie z.B. Stil-Ausdruck, Schriftfarbe-Ausdruck, Hintergrundfarbe-Ausdruck, etc.

Folgendes Beispiel verwendet die Projektmethode *Farbe*: Geben Sie *Farbe* im Bereich **Meta Info Ausdruck** ein



In der Methode *Farbe* schreiben Sie:

```
//Methode <em>Farbe</em>
//Setzt Schriftfarbe für bestimmte Zeilen und Hintergrundfarbe für eine bestimmte Spalte:
C_OBJECT($0;$1)
if(This.ID>5) //ID ist ein Attribut von Collection Objekten/Entities
    $0:=New object("stroke";"purple";"cell";New object("Column2";New object("fill";"black")))
Else
    $0:=New object("stroke";"orange")
End if
```

Hinweise:

- Weitere Informationen dazu finden Sie unter dem Befehl **This**.
- In diesem Beispiel verwenden wir ein neues Objekt für mehr Klarheit. In der Regel wird jedoch empfohlen, beim Laden des Formulars statische Daten in das Objekt **Form** zu laden und dann z.B. mit **Form.stroke** darauf zuzugreifen.

Sie können die Einstellung für die gesamte Listbox oder für jede Spalte einzeln definieren (siehe **Schriftstil und -farbe verwalten**). Sie ist auch für Kopf- und Fußteile verfügbar.

Gruppe Aktion

Diese Gruppe enthält alle Eigenschaften zum dynamischen Verhalten der Listbox.

- **Methode** (Bearbeiten...): Über diese Schaltfläche zeigen Sie die Methode des Objekts vom Typ Listbox an. Beachten Sie, dass auch jede Spalte eine Objektmethode enthalten kann.
- **Dragfähig** und **dropfähig**: Diese Option aktiviert die Drag- und Drop-Funktionen für Listboxen. Sie können dann die Zeile einer Listbox per Drag&Drop in eine andere Listbox bzw. ein anderes Objekt ziehen und umgekehrt. Das gilt jedoch nur für Zeilen; nicht für Spalten. Spalten lassen sich nur innerhalb derselben Listbox bewegen. Drag&Drop in Listboxen erfolgt über die Standard Funktionalitäten von 4D. Das sind die Formularereignisse **On Drop** und **On Drag Over** sowie die 4D Befehle **DRAG AND DROP PROPERTIES** und **Drop position**.
- **Verschiebbare Zeilen** (nur Listboxtyp Array): Diese Option ist standardmäßig aktiv. Damit lassen sich Zeilen während der Ausföhrung bewegen. Die Option ist für Listboxen vom Typ Auswahl sowie im hierarchischen Modus (Option **Hierarchische Listbox** ist markiert) nicht verfügbar.
- **Sortierfähig**: Diese Option ist standardmäßig aktiv. Damit können Sie durch Anklicken des Spaltentitels Daten in Spalten sortieren. Es lassen sich jedoch keine Arrays (Spalten) vom Typ Bild sortieren. In Listboxen, die auf einer Datensatzauswahl basieren, sind nur die standardmäßigen Sortierfunktionen aktiv:
 - Wenn die Datenquelle die **aktuelle Auswahl** ist
 - Für Spalten, denen Datenfelder vom Typ Alphanumerisch, Zahl, Datum, Zeit oder Boolean zugewiesen sind
 In anderen Fällen (Listboxen mit temporären Auswahlen, Spalten mit Ausdrücken) ist die Standardsortierfunktion nicht verfügbar.

Eine Standardsortierung stimmt alle Spalten der Listbox aufeinander ab, inkl. der berechneten Spalten.

Gruppe Kompatibilität

Für Listboxen, die von rollbaren Bereichen kommen, ist unter dieser Gruppe automatisch die Option **Rollbarer Bereich** markiert. Bisherige rollbare Bereiche werden ab 4D v13 automatisch in Listboxen umgewandelt. Die Option ist jedoch für alle Listboxen verfügbar:



Ist diese Option markiert, funktioniert die Listbox auf eine spezielle Art:

- Hat das Array (einmalig) der Listbox die Eigenschaft "unsichtbar", ist das Objekt Listbox ebenfalls komplett unsichtbar.
- Wird dem Array ein Wert zugewiesen, wird die entsprechende Zeile in der Listbox gewählt. Beispiel: MeinArray:=5 wählt die 5. Zeile der Listbox.
- Umgekehrt wird durch Anklicken einer Zeile der aktuelle Wert des Array geändert.
- Erfolgt von einer Zeile der Listbox ein Drop zu einem externen Objekt, gibt der in diesem Objekt ausgeführte Befehl **DRAG AND DROP PROPERTIES** einen Zeiger auf das Array der Listbox und nicht auf die Listbox selbst zurück.

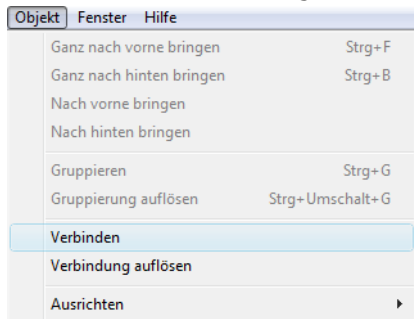
Verbundene Listboxen (Kompatibilität)

Listboxen, die aus den früher vorhandenen gruppierten rollbaren Bereichen konvertiert wurden, sind **verbunden** und funktionieren in koordinierter Form:

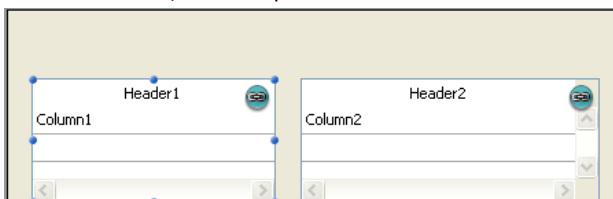
- Bei Auswählen einer Zeile in einer Listbox wird dieselbe Zeile in allen Listboxen ausgewählt, die zur verbundenen Gruppe gehören
- Bei vertikalem Scrollen in einer Listbox wird auch in allen anderen Listboxen gescrollt, die zur verbundenen Gruppe gehören.

Hinweis: Konvertierte Listboxen sind im Formular auch **gruppiert** (standardmäßige 4D Funktion).

Sie können diese Listboxen miteinander verbinden oder die Verbindung auflösen. Wählen Sie dazu im Menü **Objekt** des Formulareditors den Eintrag **Verbinden** bzw. **Verbindung auflösen**:



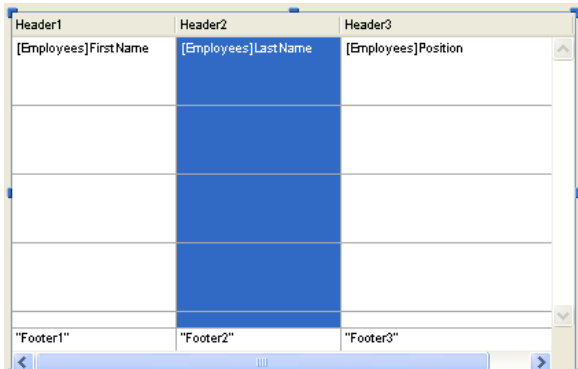
Diese Befehle werden aktiviert, wenn mehrere Listboxen im Formular ausgewählt sind. Wird eine verbundene Listbox ausgewählt (sie gehört z.B. zu einer verbundenen Gruppe), erscheint in allen Listboxen, die mit derselben Gruppe verbunden sind, ein entsprechendes Icon:



Hinweis zur Kompatibilität: Diese Mechanismen ermöglichen, die Funktionsweise gruppiert rollbarer Bereiche zu reproduzieren. Wir empfehlen Ihnen jedoch, konvertierte Formulare mit Hilfe der standardmäßigen Funktionalitäten für Listboxen anzupassen.

Eigenschaften für Spalten der Listbox

Ist das Objekt Listbox gewählt, können Sie im Formulareditor eine Spalte der Listbox durch Anklicken auswählen:



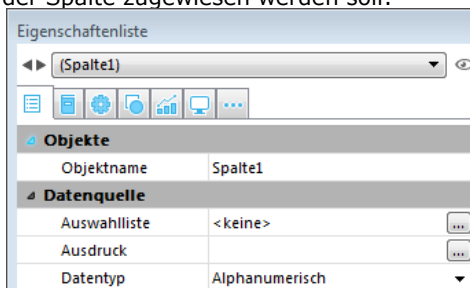
Sie können für jede Spalte der Listbox Standardeigenschaften, wie Text, Hintergrundfarbe, etc. setzen; diese Eigenschaften haben Vorrang vor den Objekteigenschaften der Listbox.

Sie können auch auf spezifische Eigenschaften zugreifen. Diese werden nachfolgend beschrieben.

Gruppe Datenquelle

Hinweis: Für Listboxspalten des Objekttyps Array ist die Gruppe **Datenquelle** nicht verfügbar. Der Inhalt jeder Zelle der Spalte richtet sich nach den Attributen, die dem jeweiligen Element des Objekt Array zugewiesen ist. Weitere Informationen dazu finden Sie im Abschnitt **Listbox: Objekt Arrays in Spalten verwenden (4D View Pro)**.

- **Ausdruck** (Listbox vom Typ Auswahl und Collection/Entity-Selection): Damit definieren Sie, welcher 4D Ausdruck der Spalte zugewiesen werden soll:



Sie können folgendes eingeben:

- Eine einfache Variable (sie muss explizit zum Kompilieren deklariert sein). Sie können jede Variablenart verwenden mit Ausnahme von BLOBs und Arrays. Der Variablenwert wird im allgemeinen im Ereignis On Display Detail berechnet.
- Ein Datenfeld mit der Standardsyntax [Tabelle]Feld (nur Listbox vom Typ Auswahl), wie [Angestellte]Nachname). Es sind folgende Datenfeldtypen möglich:

- String
- Zahl
- Datum
- Zeit
- Bild
- Boolean

Sie können Datenfelder von der Haupttabelle oder von anderen Tabellen verwenden.

- Einen *4D Ausdruck* (einfacher Ausdruck, Formel oder 4D Methode). Der Ausdruck muss einen Wert zurückgeben. Der Wert wird in den Ereignissen On Display Detail und On Data Change berechnet. Das Ergebnis des Ausdrucks erscheint automatisch beim Wechseln in den Anwendermodus. Der Ausdruck wird für jeden Datensatz der Auswahl (aktuell oder temporär) der Haupttabelle gewertet (für Listboxen vom Typ Auswahl), jedes Element der Collection (für Listboxen vom Typ Collection) oder jede Entity der Selection (für Listboxen vom Typ Entity-Selection). Ist er leer, zeigt die Spalte keinerlei Ergebnisse.

Folgende Typen werden unterstützt:

- String
- Zahl
- Datum
- Bild
- Boolean
- Für Listboxen vom Typ Collection/Entity-Selection werden Null oder nicht unterstützte Typen als leere Strings angezeigt.

Bei Collections oder Entity-Selections deklarieren Sie die der Spalte zugeordnete Eigenschaft des Collection-Elements oder der Entity in der Regel mit einem Ausdruck mit **This**. Das ist ein spezifischer 4D Befehl, der eine Referenz auf das aktuell bearbeitete Element zurückgibt (siehe unter dem Befehl **This**). Sie können z.B. **This**.

<propertyPath> verwenden, wobei <propertyPath> der Pfad zur Eigenschaft in der Collection oder zum Entity-Attribut ist, um auf den aktuellen Wert jedes Elements bzw. jeder Entity zuzugreifen. Bei einer Collection mit skalaren Werten erstellt 4D ein Objekt für jedes Collection-Element mit einer einzigen Eigenschaft (genannt "value"), die mit dem Wert des Elements gefüllt wird. In diesem Fall verwenden Sie **This.value** als Ausdruck.

Sie können den Ausdruck in jedem Fall über den 4D Formulareditor definieren. Klicken Sie dazu in der Eigenschaftenliste auf die Schaltfläche [...].

Bei einem nicht-zuweisbaren Ausdruck (z.B. `[Person]FirstName` " "+`[Person]LastName`) ist die Spalte nie eingebbar, selbst wenn die Option **Eingebbar** aktiv ist.

Bei einem Datenfeld, einer Variablen oder einem zuweisbaren Ausdruck (z.B. `Person.lastName`) kann die Spalte eingebbar sein oder nicht, je nachdem, ob die Option **Eingebbar** aktiv ist.

Im Designmodus erscheint der Typ der Datenquelle in der ersten Zeile der Spalte, z.B. `[Tabelle1]MeinFeld`.

Ist der definierte Ausdruck nicht korrekt, zeigt die Spalte der Listbox im Anwendungsmodus eine Fehlermeldung.

- **Standardwerte** (Array vom Typ Listbox): Klicken Sie auf die Schaltfläche **Bearbeiten**, um das Dialogfenster zum Eingeben der Standardwerte für die Spalte anzuzeigen. Diese Werte sind dann beim Ausführen des Formulars automatisch in der Array Variablen verfügbar, die dieser Spalte zugewiesen ist. Sie müssen die einzelnen Werte der Liste getrennt durch Zeilenschaltung eingeben und dann das Dialogfenster bestätigen. Weitere Informationen dazu finden Sie im Abschnitt **Standardwerte** unter **Eingabekontrollen**.
- **Datentyp** (Listbox vom Typ Auswahl und Collection/Entity-Selection): Damit definieren Sie die Art des Ausdrucks bzw. der Variablen, die der Spalte zugeordnet ist. Sie geben das anzuwendende Anzeigeformat an und können das Menü **Anzeigetyp** unter dem Thema „Anzeige“ aktualisieren. Geben Sie im Bereich Ausdruck ein Datenfeld ein, wird die Eigenschaft **Datentyp** nicht angezeigt; es wird das zum Feldtyp passende Anzeigeformat verwendet.
- **Auswahlliste**: Damit können Sie der Spalte eine Auswahlliste zuordnen. Dann können Benutzer die Werte dieser Liste (angezeigt über ein PopUp-Menü) zum Ändern der Werte der Spalte und des zugewiesenen Arrays verwenden:

Vorname	Nachname	Position
Michael	Meier	Administrator
Alexander	Baumann	Administrator
Franz	Braun	Angestellte
Susi	Meier	Lager
Birgitt	Vogel	Vertrieb
Annette	Schwab	Marketing
Michaela	Pesch	Buchführung
Anne	Krimm	Buchführung
Klaus	Ben	Angestellte
Richard	Beck	Angestellte

Sie können den Wert weiterhin über Tastatur eingeben. Sollen Werte nicht durch ein Auswahlménü änderbar sein, wählen Sie eine zwingende Liste (siehe unten).

Hinweise:

- Bei einer Auswahlliste für eine numerische Spalte müssen Sie in der Liste das Zeichen "." als Dezimaltrenner verwenden. Es wird automatisch in einen gängigen Dezimaltrenner konvertiert und die anderen Symbole ("," "\$" etc.) werden im PopUp-Menü ignoriert.
- Ist die angegebene Liste hierarchisch, werden nur die Einträge der ersten Ebenen berücksichtigt.
- **Sichern als Wert/Referenz**: Bei einer Spalte mit zugewiesener Auswahlliste und Liste mit erforderlichen Werten kann diese Eigenschaft angegeben, wie der Inhalt im Feld oder der Variable gespeichert werden soll. Weitere Information dazu finden Sie im Abschnitt **Als Wert oder Referenz sichern**.

Gruppe Koordinaten & Größe

Hier können Sie die Breite der Spalte einstellen.

- **Breite**: Standardmäßige Spaltenbreite (in Pixel). Dieser Wert ändert sich, wenn Sie im Formulareditor den Spaltentrenner mit der Maus bewegen. Ist die Eigenschaft Vergrößerbar aktiv, kann der Benutzer die Spalte auch manuell anpassen.
- **Automatische Zeilenhöhe (*)**: Ist diese Eigenschaft für eine Spalte markiert, berechnet 4D die Höhe jeder Zeile automatisch gemäß dem Spalteninhalt. Diese Eigenschaft lässt sich auch auf Listboxebene definieren. Ist "Automatische Zeilenhöhe" auf Listboxebene markiert, bedeutet Deaktivieren dieser Option auf Spaltenebene, dass der Spalteninhalt nicht zum Berechnen der Zeilenhöhe verwendet werden darf. Weitere Informationen dazu finden Sie im unter **Automatische Zeilenhöhe** im Abschnitt **Eigenschaften für Listboxen**.
- (*)**Nur 4D View Pro**: Für dieses Feature ist eine 4D View Pro Lizenz erforderlich. Weitere Informationen dazu finden Sie im **4D View Pro Handbuch**.
- **Minimale Breite**: Mindestbreite der Spalte (in Pixel). Die Spalte lässt sich beim Anpassen der Spalte oder des Formulars nicht unter diesen Wert verkleinern.
- **Maximale Breite**: Maximale Breite der Spalte (in Pixel). Die Spalte lässt sich beim Anpassen der Spalte oder des Formulars nicht über diesen Wert hinaus vergrößern.

Hinweis: Wird das Formular angepasst, und ist in der Listbox die Eigenschaft **Maximiere horizontale Anpassung** aktiv, wird die rechts außen liegende Spalte – sofern erforderlich – über die maximale Breite hinaus erweitert.

Gruppe Vergrößerungsoptionen

Diese Gruppe enthält nur die Option **Vergrößerbar**. Sie ist standardmäßig markiert. Der Benutzer kann dann durch Bewegen der Seiten im Kopfteil die Spalte in der Größe verändern.

Gruppe Eingabe

Diese Gruppe enthält die Eigenschaften zur Dateneingabe in die Spalte der Listbox.

- **Eingebbar:** Diese Option ist standardmäßig markiert. Sie erlaubt die Eingabe in Spalten. Um den Wert einer Zelle zu ändern, muss der Benutzer auf den Wert doppelklicken. Ist diese Eigenschaft inaktiv, sind alle PopUp-Menüs, die der Spalte über eine Liste zugeordnet sind, d.h. über die Themen „Datenquelle“ und „Wertebereich“, deaktiviert.
Hinweis: Weitere Informationen dazu finden Sie im Abschnitt **Eingabe verwalten** im Handbuch 4D Programmiersprache.
- **Eingabefilter:** Weist den Zellen der Spalte einen Eingabefilter zu. Diese Eigenschaft ist nicht verfügbar, wenn die Eigenschaft **Eingebbar** inaktiv ist.

Gruppe Wertebereich

Hier bestimmen Sie Listen zum Steuern der Dateneingabe in Spalten der Listbox.

- **Erforderlich:** Hier definieren Sie eine Liste, welche die Werte zum Einfügen in die Spalte vorschreibt. Die Werte der Liste sind über ein PopUp-Menü verfügbar, das jeder Zelle zugewiesen ist. Weitere Informationen dazu finden Sie im Abschnitt "Auswahlliste" unter der oben beschriebenen Gruppe **Datenquelle**. Bei der zwingenden Liste ist im Gegensatz zur Auswahlliste keine Eingabe über Tastatur möglich. Sie können nur einen Wert aus der Liste wählen. Sind verschiedene Listen mit den Eigenschaften **Ausgenommen** und **Erforderlich** angelegt, ist **Erforderlich** vorrangig.
- **Ausgenommen:** Hier definieren Sie eine Liste, deren Werte nicht in die Spalte eingegeben werden können. Versuchen Sie, einen ausgeschlossenen Wert einzugeben, wird er abgewiesen und eine Fehlermeldung angezeigt.

Hinweis: Ist die angegebene Liste hierarchisch, werden nur die Elemente der obersten Ebene berücksichtigt.

Gruppe Anzeige

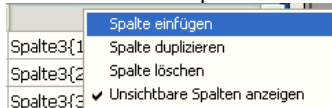
Hier bestimmen Sie das Anzeigeformat der Werte in den Spalten und die Eigenschaft **Unsichtbar**. Die Optionen variieren, je nach dem Variablentyp, der unter der Gruppe **Objekte** definiert wurde.

Hinweis: Wird die Listbox im hierarchischen Modus angezeigt, werden alle Eigenschaften dieser Gruppe für die erste Spalten deaktiviert.

- **Format Typ:** Hier weisen Sie den Spaltendaten ein Anzeigeformat zu. Die verfügbaren Formate richten sich nach dem Variablentyp (Listbox vom Typ Array) oder dem Daten/Feldtyp (Listbox vom Typ Auswahl). 4D bietet als Standardformate Alpha, Numerisch, Datum, Uhrzeit, Bild und Boolean. Der Typ Text hat keine spezifische Formatierung. Auch selbst erstellte Formate sind verfügbar.
- Arrays vom Typ Boolean lassen sich als Kontrollkästchen oder PopUp-Menüs anzeigen. Wählen Sie die Option **Ankreuzfeld**, können Sie über die Eigenschaft **Titel** eine Bezeichnung für das Kontrollkästchen eintragen. Wählen Sie die Option **Optionsfeld**, erscheinen die Eigenschaften **Text wenn wahr** und **Text wenn falsch**. Hier können Sie beide Bezeichnungen für das PopUp-Menü eintragen.
- Spalten mit Werten vom Typ Zahl lassen sich als Kontrollkästchen mit drei Zuständen anzeigen. Dazu wählen Sie in der Eigenschaftenliste unter dem Thema **Anzeige** die Option **Drei Zustände**. Es gibt folgende Werte:
 - 0 = nicht markiertes Kästchen
 - 1 = markiertes Kästchen
 - 2 = (oder jeder Wert >0) halb-markiertes Kästchen (3. Zustand). Für die Dateneingabe gibt dieser Status den Wert 2 zurück.
 - -1 = unsichtbares Kästchen
 - -2 = nicht markiertes Kästchen, nicht eingebbar
 - -3 = markiertes Kästchen, nicht eingebbar
 - -4 = halb-markiertes Kästchen, nicht eingebbar

In diesem Fall erscheint auch die Eigenschaft **Titel**, so dass die Bezeichnung des Kontrollkästchens eingegeben werden kann.

- Über die Eigenschaft **Unsichtbar** können Sie die Spalte in der Anwendungsumgebung ausblenden. In der Designumgebung klicken Sie in der Listbox auf die Spalte bzw. den Spaltentitel, um das Kontextmenü zu öffnen. Hier können Sie **Spalten ein-** oder **ausblenden:**



- **Zeilenbruch** (nur Spalten vom Typ Text): verwaltet die Anzeige des Spalteninhalts, wenn er die Spaltenbreite überschreitet. Ist diese Option markiert, bricht der Spaltentext automatisch in die nächste Zeile um, wenn er breiter als die Spalte ist und die Höhe ausreicht. Ist die Option nicht markiert, wird zu langer Text abgeschnitten und durch drei Punkte (...) ersetzt. Im folgenden Beispiel ist **Zeilenbruch** für die linke Spalte markiert, für die rechte nicht:

Kopfzeile	Kopfzeile
Für Listboxen vom Typ Array geben Sie den Namen eines Array Lange Ganzzahl ein. Jedes Element dieses Array entspricht einer Zelle der Spalte. Deshalb	Für Listboxen vom Typ Array geben Sie...
In 4D v14 haben Sie eine weitere Möglichkeit, um Hintergrundfarben, Schriftstile und -farben in Listboxen zu setzen: Sie können Arrays oder	In 4D v14 haben Sie eine weitere Möglic... Q1

Beachten Sie, dass die Zeilenhöhe unverändert bleibt, egal, ob die Option markiert ist oder nicht. Lässt sich der Text mit Zeilenumbrüchen in der Spalte nicht komplett anzeigen, wird er abgeschnitten (ohne ...). Haben die Spalten in der Listbox nur eine Zeile, erscheint nur die 1. Zeile des Textes:

Kopfzeile	Kopfzeile
Für Listboxen vom Typ Array	Für Listboxen vom Typ Array gebe...
In 4D v14 haben Sie eine weitere	In 4D v14 haben Sie eine weitere Mö... Q1

- **Unsichtbar als Standard** verwaltet generisch die Anzeige des Objekts beim Laden des Formulars (siehe Abschnitt **Unsichtbar als Standard**).
- **Abkürzen mit Auslassungspunkten** steuert die Anzeige der Werte, wenn die Spalte der Listbox für den ganzen Inhalt zu schmal ist. Diese Option ist für verschiedene Arten von Inhalt in Spalten verfügbar. Davon ausgenommen sind Bilder und Objekte.
 - Ist die Option **markiert** (Standard) und ist der Inhalt größer als die Breite der Spalte, wird er abgeschnitten und es erscheinen Auslassungspunkte:

Cities	Popu...
Charlotte	792862
New York	8406...
Philadelp...	1553...
San Fran...	837442
San Jose	288054

Hinweis: Die Position der Auslassungspunkte richtet sich nach dem Betriebssystem. Im obigen Beispiel (Windows) werden sie auf der rechten Seite des Textes angefügt. Unter OS X werden sie in der Mitte des Textes eingefügt.

- Ist die Option **nicht markiert** und ist der Inhalt größer als die Breite der Spalte, wird er nur abgeschnitten:

Cities	Popu...
Charlotte	792862
New York	8406000
Philadelphi	1553000
San Francis	837442
San Jose	288054

Die Option **Abkürzen mit Auslassungspunkten** ist standardmäßig markiert und lässt sich für Listboxen vom Typ Array oder Auswahl angeben.

Die Option **Abkürzen mit Auslassungspunkten** ist auch getrennt nach **Spalte** oder **Fußteil** innerhalb einer Listbox verfügbar. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Spalten der Listbox**.

Hinweise:

- Für Spalten vom Typ Text oder String ist die Option **Abkürzen mit Auslassungspunkten** nur verfügbar, wenn die Option **Zeilenumbruch nicht** markiert ist. Ist **Zeilenumbruch** markiert, wird über Zellen hinausgehender Inhalt bereits über diese Option verwaltet. Deshalb ist die Option **Abkürzen mit Auslassungspunkten** hier nicht verfügbar.
- Die Option **Abkürzen mit Auslassungspunkten** lässt sich auch für Spalten vom Typ Boolean anwenden. Das Ergebnis ist aber je nach Boolean Format unterschiedlich:
 - Für Boolean Formate vom Typ PopUp werden die Einträge mit Auslassungspunkten abgekürzt,
 - Für Boolean Formate vom Typ Optionsfeld werden die Bezeichnungen immer abgeschnitten.

Hintergrund und Rahmen

Diese Gruppe bietet Eigenschaften für die Hintergrundfarbe in Spaltenzellen.

- **Hintergrundfarbe:** Setzt die Hintergrundfarbe der Spalte. Standardmäßig ist **Automatisch** gewählt, d.h. die Spalte verwendet die Hintergrundfarbe, die für die Listbox definiert wurde.
- **Wechselnde Hintergrundfarbe:** Ermöglicht für Zeilen mit ungeraden Nummern eine andere Hintergrundfarbe zu setzen. Standardmäßig ist **Automatisch** gewählt, d.h. die Spalte verwendet die wechselnde Hintergrundfarbe, die für die Listbox definiert wurde.
- **Zeilen Hintergrundfarbe Array** (Listboxen vom Typ Array) / **Hintergrundfarbe Ausdruck** (Listboxen vom Typ Auswahl, Collection, Entity-Selection) wendet für jede Zelle der Spalte eine eigene Hintergrundfarbe an. Es werden RGB Farbwerte verwendet.
 - Für Listboxen vom Typ Array geben Sie den Namen eines Array Lange Ganzzahl ein. Jedes Element dieses Array entspricht einer Zelle der Spalte. Deshalb muss es dieselbe Größe haben wie das Array, das der Spalte zugeordnet ist. Sie können Konstanten unter dem Thema **SET RGB COLORS** verwenden. Soll die Zelle die auf höherer Ebene definierte Farbe erben, übergeben Sie im dazugehörigen Array Element den Wert -255. Weitere Informationen dazu finden Sie im Abschnitt **Vererbung**.
 - Für Listboxen vom Typ Auswahl und Collection oder Entity-Selection geben Sie einen Ausdruck oder eine Variable (außer vom Typ Array) ein. Der Ausdruck oder die Variable werden dann für jede angezeigte Zelle bewertet. Einen Ausdruck können Sie über den Formeleditor definieren. Dazu klicken Sie auf die Schaltfläche [...]. Sie erscheint, wenn Sie den Bereich auswählen. Sie können Konstanten unter dem Thema **SET RGB COLORS** verwenden.

Hinweis: Für Listboxen vom Typ Collection oder Entity-Selection lässt sich diese Eigenschaft auch mit der Option **Meta Info Ausdruck** setzen (siehe **Gruppe Text**).

Gruppe Text

- **Zeilenstil Array** (Listboxen vom Typ Array) / **Zeilenstil Ausdruck** (Listboxen vom Typ Auswahl, Collection, Entity-Selection) wendet einen eigenen Schriftstil für jede Zelle einer Spalte an.
 - Für Listboxen vom Typ Array geben Sie den Namen eines Array Länge Ganzzahl ein. Da jedes Element dieses Array einer Zelle der Spalte entspricht, muss das Array dieselbe Größe haben wie das der Spalte zugewiesene Array. Zum Füllen des Array (über eine Methode) verwenden Sie Konstanten unter dem Thema **Schriftstile**. Sie können auch mehrere Konstanten miteinander kombinieren. Soll die Zelle den auf höherer Ebene definierten Stil erben, übergeben Sie im dazugehörigen Array Element den Wert -255. Weitere Informationen dazu finden Sie im Abschnitt **Vererbung**.
 - Für Listboxen vom Typ Auswahl, Collection, Entity-Selection geben Sie einen Ausdruck oder eine Variable (außer vom Typ Array) ein. Ausdruck oder Variable werden für jede angezeigte Zelle bewertet. Einen Ausdruck können Sie über den Formeleditor definieren. Dazu klicken Sie auf die Schaltfläche [...]. Sie erscheint, wenn Sie den Bereich auswählen. Sie können Konstanten unter dem Thema **Schriftstile** verwenden.
Hinweis: Für Listboxen vom Typ Collection oder Entity-Selection lässt sich diese Eigenschaft auch mit der Option **Meta Info Ausdruck** setzen (see **Gruppe Text**).
- **Zeilenschriftfarbe Array** (Listboxen vom Typ Array) / **Zeilenschriftfarbe Ausdruck** (Listboxen vom Typ Auswahl, Collection, Entity-Selection) wendet für jede Zelle der Spalte eine eigene Schriftfarbe an. Es werden RGB Farbwerte verwendet.
 - Für Listboxen vom Typ Array geben Sie den Namen eines Array Länge Ganzzahl ein. Jedes Element dieses Array entspricht einer Zelle der Spalte. Deshalb muss es dieselbe Größe haben wie das Array, das der Spalte zugeordnet ist. Sie können Konstanten unter dem Thema **SET RGB COLORS** verwenden. Soll die Zelle die auf höherer Ebene definierte Farbe erben, übergeben Sie im dazugehörigen Array Element den Wert -255. Weitere Informationen dazu finden Sie im Abschnitt **Vererbung**.
 - Für Listboxen vom Typ Auswahl, Collection, Entity-Selection geben Sie einen Ausdruck oder eine Variable (außer vom Typ Array) ein. Der Ausdruck oder die Variable werden dann für jede angezeigte Zelle bewertet. Einen Ausdruck können Sie über den Formeleditor definieren. Dazu klicken Sie auf die Schaltfläche [...]. Sie erscheint, wenn Sie den Bereich auswählen. Sie können Konstanten unter dem Thema **SET RGB COLORS** verwenden.
Hinweis: Für Listboxen vom Typ Collection oder Entity-Selection lässt sich diese Eigenschaft auch mit der Option **Meta Info Ausdruck** setzen (siehe **Gruppe Text**).

Eigenschaften für Spaltentitel der Listbox

Vorbemerkung: Damit die Eigenschaften für Kopfteile in der Listbox zugänglich sind, müssen Sie in der Eigenschaftsliste der Listbox die Option **Kopfteil anzeigen** markieren. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Listboxen**.

Ist das Objekt Listbox gewählt, können Sie im Formulareditor einen Spaltentitel durch Anklicken auswählen:

Vorname	Nachname	Position
Spalte1{1}	Spalte2{1}	Spalte3{1}
Spalte1{2}	Spalte2{2}	Spalte3{2}
Spalte1{3}	Spalte2{3}	Spalte3{3}

Sie können für jeden Spaltentitel der Listbox standardmäßige Texteingenschaften setzen; diese haben dann Vorrang vor den Eigenschaften der Spalte oder der Listbox selbst.

Darüberhinaus können Sie auf spezifische Eigenschaften zugreifen, die im folgenden beschrieben werden.

Gruppe Objekte


Hier definieren Sie die Eigenschaften für den Kopfteil.

- **Objektname:** Name des Kopfteils
- **Variablenname:** Name der Variablen, die dem Kopfteil zugeordnet ist. Über diese Variable vom Typ Zahl können Sie die Sortierung und die Sortierrichtung per Programmierung steuern. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Einführung in Listboxen**.
- **Titel:** Bezeichnung, die im Kopfteil der Spalte erscheint.

Hinweis: Sie können das Zeichen \ (backslash) für einen Zeilenumbruch in der Bezeichnung des Titels verwenden. Für das Zeichen \ in der Bezeichnung geben Sie \\ ein.

Gruppe Bild

Hier bestimmen Sie die Eigenschaften, um ein Bild im Spaltentitel anzuzeigen (optional). Sie können ein Icon neben dem Text des Spaltentitels oder anstelle der Spalte anzeigen, insbesondere beim Ausführen angepasster Sortierläufe.

 Vorname	Nachname
Michael	Meierr
Alexander	Baumann

- **Symbol:** Definiert die Bildquelle zum Einfügen in den Spaltentitel. Sie können, wie für Bildschaltflächen von 4D, ein Bild aus einer Variablen, einer Bildbibliothek, einer Ressourcen-Datei oder einer Datei verwenden.
- **Name/ID:** Hier setzen Sie das Bild aus der zuvor definierten Quelle. Die Eingabe in dieses Feld richtet sich nach der gewählten Quelle: Geben Sie den Namen ein, wenn das Bild eine Variable ist, die Nummer, wenn es aus der Bildbibliothek oder einer Ressourcendatei stammt, oder den Pfadnamen des Bildes, wenn es aus einer Bilddatei stammt; in diesem Fall muss der Pfad zum Ordner **Resources** der Datenbank passen. Weitere Informationen dazu finden Sie im Abschnitt **Automatisches Verweisen auf Bilder**.
- **Symbolposition:** Position des Symbols im Spaltentitel. Sie können es **Links** oder **Rechts** vom Spaltentitel anordnen.
- **Spiegeleffekt (Windows):** Setzt die Anzeige des Icon im Modus "Rechts nach Links". Weitere Informationen dazu finden Sie im Abschnitt **Spiegeleffekt (Windows)**.

Gruppe Hilfe

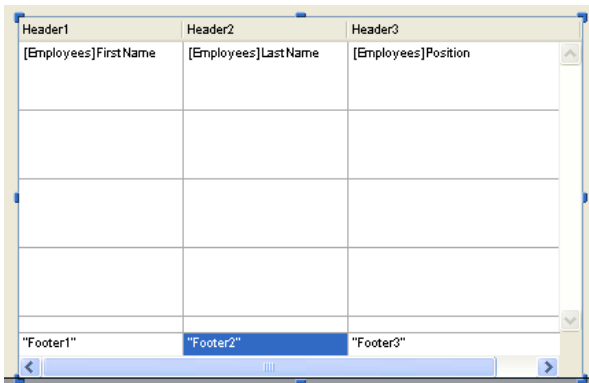
Jeder Kopfteil kann eine eigene Hilfmeldung haben. Weitere Informationen dazu finden Sie im Abschnitt **Hilfmeldungen**.

Eigenschaften für Fußteile der Listbox

Listboxen können nicht-eingebbare "Fußteile" mit zusätzlichen Informationen haben. Bei Daten in Tabellenformularen dienen Fußteile in der Regel zum Anzeigen von Berechnungen wie Gesamtsummen oder Durchschnittswerte.

Vorbemerkung: Um die Eigenschaften von Fußteilen nutzen zu können, müssen Sie in der Eigenschaftenliste für Listboxen die Option **Fußteil anzeigen** markieren. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Listboxen**.

Werden Fußteile angezeigt, können Sie darauf klicken, um einen Bereich auszuwählen, wenn im Formulareditor das Objekt Listbox ausgewählt ist.



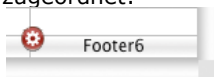
Für jeden Fußteil einer Spalte der Listbox können Sie standardmäßige Texteneigenschaften setzen. Diese haben dann Vorrang vor Eigenschaften, die für die Spalte oder die Listbox definiert wurden.

Sie können auch bestimmte Eigenschaften nutzen, die im folgenden näher erläutert werden.

Gruppe Objekte

Diese Gruppe enthält Eigenschaften zum Definieren von Fußteilen.

- **Objektname** und **Variablenname**: Der Fußteilbereich ist ein spezifisches Objekt mit eigenem Objektnamen (er muss auf der Seite einmalig sein) und zugewiesener Variable. Der Variablenname ist standardmäßig leer und 4D verwendet dynamische Variablen.
- **Variablentyp**: Dieses DropDown-Menü setzt den Variablentyp und aktualisiert die Optionen der Eigenschaftenliste. Beachten Sie, dass bei nicht-dynamischen Variablen (wenn Sie den Namen vergeben) die Typisierung über die Programmiersprache erfolgen muss.
- **Variable Berechnung**: Diese Option bestimmt die Art der Berechnung, die im Fußteil ausgeführt werden soll. Es gibt verschiedene Varianten sowie die Option **Eigene**:
 - **Minimum, Maximum, Summe, Anzahl, Durchschnitt, Standard Abweichung(*), Varianz(*)** und **Quadratwurzel (*)**. Die verschiedenen Berechnungsarten werden im nachfolgenden Abschnitt "Automatische Berechnung" beschrieben. Ist eine Berechnungsart ausgewählt, wird sie automatisch auf alle gefundenen Werte in der Spalte der Listbox angewandt. Beachten Sie, dass die Berechnung dabei nicht berücksichtigt, ob Zeilen in der Listbox ein- oder ausgeblendet sind. Soll eine Berechnung nur für sichtbare Zeilen gelten, müssen Sie eine eigene Berechnung verwenden.
Wurde dem Fußteilbereich eine automatische Berechnung zugewiesen, wird ihm das Zeichen "Standardaktion" zugeordnet:



- **Eigene**: Wählen Sie diese Option, führt 4D keine automatische Berechnung durch. Sie müssen dann den Wert der Variablen in diesem Bereich per Programmierung zuweisen.

Automatische Berechnung

Sie können dem Fußteilbereich verschiedene automatische Berechnungen zuweisen. Nachfolgende Tabelle zeigt, welche Berechnungen je nach Datentyp in einer Spalte möglich sind und gibt den von 4D zugewiesenen Typ an (wenn er nicht per Code typisiert wird):

	Zahl	Text	Datum	Zeit	Boolean	Bild	Variablentyp Fußteil
Minimum	X		X	X	X		wie Spaltentyp
Maximum	X		X	X	X		wie Spaltentyp
Summe	X			X	X		wie Spaltentyp
Anzahl	X	X	X	X	X	X	Lange Ganzzahl
Durchschnitt	X			X			Zahl
Standard Abweichung(*)	X			X			Zahl
Varianz(*)	X			X			Zahl
Quadratwurzel(*)	X			X			Zahl

(*) Nur für Listboxen vom Typ Array.

Hinweis: Sie können in Fußteilmustern keine automatischen Berechnungen zuweisen, die auf Formeln basieren. In solchen Fällen müssen Sie eigene Berechnungen durchführen.

Gruppe Hilfe

Jeder Fußteilmustern hat eine eigene Hilfemeldung (siehe Abschnitt [Hilfemeldungen](#)).

Hierarchische Listboxen erstellen und verwenden

Ab 4D v12 können Sie hierarchische Listboxen definieren und verwenden. Das ist eine Listbox, in welcher der **Inhalt der ersten Spalte** in hierarchischer Form erscheint. Diese Art von Darstellung ermöglicht die Präsentation von Information, die wiederholte bzw. hierarchisch gegliederte Werte enthält, z.B. Land/Bundesland/Stadt o.ä.

Nur Listboxen vom **Typ Array** können hierarchisch sein.

Hierarchische Listboxen sind eine besondere Darstellungsart von Daten. Sie verändern jedoch nicht die Struktur dieser Daten (die Arrays). Hierarchische Listboxen werden exakt wie normale Listboxen verwaltet.

Um eine hierarchische Listbox zu definieren, gibt es drei Möglichkeiten:

- Hierarchische Elemente manuell mit der Eigenschaftsliste des Formulareditors konfigurieren.
- Die Hierarchie visuell über das PopUp-Menü zur Listbox Verwaltung im Formulareditor erzeugen.
- Die 4D Befehle **LISTBOX SET HIERARCHY** und **LISTBOX GET HIERARCHY** verwenden. Sie werden im Handbuch *4D Programmiersprache* beschrieben.

Dieser Abschnitt beschreibt, wie Sie hierarchische Listboxen im 4D Formulareditor anlegen und wie sie während der Ausführung arbeiten. Weitere Informationen zur programmierten Verwaltung von hierarchischen Listboxen (Auswahl, Umbrüche, Formularereignisse *On Expand* und *On Collapse*, etc.) finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Hierarchische Listboxen verwalten**.

Hierarchie über die Eigenschaftsliste verwalten

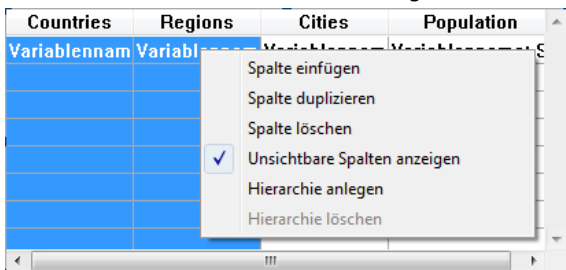
Sie können den hierarchischen Modus in der Eigenschaftsliste unter der Gruppe "Hierarchie" aktivieren und konfigurieren. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Listboxen**.

Hierarchie über das Kontextmenü verwalten

Klicken Sie auf den Spaltenbereich einer Listbox, enthält das Kontextmenü des Formulareditors die Einträge **Hierarchie anlegen** und **Hierarchie löschen**.

Hierarchie erstellen

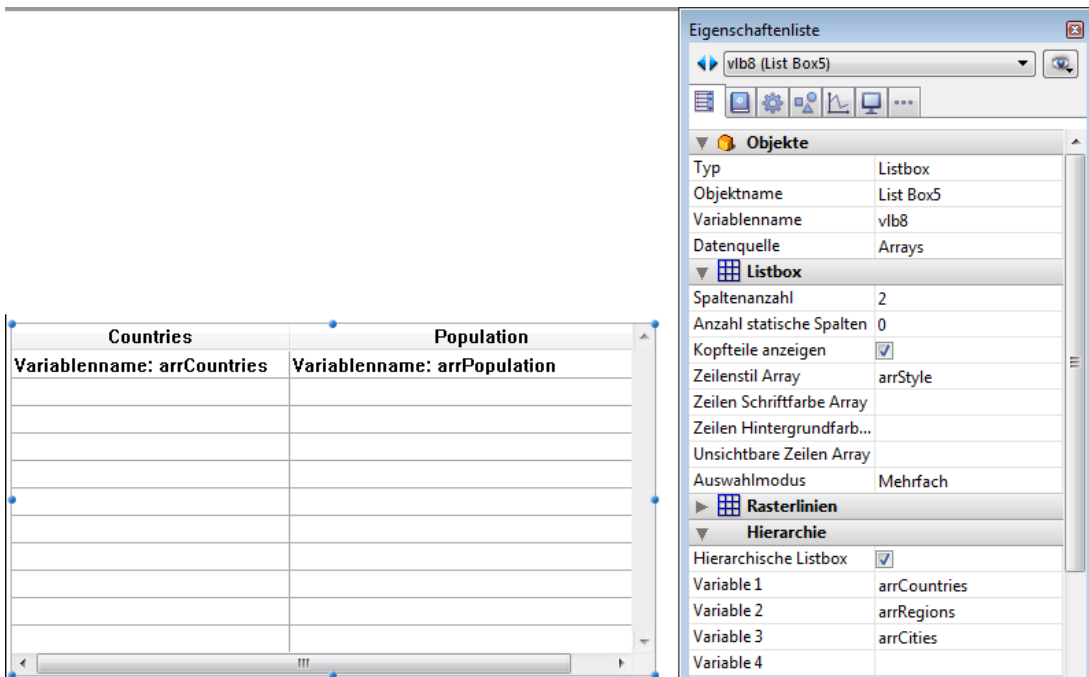
Wählen Sie im Formulareditor in einem Objekt Listbox vom Typ Array mindestens eine Spalte zusätzlich zur ersten aus, erscheint im Kontextmenü der Eintrag **Hierarchie anlegen**:



Markieren Sie diesen Eintrag, werden folgende Aktionen ausgeführt:

- In der Eigenschaftsliste ist die Option "Hierarchische Listbox" ist für das Objekt markiert.
- Die Variablen der Spalten werden zum Definieren der Hierarchie verwendet. Sie ersetzen alle bereits deklarierten Variablen.
- Die gewählten Spalten erscheinen nicht länger in der Listbox, mit Ausnahme der ersten.

Beispiel: Wir haben eine Listbox mit den Spalten *Countries*, *Regions*, *Cities* und *Population*. Sind *Countries*, *Regions* und *Cities* ausgewählt und markieren Sie im Kontextmenü den Eintrag **Hierarchie anlegen**, wird in der ersten Spalte eine dreistufige Hierarchie erstellt, Spalte 2 und 3 werden entfernt und die Spalte *Population* wird die 2. Spalte:



Hierarchie entfernen

Ist die 1. Spalte ausgewählt und bereits als Hierarchie definiert, können Sie den Eintrag **Hierarchie löschen** verwenden. Markieren Sie diesen Eintrag, werden folgende Aktionen ausgeführt:

- Die Option "Hierarchische Listbox" wird für das Objekt inaktiv,
- Die hierarchischen Ebenen 2 bis X werden entfernt und wieder in Spalten in der Listbox umgewandelt.

Funktionsweise

Wird ein Formular mit hierarchischer Listbox zum ersten Mal geöffnet, werden standardmäßig alle Zeilen aufgeklappt. Ein Umbruch und ein hierarchischer Knotenpunkt werden automatisch in der Listbox hinzugefügt, wenn Werte in den Arrays wiederholt werden.

Nehmen wir z.B. eine Listbox mit vier Arrays mit den Spalten "Countries", "Regions", "Cities" und "Population":

Countries	Regions	Cities	Population
Deutschland	Bayern	München	1.326.000
Deutschland	Bayern	Freising	45.000
Deutschland	Bayern	Rosenheim	60.000
Deutschland	Hessen	Frankfurt	675.000
Deutschland	Hessen	Darmstadt	138.000

Wird diese Listbox in hierarchischer Form angezeigt (die drei ersten Arrays sind in der Hierarchie enthalten), erhalten Sie folgendes:

Countries	Population
Deutschland	
Bayern	
München	1.326.000
Freising	45.000
Rosenheim	60.000
Hessen	
Frankfurt	675.000
Darmstadt	138.000
Frankreich	
Normandie	
Caen	220.000
Deauville	4.000

Die Arrays werden vor dem Aufbau der Hierarchie nicht sortiert. Enthält ein Array z.B. die Daten AAABBAACC, lautet die Hierarchie:

- > A
- > B
- > A
- > C

Um einen hierarchischen Knotenpunkt auf- oder zuzuklappen, klicken Sie darauf. Klicken Sie mit der Tastenkombination **Alt+Klick** unter Windows, bzw. **Wahltaste+Klick** auf Mac OS auf den Knoten, werden ebenfalls alle Unterelemente auf- oder zugeklappt.

Sortierung verwalten

Bei einer Listbox im hierarchischen Modus wird die Standardsortierung, d.h. wenn Sie auf den Titel einer Spalte der Listbox klicken, immer wie folgt aufgebaut:

- Zuerst werden alle Ebenen der hierarchischen Spalte, d.h. der ersten Spalte, in aufsteigender Reihenfolge sortiert.
- Die Sortierung wird dann, je nach der Benutzeraktion, auf die Werte der angeklickten Spalte in aufsteigender oder absteigender Reihenfolge ausgeführt.
- Alle Spalten werden aufeinander abgestimmt.
- Bei nachfolgenden Sortierungen, die in nicht-hierarchischen Spalten der Listbox ausgeführt werden, wird nur die letzte Ebene der ersten Spalte sortiert. Sie können die Sortierung dieser Spalte durch Klick in den Spaltentitel ändern.

Im folgenden sehen Sie ein weiteres Beispiel. Wir nehmen folgende Listbox, die keine spezifische Sortierung enthält:

Pays	Population
France	
Bretagne	
Rennes	200000
Quimper	80000
Brest	120000
Lannion	20300
Lorient	35000
Normandie	
Caen	220000
Deauville	4000
Cherbourg	41000
Auvergne	
Vichy	27000
Moulins	20600
Belgique	
Wallonie	
Namur	111000
Liège	200000
Flandre	
Anvers	472000
Louvain	95000
Bruxelles-Capitale	
Bruxelles	155000

Klicken Sie auf den Spaltentitel "Population", um die Einwohner in aufsteigender oder alternativ in absteigender Reihenfolge zu sortieren. Auf dem Bildschirm erscheint folgendes Ergebnis:

Pays	Population
Belgique	
Bruxelles-Capitale	
Bruxelles	155000
Flandre	
Anvers	472000
Louvain	95000
Wallonie	
Liège	200000
Namur	111000
France	
Auvergne	
Vichy	27000
Moulins	20600
Bretagne	
Rennes	200000
Brest	120000
Quimper	80000
Lorient	35000
Lannion	20300
Normandie	
Caen	220000
Cherbourg	41000
Deauville	4000

1- Automatische Sortierung aller hierarchischen Ebenen in aufsteigender Reihenfolge

Bei nachfolgenden Klicks auf den Spaltentitel Population wird nur die letzte Ebene synchronisiert

Klick auf Spaltentitel: Sortierung nach Population (aufsteigend)

2 - Sortierung der Population in aufsteigender Reihenfolge innerhalb der letzten Ebene

Sie können, wie für alle Listboxen, die standardmäßig vorgegebene Sortierung durch Deaktivieren der Option "Sortierbar" für die Listbox ausschalten und die Sortierung per Programmierung verwalten.

Datum und Zeit anzeigen

Sind in einer hierarchischen Listbox Werte vom Typ Datum oder Zeit enthalten, erscheinen sie in einem Standardformat:

- Datum wird im Format *systemabhängig kurz* angezeigt, z.B. für den 30. Mai 2009 auf einem amerikanischen System "05/30/09", auf einem europäischen System "30.05.09".
- Zeit wird ebenfalls im Format *systemabhängig kurz* angezeigt, d.h. je nach den Einstellungen des Systems "12:15:30" oder "12:15".

Standardaktionen verwenden

Sie können Listboxen vom Typ Auswahl über **Standardaktionen** verwalten, die Schaltflächen oder Menüs zugewiesen sind. Sie können beispielsweise die Standardaktion **Neu im Unterformular** verwenden, um über eine Listbox einen neuen Datensatz in der Tabelle hinzuzufügen. Auf diese Weise können Sie ohne große Mühe moderne Oberflächen gestalten, die auf Listboxen basieren

Für Listboxen sind drei Standardaktionen verfügbar: **Neu im Unterformular**, **Öffnen im Unterformular** und **Auswahl löschen**.

Neu im Unterformular

Gibt es mindestens eine Listbox vom Typ Auswahl mit Fokus im Formular, wird automatisch die Schaltfläche bzw. der Menübefehl zur Standardaktion **Neu im Unterformular** aktiviert.

Klickt der Benutzer auf die Schaltfläche oder den Menüeintrag, erscheint im Detailformular für die Listbox ein leerer Datensatz (siehe "Detailformular" unter **Gruppe Listbox**). Gibt er Werte ein und bestätigt den Datensatz, erscheint automatisch ein neuer leerer Datensatz. Das passiert solange, bis der Benutzer auf die Schaltfläche **Abbrechen** klickt. Ist die aktuelle Auswahl die Datenquelle der Listbox, erscheinen alle angelegten Datensätze in der Liste.

Öffnen im Unterformular

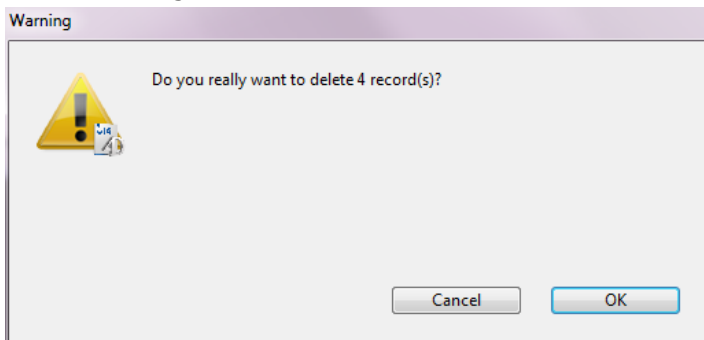
Ist im Formular mindestens eine Zeile in der Listbox vom Typ Auswahl markiert, wird automatisch die Schaltfläche bzw. der Menübefehl zur Standardaktion **Öffnen im Unterformular** aktiviert. Sind mehrere Zeilen markiert, wird die Aktion auf die zuletzt in die Auswahl eingefügte Zeile angewandt.

Klickt der Benutzer auf die Schaltfläche oder den Menüeintrag, erscheint im Detailformular für die Listbox der entsprechende Datensatz (siehe "Detailformular" unter **Gruppe Listbox**). Der Benutzer kann die Werte ändern, dann das Formular bestätigen oder abbrechen, um zur Listbox zurückzukehren.

Auswahl löschen

Ist im Formular mindestens eine Zeile in der Listbox vom Typ Auswahl markiert, wird automatisch die Schaltfläche bzw. der Menübefehl zur Standardaktion **Auswahl löschen** aktiviert. Sind mehrere Zeilen markiert, wird die Aktion auf alle Datensätze angewandt.

Klickt der Benutzer auf die Schaltfläche oder den Menüeintrag, erscheint ein Bestätigen-Dialog, in dem der Benutzer das Löschen bestätigen oder abbrechen kann.



📄 Anzeige der Felder in Listboxen

Sie können Spalten von Listboxen Datenfelder aus der Haupttabelle und/oder Datenfelder aus verschiedenen Tabellen zuweisen. Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Listboxen**.

Der Inhalt von Listboxen basiert jedoch in jedem Fall auf der aktuellen bzw. temporären Auswahl der Haupttabelle der Listbox:

- Verwenden Sie nur Datenfelder der Haupttabelle, richtet sich der Inhalt der Listbox einfach nach der Datenfeldauswahl der Haupttabelle.
- Verwenden Sie Datenfelder, die nicht zur Haupttabelle gehören, müssen diese "fremden" Tabellen mit einer Viele-zu-Eine Verknüpfung mit der Haupttabelle verbunden werden, denn sonst erscheinen die "fremden" Felder leer. Für jeden Datensatz in der Auswahl der Haupttabelle werden automatische Verknüpfungen aktiviert und die Listbox zeigt die entsprechenden Daten in den verknüpften Datenfeldern.
Bei manuellen Verknüpfungen müssen Sie die Aktivierung der Verknüpfungen programmieren, damit die Daten in der Listbox angezeigt werden.

Sind die Spalten der Listbox aufgrund inkonsistenter Definition leer, erscheint im Anwendungsmodus in jeder inkorrekten Spalte eine Fehlermeldung.

Wir erläutern die verschiedenen Fälle an einem Beispiel.

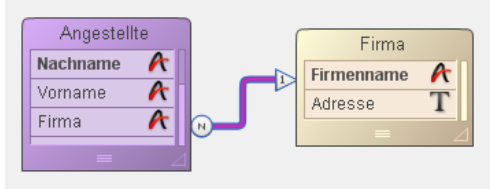
Ausgangsbasis sind die beiden Tabellen: [Firma] und [Angestellte].

- Die aktuelle Auswahl für die Tabelle [Firma] ist:
Firmenname
Verlag Encyclopedia
Compu und Soft
Reisebörse
- Die aktuelle Auswahl für die Tabelle [Angestellte] ist:

Nachname	Vorname	Firma
Schmidt	Carla	Reisebörse
Schwarz	Andreas	Compu und Soft
Lang	Vinzenz	Reisebörse
Daunert	Oliver	Reisebörse
Galvez	Sylvia	Compu und Soft
Schreiber	Robert	Verlag Encyclopedia
Müller	Arno	Reisebörse
Jones	Elisabeth	Verlag Encyclopedia
Faust	Viola	Compu und Soft
Pratts	Peter	Compu und Soft

Das Datenfeld [Firma]Firmenname wird mit der ersten Spalte einer Listbox verknüpft. Die Datenfelder [Angestellte]Nachname und [Angestellte]Vorname werden mit den nächsten beiden Spalten verknüpft. Die Datenquelle der Listbox ist die aktuelle Auswahl.

- **Fall 1:** Beide Tabellen sind über eine automatische Verknüpfung verbunden



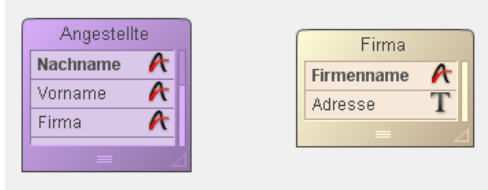
1) Die Haupttabelle der Listbox ist [Angestellte]. Die Listbox zeigt die aktuelle Auswahl der Tabelle [Angestellte] und aktiviert die automatische Verknüpfung, um für jeden Angestellten den dazugehörigen Firmennamen zu zeigen:

Firma	Nachname	Vorname
Reisebörse	Schmidt	Carla
Compu und Soft	Schwarz	Andreas
Reisebörse	Lang	Vinzenz
Reisebörse	Daunert	Oliver
Compu und Soft	Galvez	Sylvia
Verlag Encyclopedia	Schreiber	Robert
Reisebörse	Müller	Arno
Verlag Encyclopedia	Jones	Elisabeth
Compu und Soft	Faust	Viola
Compu und Soft	Pratts	Peter

2) Die Haupttabelle der ListBox ist [Firma]. Die ListBox zeigt die aktuelle Auswahl der Tabelle [Firma]. Da diese Auswahl nur drei Datensätze enthält, erscheinen in der ListBox drei Zeilen. Die Spalten für [Angestellte]Nachname und [Angestellte]Vorname sind leer:

Firma	Nachname	Vorname
Reisebörse		
Compu und Soft		
Verlag Encyclopedia		

- **Fall 2:** Die beiden Tabellen sind nicht miteinander verknüpft bzw. nur über eine manuelle Verknüpfung.



1) Die Haupttabelle der ListBox ist [Angestellte]. Die ListBox zeigt die aktuelle Auswahl der Tabelle [Angestellte]. Die Spalte des Datenfeldes [Firma] ist leer:

Firma	Nachname	Vorname
	Schmidt	Carla
	Schwarz	Andreas
	Lang	Vinzenz
	Dauert	Oliver
	Galvez	Sylvia
	Schreiber	Robert
	Müller	Arno
	Jones	Elisabeth
	Faust	Viola
	Pratts	Peter

2) Die Haupttabelle der ListBox ist [Firma]. Die ListBox zeigt die aktuelle Auswahl der Tabelle [Firma]. Da diese Auswahl nur drei Datensätze enthält, erscheinen in der ListBox nur drei Zeilen. Die Spalten für [Angestellte]Nachname und [Angestellte]Vorname sind leer:

Firma	Nachname	Vorname
Reisebörse		
Compu und Soft		
Verlag Encyclopedia		

Sie können die Auswahl der verschiedenen Tabellen natürlich auch per Programmierung verwalten und auf diese Weise Spalten anzeigen, die mit Feldern verknüpft sind, die nicht zur Haupttabelle gehören.

Ergebnis einer SQL-Anfrage in Listbox anzeigen

Ergebnis einer SQL-Anfrage in Listbox anzeigen

Sie können die Ergebnisse einer SQL-Anfrage direkt in ein Array vom Typ **Listbox** setzen. So können Sie sich rasch die Ergebnisse von SQL Anfragen ansehen. Es sind nur Suchläufe vom Typ **SELECT** möglich (siehe Handbuch *4D SQL Reference*). Dieser Mechanismus lässt sich nicht mit einer externen SQL Datenbank verwenden.

Dabei gilt folgendes:









- Sie erstellen die Listbox zum Empfangen des Suchergebnisses. Die Datenquelle der Listbox muss vom Typ **Array** sein.
- Führen Sie eine SQL-Suche vom Typ **SELECT** aus und weisen Sie das Ergebnis der mit der Listbox verknüpften Variablen zu. Sie können die 4D Tags **Begin SQL/End SQL** verwenden.
- Listbox-Spalten lassen sich vom Benutzer sortieren oder ändern.
- Bei erneuter Ausführung einer Anfrage mit **SELECT** werden die Spalten neu gefüllt. Es ist nicht möglich, dieselbe Listbox mit mehreren **SELECT** Anfragen nacheinander zu füllen.
- Es wird empfohlen, der Listbox die gleiche Anzahl Spalten zu geben wie im Ergebnis der SQL Anfrage sind. Gibt es weniger Listbox-Spalten als für die **SELECT** Anfrage erforderlich, werden automatisch Spalten hinzugefügt. Gibt es mehr Spalten als für die **SELECT** Anfrage erforderlich, werden die nicht benötigten Spalten automatisch ausgeblendet.
Hinweis: Den automatisch hinzugefügten Spalten werden Variablen vom Typ **Dynamische Variablen** zugewiesen, d.h. sie bestehen solange wie das Formular. Eine dynamische Variable wird auch für jeden Kopfteil erstellt. Beim Aufrufen des 4D Befehls **LISTBOX GET ARRAYS** enthält der Parameter *arrColVars* Zeiger auf die dynamischen Arrays und der Parameter *arrHeaderVars* Zeiger auf die dynamischen Kopfteilvariablen. Ist die hinzugefügte Spalte z.B. die fünfte Spalte, lautet ihr Name *sql_column5* und ihr Kopfteil *sql_header5*.
- Im interpretierten Modus lassen sich vorhandene Arrays, welche die Listbox verwendet, automatisch entsprechend der von der SQL-Anfrage gesendeten Daten erneut tippen.

Beispiel:


Wir wollen nach allen Datenfeldern der Tabelle [PEOPLE] suchen und den Inhalt mit der Variablen *vlistbox* in die Listbox setzen. Für die Objektmethode einer Schaltfläche (als eine Möglichkeit) schreiben Sie:

```
Begin SQL
SELECT * FROM PEOPLE INTO <<vlistbox>>
End SQL
```

Objekte der Bibliothek






























-  Überblick
-  Schaltflächen
-  Ablaufanzeigen
-  Eingabe Bereiche
-  Plug-Ins und Unterformulare
-  Statische Objekte
-  Registerkarten und Splitter
-  Widgets



























Über die vorkonfigurierte Objektbibliothek

Die Objektbibliothek enthält eine Sammlung vorkonfigurierter Objekte, die Sie einfach per Drag-and-Drop oder Copy-Paste in Ihre Formulare übernehmen können. Sie können die Objektbibliothek im Formulareditor über den Icon  am oberen Rand öffnen. Weitere Informationen dazu finden Sie im Abschnitt [Vorkonfigurierte Objektbibliothek verwenden](#).

Enthaltene Objekte

Nachfolgende Übersicht listet alle Objekte in der vorkonfigurierten Bibliothek. Klicken Sie auf den Objektnamen, erhalten Sie die ausführliche Beschreibung dazu.

Objekt	Icon	Beschreibung
Variable		Fügt eine eingebbare Variable vom Typ String hinzu.
Mehrfachstil Text		Fügt einen Textbereich mit Stilelementen hinzu (Variable vom Typ Text). Sie können die Schriftart, gröÙe und -farbe definieren.
Hierarchische Liste		Fügt eine hierarchische Liste hinzu mit Beispielcode in der Objektmethode für eine einfache mehrstufige Hierarchie.
Listbox		Fügt ein Objekt Listbox mit drei Spalten zum Anzeigen von Arrays mit Daten hinzu.
Rollbarer Bereich		Fügt einen rollbaren Bereich hinzu, bestehend aus einer einspaltigen Listbox ohne Kopf- und Fußteile.
Bild		Fügt einen konfigurierbaren Bildbereich hinzu. Über die Eigenschaftenliste lässt sich z.B. ein Kontextmenü oder ein Anzeigeformat hinzuzufügen.
Kennwort		Fügt eine Textvariable mit der Stilvorlage "Password" hinzu, welche die eingegebenen Zeichen als Sternchen anzeigt.
Suchbereich		Fügt einen Suchbereich mit anpassbarem Beispielcode in der Objektmethode hinzu.
Rich Text Bereich		Fügt einen Rich Text Bereich mit einem Satz Menüs und Schaltflächen zum Verwalten von Schriftstil und Referenzen hinzu.
4D Write Pro Area		Fügt einen vorkonfigurierten 4D Write Pro Bereich mit 4D Write Pro Widget mit verschiedenen Steuer-Panels zum Gestalten des Inhalts hinzu.
Schaltfläche		Fügt eine einfache Schaltfläche mit Variable hinzu. Ihre Aktion muss per Methode oder Standardaktion in der Eigenschaftenliste festgelegt werden.
Schaltfläche OK Abbrechen		Fügt ein Paar Abbrechen/OK Schaltflächen hinzu, jeweils mit den Standardaktionen Abbrechen und Bestätigen.
3D Schaltfläche		Fügt eine 3D Schaltfläche mit Basisfunktion hinzu.
Ankreuzfeld		Fügt ein Kontrollkästchen mit Basisfunktion hinzu.
Kleines Ankreuzfeld (nur Mac)		Fügt ein kleines Kontrollkästchen hinzu (nur auf Mac verfügbar).
Optionsfeld		Fügt ein Optionsfeld mit Basisfunktion hinzu.
Kleines Optionsfeld (nur Mac)		Fügt ein kleines Optionsfeld hinzu (nur auf Mac verfügbar).
Runde Schaltfläche		Fügt eine abgerundete 3D Schaltfläche mit vorformatierter Darstellung hinzu.
Eckige Schaltfläche (nur Mac)		Fügt eine eckige 3D Schaltfläche mit vorformatierter Darstellung hinzu.
Abgestufte Schaltfläche (nur Mac)		Fügt eine 3D Schaltfläche mit Graustufung mit vorformatierter Darstellung hinzu.
Schaltfläche mit 2 Farbstufen (nur Mac)		Fügt eine 3D Schaltfläche mit 2 Farbstufen mit vorformatierter Darstellung hinzu.
Toolbar Schaltfläche		Fügt eine 3D Toolbar Schaltfläche mit vorformatierter Darstellung hinzu.
XP Toolbar Schaltfläche		Fügt eine 3D XP Toolbar Schaltfläche mit vorformatierter Darstellung hinzu.
Runde Mac Schaltfläche		Fügt eine abgerundete Mac Schaltfläche hinzu (nicht verfügbar unter Windows).
Hilfe Schaltfläche		Fügt eine 3D Hilfe Schaltfläche mit vorformatierter Darstellung hinzu.
Schaltfläche Aufklappen-Zuklappen		Fügt eine 3D Aufklappen/Zuklappen Schaltfläche mit vorformatierter Darstellung hinzu.
Pop-up Menü		Fügt ein Pop-Up Menü mit zugeordneter Variable vom Typ Array hinzu, welche die meisten Datentypen darstellen kann.
Pop-Up Schaltfläche (nur Mac)		Fügt ein Pop-Up Menü mit auf Schaltfläche reduzierte Breite hinzu. (Nur auf Mac verfügbar)
Hierarchisches Pop-Up		Fügt ein hierarchisches Pop-Up Menü mit einer String Variablen vom Typ hierarchische Liste hinzu.
Horizontaler		Fügt einen horizontalen Ablaufbalken mit numerischer Variable mit den Standard

Ablaufbalken		Min und Max Werten 0 und 100 hinzu.
Vertikaler Ablaufbalken		Fügt einen vertikalen Ablaufbalken mit numerischer Variable mit den Standard Min und Max Werten 0 und 100 hinzu.
Animierter Balken (Barber Shop)		Fügt einen asynchronen Ablaufbalken mit numerischer Variable hinzu.
Drehendes Rad		Fügt ein asynchrones Ablaufrad mit numerischer Variable hinzu.
Horizontales Lineal		Fügt ein horizontales Lineal mit numerischer Variable mit den Standard Min und Max Werten 0 und 100 hinzu.
Vertikales Lineal		Fügt ein vertikales Lineal mit numerischer Variable mit den Standard Min und Max Werten 0 und 100 hinzu.
Horizontales Lineal mit Skala		Fügt ein abgestuftes horizontales Lineal mit numerischer Variable mit den Standard Min und Max Werten 0 und 100 hinzu.
Schrittschaltfläche		Fügt ein Lineal in Form einer Schaltfläche hinzu, deren Wert schrittweise erhöht oder verringert wird.
		Fügt eine Registerkarte mit dem Inhalt aus einem Array oder einer hierarchischen Liste hinzu.
		Fügt eine Variante der Registerkarte mit den gleichen Eigenschaften hinzu (Standardaktion "Goto Page", Inhalt aus Array oder hierarchischer Liste), jedoch Anzeige ohne Rahmen (nur auf Mac verfügbar)
		Fügt einen horizontalen Splitter hinzu, um das Formular oder einen Teil davon in zwei Bereiche zu unterteilen, die sich unabhängig vergrößern lassen.
		Fügt einen horizontalen Splitter hinzu, um das Formular oder einen Teil davon in zwei Bereiche zu unterteilen, die sich unabhängig vergrößern lassen.
		Fügt ein Viereck hinzu. Seine Eigenschaften (Farbe, Linienstärke, Muster, etc.) lassen sich über die Eigenschaftenliste oder per Programmierung festlegen.
		Fügt ein Etikett hinzu. Seine Eigenschaften (Farbe, Linienstärke, Muster, etc.) lassen sich über die Eigenschaftenliste oder per Programmierung festlegen.
		Fügt einen Gruppenrahmen hinzu. Seine Eigenschaften (Farbe, Linienstärke, Muster, etc.) lassen sich über die Eigenschaftenliste oder per Programmierung festlegen.
		Fügt ein Unterformular hinzu, um ein Formular aus einer anderen Struktur oder ein Projektformular einer Komponente anzuzeigen.
		Fügt einen 4D Write Bereich hinzu.
		Fügt einen 4D View Bereich hinzu.
		Fügt einen Web Bereich zum Anzeigen von HTML Seiten (lokal oder aus dem Web) hinzu.
		Fügt einen Plug-In Bereich zur Nutzung externer 4D Tools, wie 4D Write, 4D View, 4D Web Areas, oder von Drittherstellern entwickelte Tools hinzu.
Date Picker(*)		Fügt einen Kalender mit Datumsvariable hinzu. Sie kann konfiguriert werden, um Feiertage, Zeitspannen u.ä. anzuzeigen.
Pop-Up Datum(*)		Fügt einen Pop-Up Bereich hinzu, um einen Kalender zum Auswählen von Datum anzuzeigen. Das ausgewählte Datum wird direkt der verknüpften Datumsvariablen zugewiesen.
Datumseingabebereich(*)		Fügt einen Datums-Eingabebereich mit drei getrennten Feldern zum Eingeben von Tag, Monat und Jahr hinzu.
Time Picker		Fügt einen Pop-Up Bereich zur Zeiteingabe hinzu.
Zeiteingabebereich		Fügt einen Zeiteingabe-Bereich mit drei getrennten Feldern zum Eingeben von Stunde, Minute und Sekunde hinzu.
Zeitanzeigebereich		Fügt einen Zeitanzeige-Bereich (Uhr oder LCD) hinzu, der je nach zugewiesener Variable, eine statische Zeit oder eine laufende Uhr anzeigt

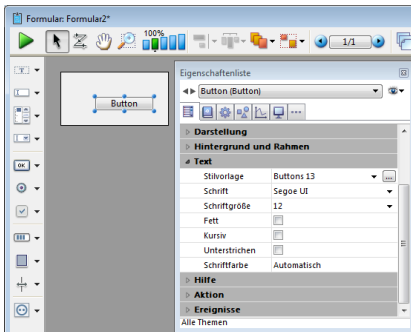
(*)Die Objekte Date Picker, Pop-up Datum und Datumseingabe-Bereich nutzen anfangs alle die gleiche Datumsvariable, und zeigen bei Verwenden im gleichen Formular die entsprechenden Datumsangaben

Schaltflächen

Schaltfläche



Fügt eine einfache Schaltfläche mit Variable hinzu. Die Aktion muss über eine Methode oder eine Standardaktion in der Eigenschaftensliste angegeben werden. Es ist auch eine Stilvorlage zugewiesen, die Sie bei Bedarf ändern können.

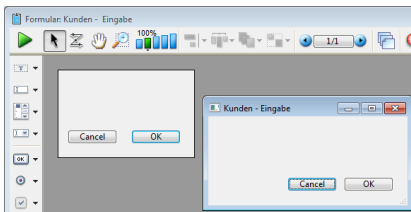


Weitere Informationen dazu finden Sie in den Abschnitten [Schaltflächen](#) und [Stilvorlagen](#).

Schaltfläche OK Abbrechen



Fügt das Schaltflächenpaar Abbrechen und OK hinzu mit den jeweiligen Standardaktionen Abbrechen und Bestätigen und dem Tastenkürzel "Return" für OK und "Escape" für Abbrechen. Sie sind nebeneinander angeordnet und bleiben beim Bewegen des Fensters in der unteren rechten Ecke.

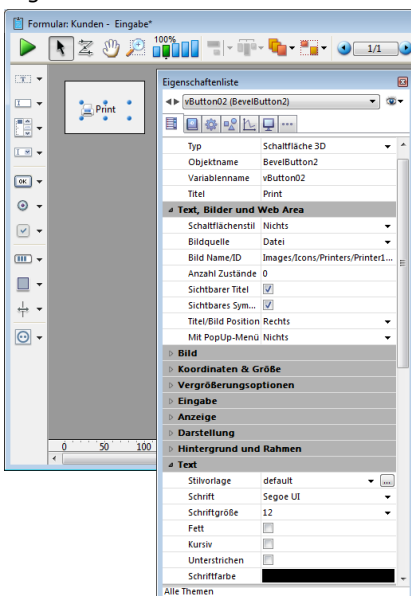


Weitere Informationen dazu finden Sie im Abschnitt [Schaltflächen](#).

3D Schaltfläche



Fügt eine einfache 3D Schaltfläche hinzu. Sie können die Eigenschaften über die Eigenschaftensliste anpassen.

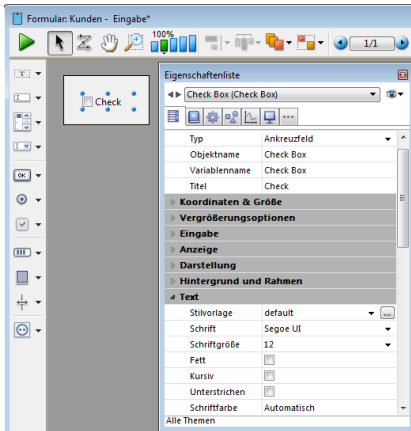


Weitere Informationen dazu finden Sie im Abschnitt [3D Schaltflächen](#), [3D Kontrollkästchen](#), [3D Optionsfelder](#).

Ankreuzfeld



Fügt ein einfaches Ankreuzfeld hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

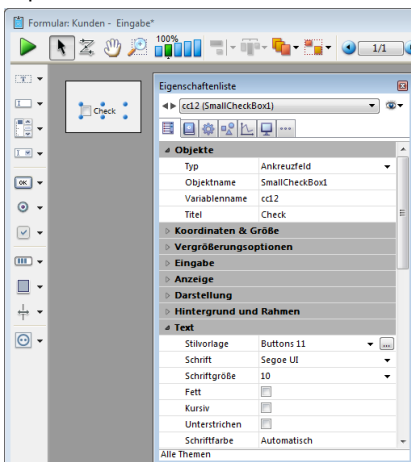


Weitere Informationen dazu finden Sie im Abschnitt **Kontrollkästchen**.

Kleines Ankreuzfeld (nur Mac)



Fügt ein kleines Ankreuzfeld hinzu (nur auf Mac verfügbar). Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

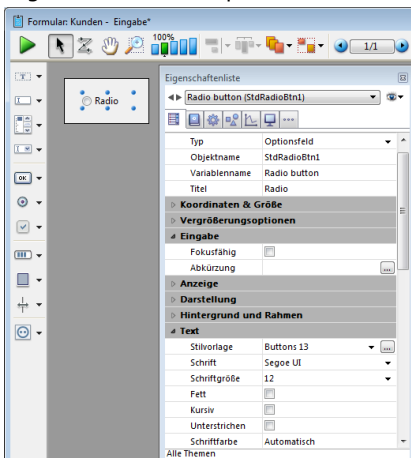


Weitere Informationen dazu finden Sie im Abschnitt **Kontrollkästchen**.

Optionsfeld



Fügt ein einfaches Optionsfeld hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

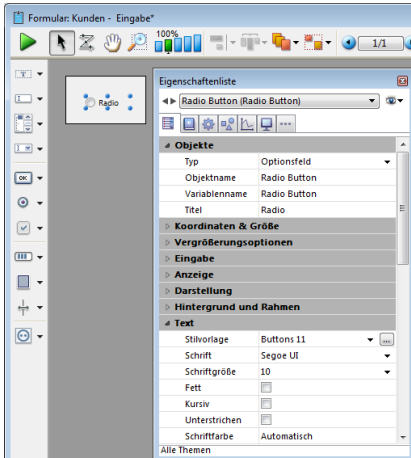


Weitere Informationen dazu finden Sie im Abschnitt **Optionsfelder und Optionsbilder**.

Kleines Optionsfeld (nur Mac)



Fügt ein kleines Optionsfeld hinzu (nur auf Mac verfügbar). Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

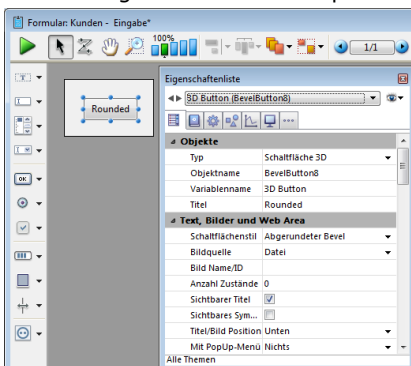


Weitere Informationen dazu finden Sie im Abschnitt [Optionsfelder und Optionsbilder](#).

Runde Schaltfläche



Fügt eine Schaltfläche mit abgerundeten Ecken mit vorformatierter 3D Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

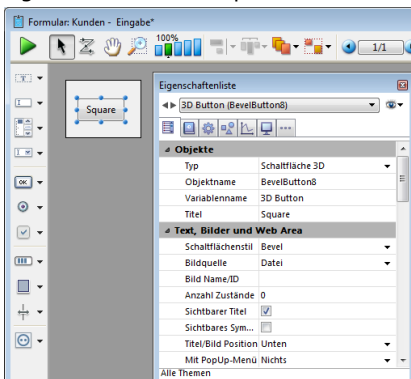


Weitere Informationen dazu finden Sie im Abschnitt [3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder](#).

Eckige Schaltfläche (nur Mac)



Fügt eine eckige 3D Schaltfläche mit vorformatierter Darstellung hinzu. Sie können die Eigenschaften in der Eigenschaftenliste anpassen.

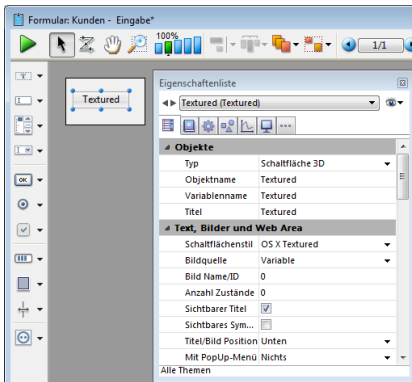


Weitere Informationen dazu finden Sie im Abschnitt [3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder](#).

Abgestufte Schaltfläche (nur Mac)



Fügt eine 3D Schaltfläche mit Graustufung mit vorformatierter Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

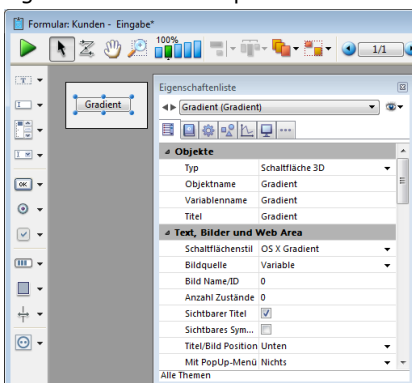


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen**, **3D Kontrollkästchen**, **3D Optionsfelder**.

Schaltfläche mit 2 Farbstufen (nur Mac)



Fügt eine 3D Schaltfläche mit 2 Farbstufen mit vorformatierter Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

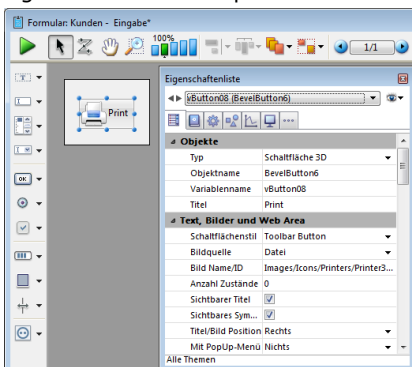


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen**, **3D Kontrollkästchen**, **3D Optionsfelder**.

Toolbar Schaltfläche



Fügt eine Toolbar Schaltfläche mit vorformatierter 3D Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

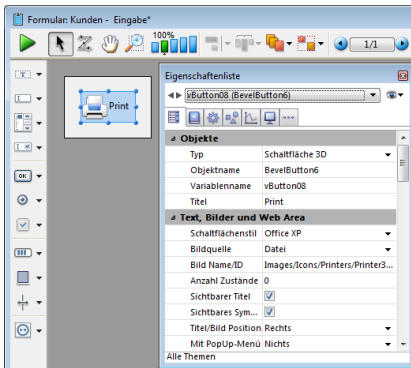


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen**, **3D Kontrollkästchen**, **3D Optionsfelder**.

XP Toolbar Schaltfläche



Fügt eine XP Toolbar Schaltfläche mit vorformatierter 3D Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

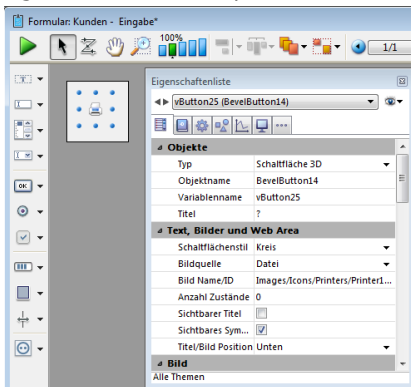


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder**.

Runde Mac Schaltfläche



Fügt eine runde Mac Schaltfläche hinzu (auf Windows nicht verfügbar). Sie können die Eigenschaften über die Eigenschaftensliste anpassen.

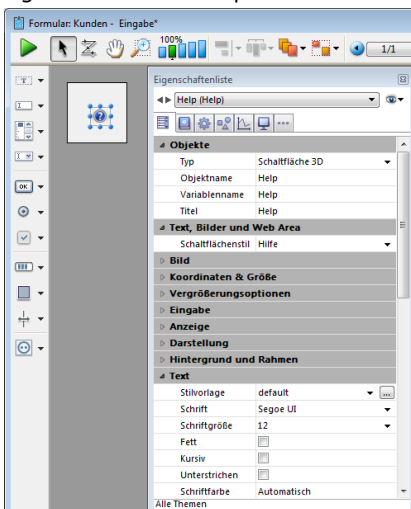


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder**.

Hilfe Schaltfläche



Fügt eine Hilfes Schaltfläche mit vorformatierter 3D Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftensliste anpassen.

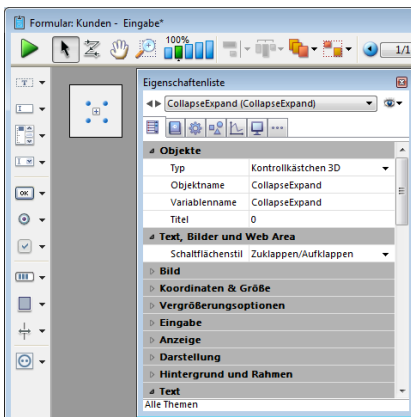


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder**.

Schaltfläche Aufklappen-Zuklappen



Fügt eine Schaltfläche Aufklappen-Zuklappen mit vorformatierter 3D Darstellung hinzu. Sie können die Eigenschaften über die Eigenschaftensliste anpassen.

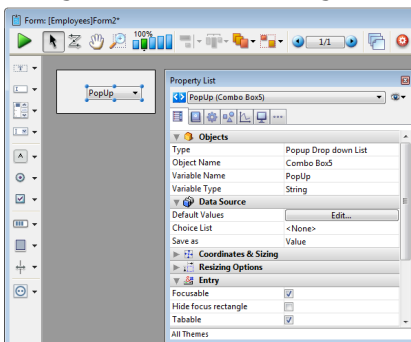


Weitere Informationen dazu finden Sie im Abschnitt **3D Schaltflächen, 3D Kontrollkästchen, 3D Optionsfelder**.

Pop-up Menü



Fügt ein Pop-Up Menü mit einer Variablen vom Typ Array hinzu, das die meisten Datentypen anzeigen kann. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

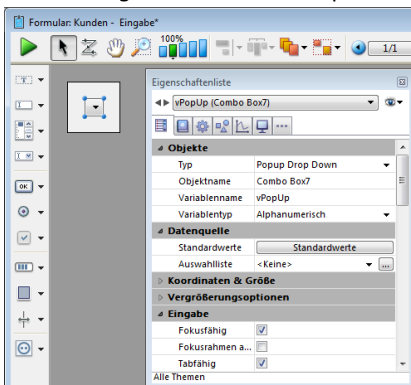


Weitere Informationen dazu finden Sie im Abschnitt **PopUp-Menüs, Dropdown-Listen**.

Pop-Up Schaltfläche (nur Mac)



Fügt ein Pop-Up Menü mit auf Schaltfläche reduzierte Breite hinzu (nur auf Mac verfügbar). Sie können die Eigenschaften über die Eigenschaftenliste anpassen.

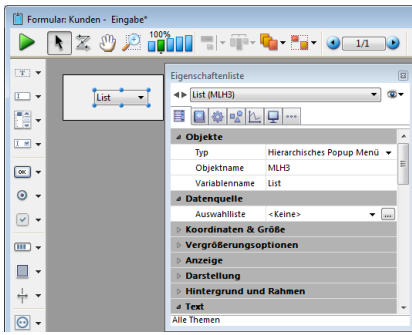


Weitere Informationen dazu finden Sie im Abschnitt **PopUp-Menüs, Dropdown-Listen**.

Hierarchisches Pop-Up



Fügt ein hierarchisches Pop-Up Menü mit einer String Variablen vom Typ hierarchische Liste hinzu. Sie können die Eigenschaften über die Eigenschaftenliste anpassen.



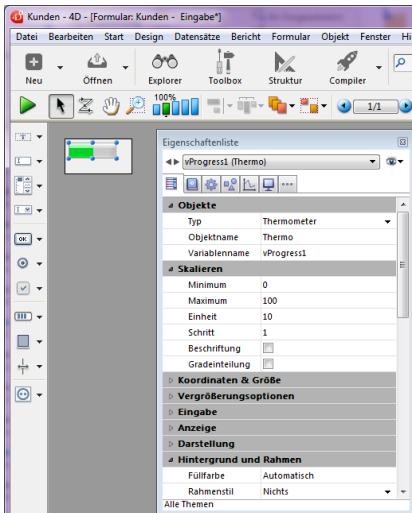
Weitere Informationen dazu finden Sie im Abschnitt **Hierarchische PopUp-Menüs und hierarchische Listen**.

Ablaufanzeigen

Horizontaler Ablaufbalken



Fügt einen horizontalen Ablaufbalken mit einer numerischen Variable hinzu mit dem jeweiligen Mindest- und Maximumwert 0 und 100. Sie können diese Werte über die Eigenschaftensliste verändern.

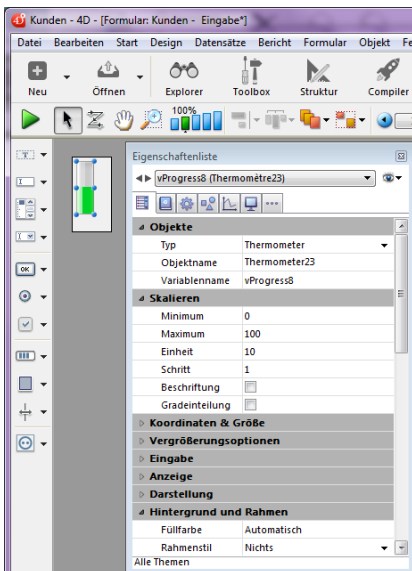


Weitere Informationen dazu finden Sie im Abschnitt [Indikatoren](#).

Vertikaler Ablaufbalken



Fügt einen vertikalen Ablaufbalken mit einer numerischen Variablen hinzu mit dem jeweiligen Mindest- und Maximumwert 0 und 100. Sie können diese Werte über die Eigenschaftensliste verändern.

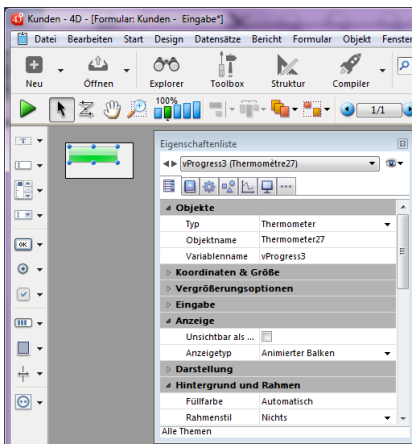


Weitere Informationen dazu finden Sie im Abschnitt [Indikatoren](#).

Animierter Balken (Barber Shop)



Fügt einen asynchronen Ablaufbalken mit einer numerischen Variable hinzu. Er wird animiert, wenn der Wert der Variablen größer als Null ist. Sie können seine Darstellung per Programmierung ändern.

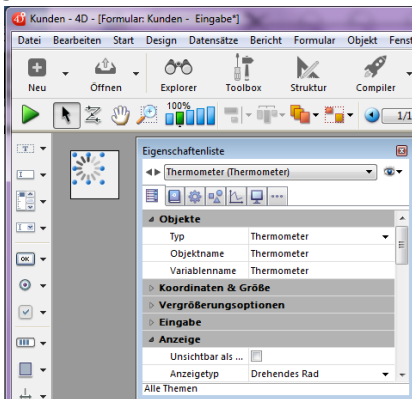


Weitere Informationen dazu finden Sie im Abschnitt **Indikatoren**.

Drehendes Rad



Fügt ein asynchrones Ablaufrad mit einer numerischen Variable hinzu. Es wird animiert, wenn der Wert der Variablen größer als Null ist.

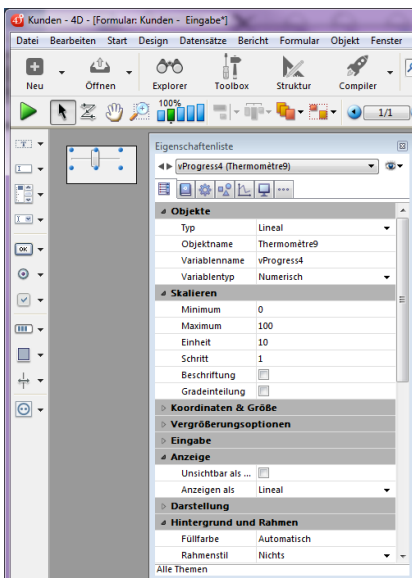


Weitere Informationen dazu finden Sie im Abschnitt **Indikatoren**.

Horizontales Lineal



Fügt ein horizontales Lineal mit einer numerischen Variable hinzu mit dem jeweiligen Mindest- und Maximumwert 0 und 100. Sie können diese Werte über die Eigenschaftenliste verändern.

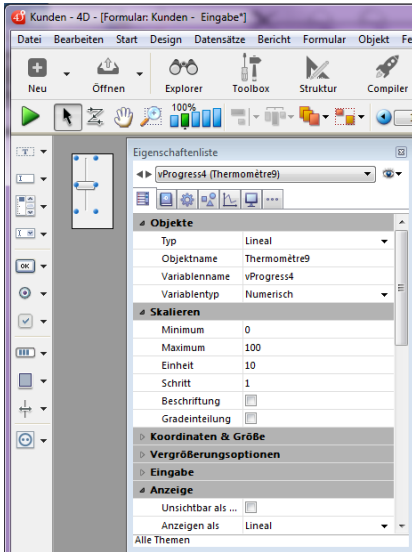


Weitere Informationen dazu finden Sie im Abschnitt **Indikatoren**.

Vertikales Lineal



Fügt ein vertikales Lineal mit numerischer Variable hinzu mit dem jeweiligen Mindest- und Maximumwert 0 und 100. Sie können diese Werte über die Eigenschaftensliste verändern.

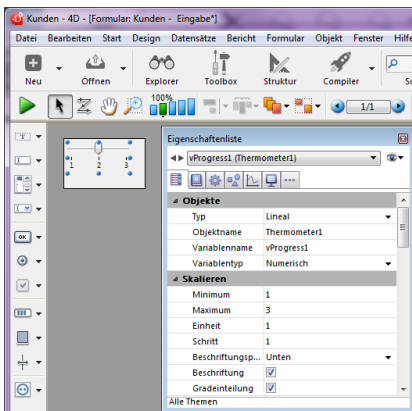


Weitere Informationen dazu finden Sie im Abschnitt **Indikatoren**.

Horizontales Lineal mit Skala



Fügt ein horizontales Lineal mit Skala und numerischer Variable hinzu mit dem jeweiligen Mindest- und Maximumwert 0 und 100. Sie können diese Werte über die Eigenschaftensliste verändern.

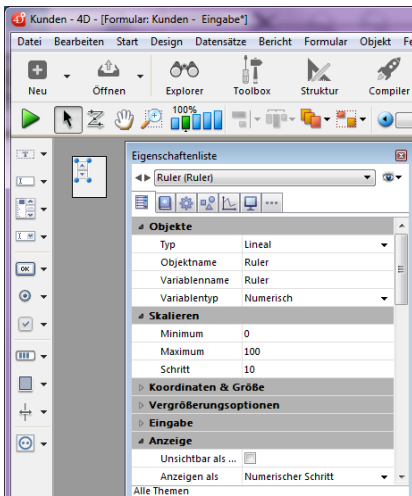


Weitere Informationen dazu finden Sie im Abschnitt **Indikatoren**.

Schrittschaltfläche



Fügt eine Schaltfläche mit Skala hinzu mit schrittweise auf- und absteigenden Werten. Sie können den jeweiligen Mindest- und Maximumwert, und die Schrittweite über die Eigenschaftensliste setzen.



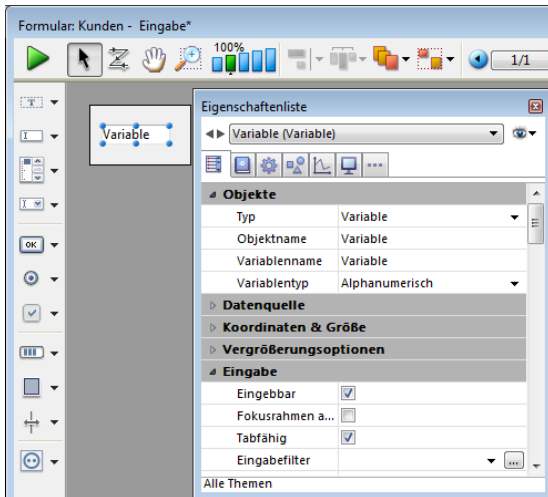
Weitere Informationen dazu finden Sie im Abschnitt **Indikatoren**.

Eingabe Bereiche

Variable



Fügt eine eingebare Variable vom Typ String hinzu. Sie können den Typ oder andere Eigenschaften über die Eigenschaftsliste ändern.

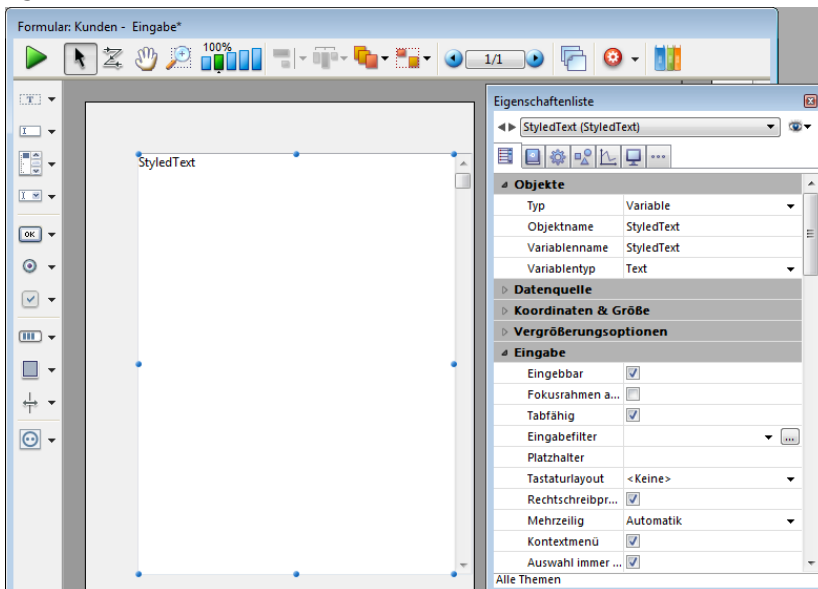


Weitere Informationen dazu finden Sie im Abschnitt [Get list item font](#).

Mehrfachstil Text



Fügt einen Textbereich mit Mehrfachstil (Variable) hinzu, wo Sie Schriftart, -stil und -farbe definieren können. Der Bereich ist vergrößerbar, hat einen vertikalen Rollbalken und ein Kontextmenü. Sie können diese Standardvorgaben über die Eigenschaftsliste ändern.

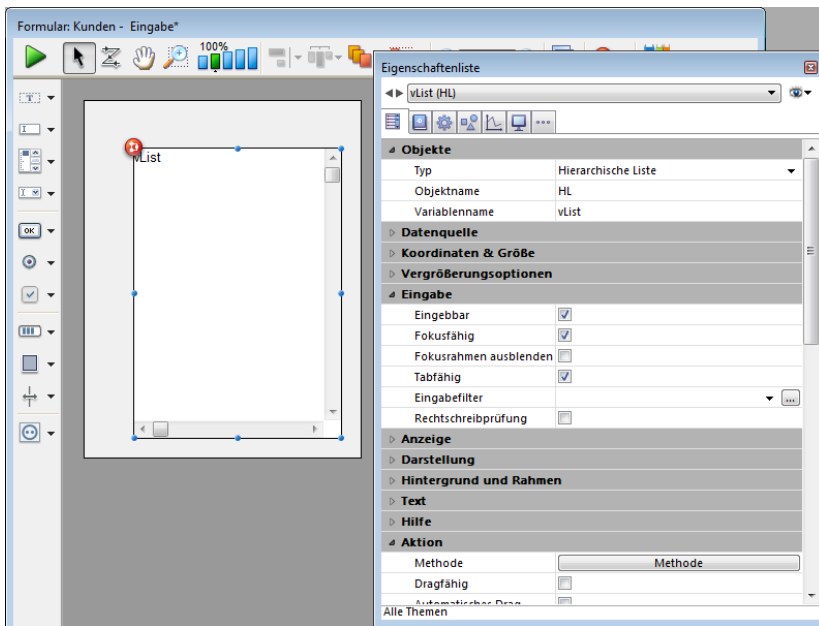


Weitere Informationen dazu finden Sie im Abschnitt [Get list item font](#).

Hierarchische Liste



Fügt eine hierarchische Liste mit Objektmethode für eine einfache mehrstufige Hierarchie hinzu. Sie können die Eigenschaften über die Eigenschaftsliste ändern.



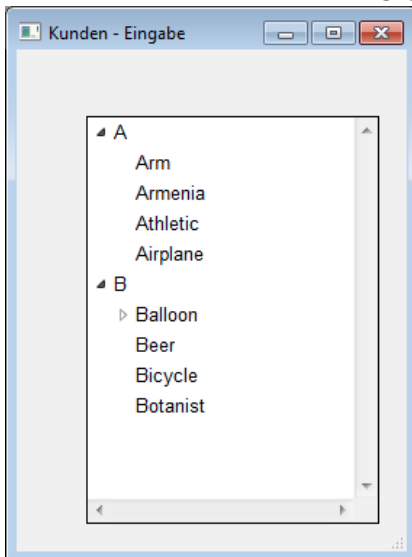
Sie können den Code der Objektmethode verändern, um die Hierarchie an Ihre eigenen Bedürfnisse anzupassen.

```

1 Case of
2   : (Form event=On_Load)
3
4   vList:=New list
5   vSublist1:=New list
6   vSublist2:=New list
7   vSubSublist2:=New list
8
9   APPEND TO LIST(vSublist1;"Arm";1001)
10  APPEND TO LIST(vSublist1;"Armenia";1002)
11  APPEND TO LIST(vSublist1;"Athletic";1003)
12  APPEND TO LIST(vSublist1;"Airplane";1004)
13
14  APPEND TO LIST(vSubSublist2;"Volley";20001)
15  APPEND TO LIST(vSubSublist2;"Soccer";20002)
16  APPEND TO LIST(vSubSublist2;"Rugby";20003)
17
18  APPEND TO LIST(vSublist2;"Balloon";2001;vSubSublist2:False)
19  APPEND TO LIST(vSublist2;"Beer";2002) //beer
20  APPEND TO LIST(vSublist2;"Bicycle";2003) //bicycle
21  APPEND TO LIST(vSublist2;"Botanist";2004) //botanist
22
23  APPEND TO LIST(vList;"A";1;vSublist1:True)
24  APPEND TO LIST(vList;"B";2;vSublist2:True)
25
26   : (Form event=On_Unload)
27
28   CLEAR LIST(vList;*)
29
30 End case

```

Die Liste im Formular mit dem vorgegebenen Code sieht so aus:

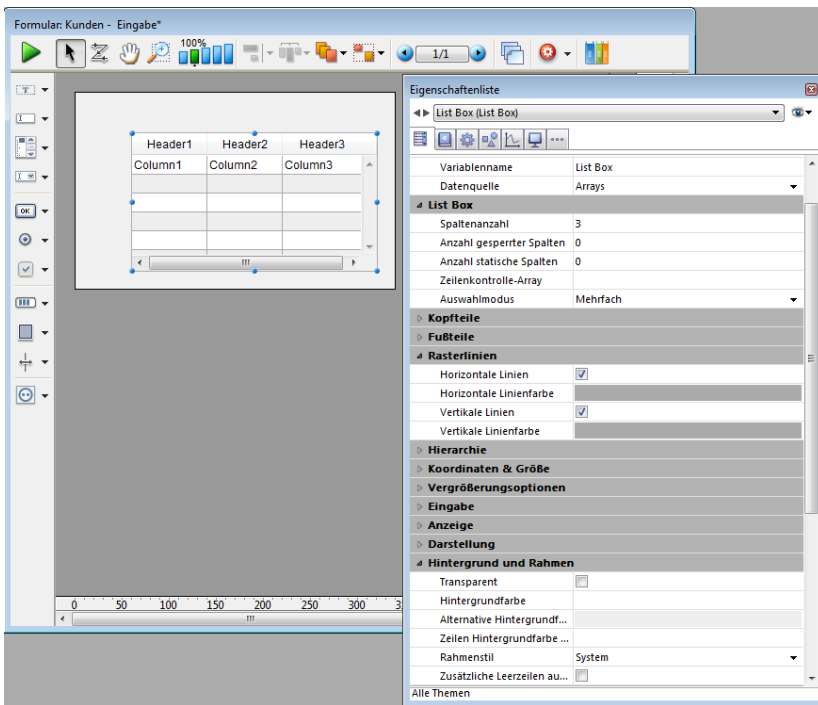


Weitere Informationen dazu finden Sie im Abschnitt **Hierarchische PopUp-Menüs und hierarchische Listen**.

Listbox



Fügt eine dreispaltige Listbox zum Anzeigen von Daten in Arrays hinzu. In der Eigenschaftenliste können Sie die Datenquelle und die gewünschten Optionen definieren.

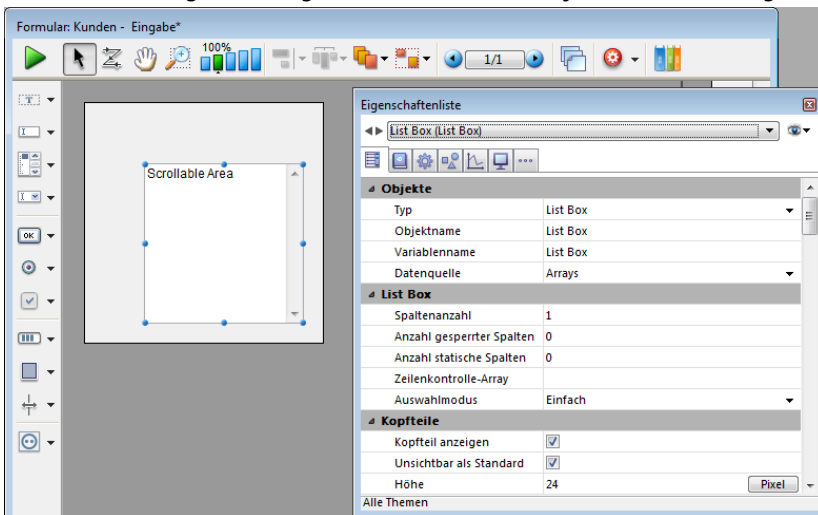


Weitere Informationen dazu finden Sie im Abschnitt **Überblick** für Listboxen.

Rollbarer Bereich



Fügt einen rollbaren Bereich hinzu, das ist eine Listbox mit einer einzelnen Spalte ohne Kopf- und Fußteile. Dieser Bereich ist standardmäßig nicht eingebbar. Sie können ihn jedoch über die Eigenschaftensliste konfigurieren.

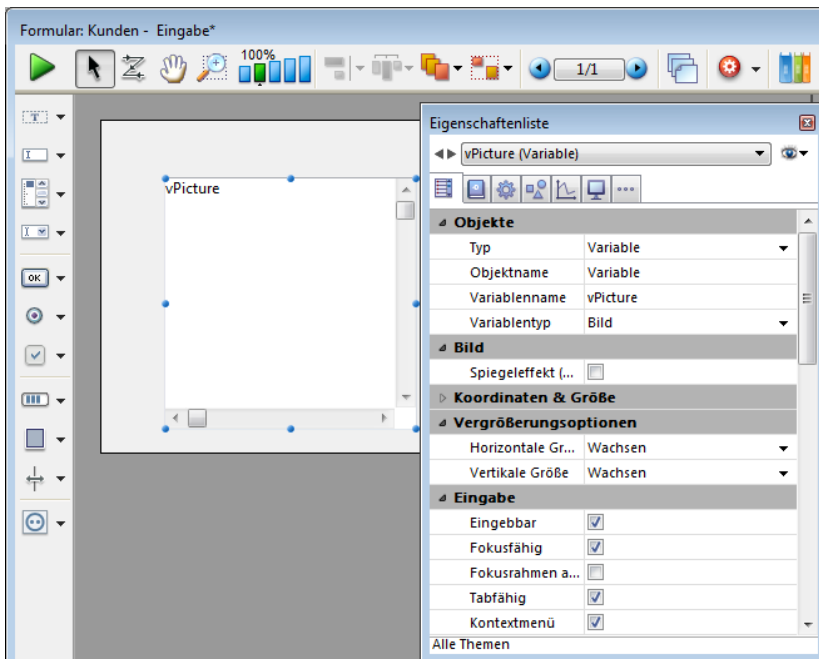


Weitere Informationen dazu finden Sie im Abschnitt **Überblick** für Listboxen.

Bild



Fügt einen Bildbereich hinzu, der sich über die Eigenschaftensliste konfigurieren lässt. Sie können z.B. ein Kontextmenü oder ein Anzeigeformat zuordnen.

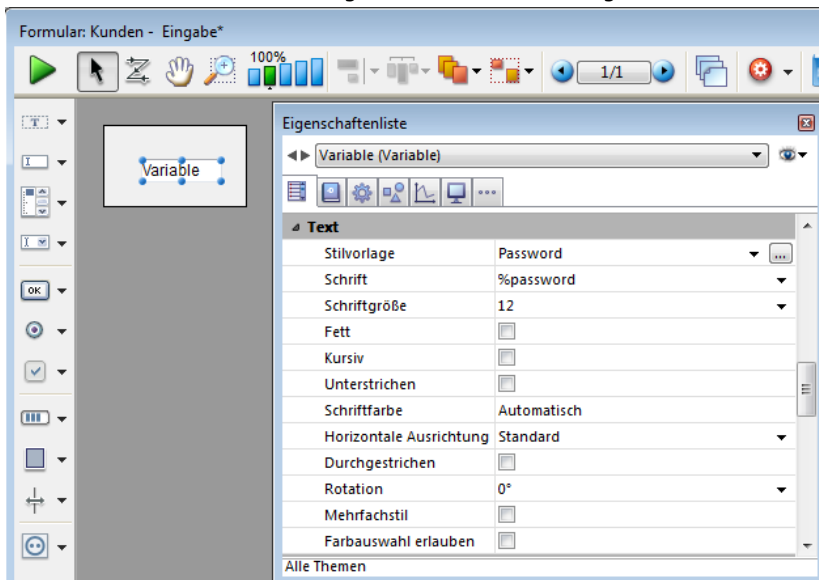


Weitere Informationen dazu finden Sie im Abschnitt .

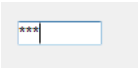
Kennwort



Fügt eine eingebbare String Variable mit der Stilvorlage "Password" hinzu, d.h. die eingegebenen Zeichen erscheinen als Sternchen. Sie können diese Eigenschaften über die Eigenschaftenliste ändern.



Zeichen, die Sie im Bereich Kennwort des Formulars eingeben, erscheinen als Sternchen:

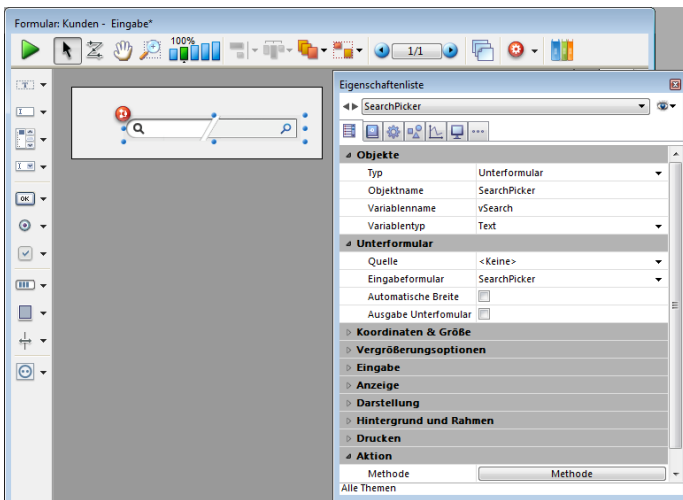


Weitere Informationen dazu finden Sie in den Abschnitten [Get list item font](#) und [Stilvorlagen](#).

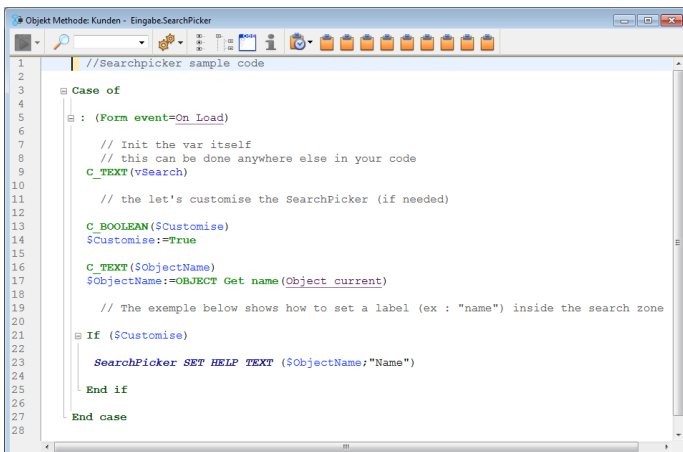
Suchbereich



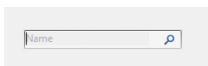
Fügt einen Bereich SearchPicker hinzu mit Objektmethode und anpassbarem Beispielcode. Sie können die Eigenschaften in der Eigenschaftenliste verändern.



Sie können den Beispielcode in der Objektmethode verändern und die Bezeichnung innerhalb der Suchzone über den Befehl **SearchPicker SET HELP TEXT** verändern.



Hier die Suchzone im Formular:

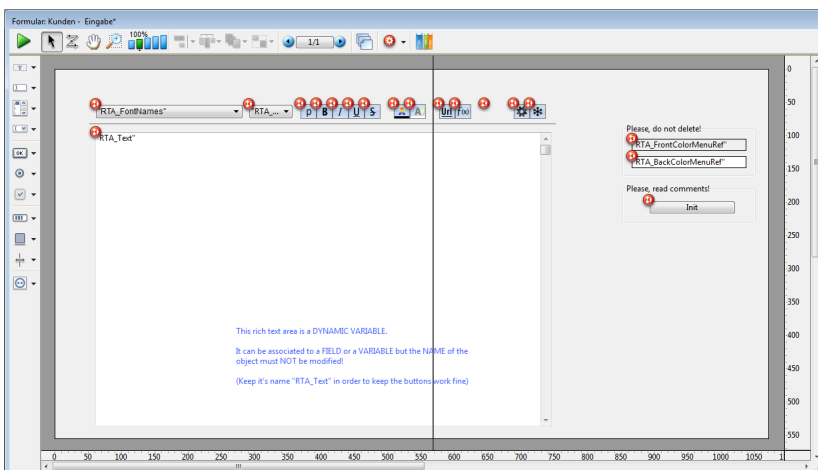


Weitere Informationen dazu finden Sie im Abschnitt **Überblick über SearchPicker** des SearchPicker Widget.

Rich Text Bereich



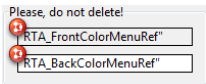
Fügt einen Rich Text Bereich mit einer Reihe Menüs und Schaltflächen zum Verwalten von Schriften und Referenzen hinzu.



Der Bereich besteht aus einer dynamischen Variable mit Namen "RTA_Text". Damit die Menüs und Schaltflächen korrekt funktionieren, dürfen Sie diesen Namen **nicht** verändern.

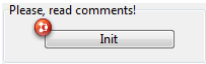
Hinweis: Damit die in Ihre Anwendung kopierten XLIFF Referenzen funktionieren, müssen Sie nach Einsetzen des Rich Text Bereichs im Formular einen Neustart machen.

Es gibt zwei Offscreen Variablen zum Sichern von Menüreferenzen für Schriftfarbe und Hintergrundfarbe. Damit werden die Menüs bei "On Load" generiert und bei "On Unload" aufgehoben.



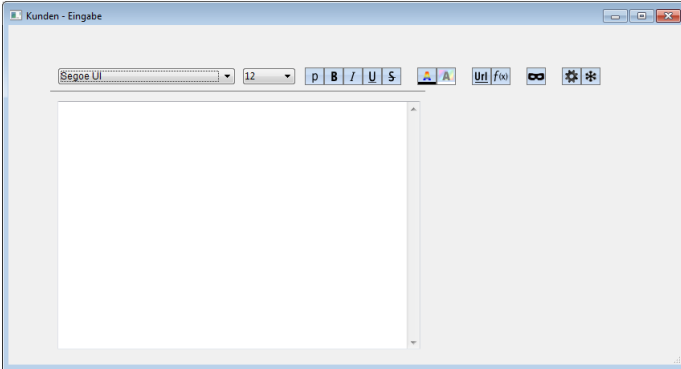
Hinweis: Werden diese Variablen gelöscht (nicht empfohlen), müssen diese Menüs jedes Mal, wenn ein Benutzer auf die Schaltflächen für Schrift- und Hintergrundfarbe klickt, neu erstellt und dann wieder gelöscht werden.

Über die offscreen Schaltfläche mit Namen "Init" werden bestimmte Ressourcen aus dem 4D Programm in Ihre Anwendung kopiert.


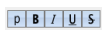






Diese Ressourcen werden innerhalb der Schaltfläche und im Farbmenü verwendet. Sie müssen nur einmal kopiert werden, diese Schaltfläche sollte nach Ausführen des Skripts gelöscht werden.

Im Formular im Benutzermodus sieht der Rich Text Bereich folgendermaßen aus:



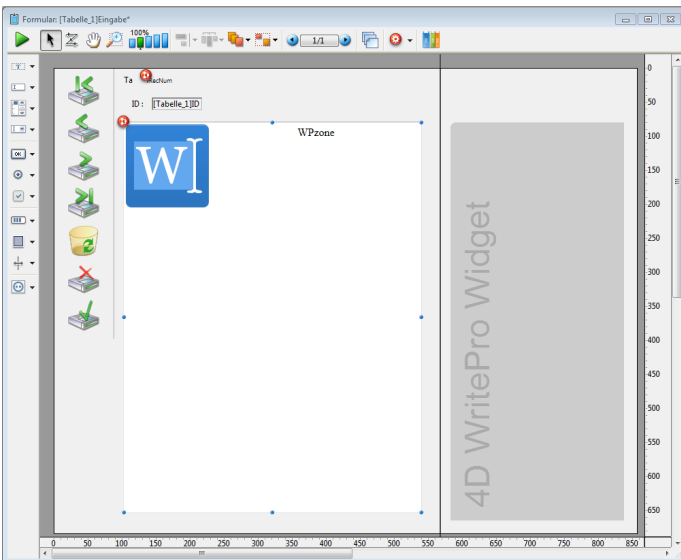
Im Rich Text Bereich gibt es standardmäßig folgende Menüs und Schaltflächen:

-  DropDown Menüs zum Auswählen von Schrifttyp und -größe
-  Schaltflächen für Schriftstile Fett, Kursiv, Unterstrichen und Durchgestrichen
-  Schaltflächen für Schriftfarbe und Hintergrundfarbe
-  Schaltfläche zum Einfügen einer URL bzw. eines Ausdrucks
-  Schaltfläche, um Ausdrücke als Referenzen (Strings) anstatt als Werte anzuzeigen.
-  Schaltflächen, um Ausdrücke im Text zu berechnen bzw. einzufrieren.

4D Write Pro Area




Fügt einen vorkonfigurierten 4D Write Pro Bereich mit 4D Write Pro Widget hinzu, das mehrere Steuer-Panels zum Verwalten des Inhalts enthält:



Hinweis: Weitere Informationen zu 4D Write Pro finden Sie im **4D Write Pro Handbuch**.

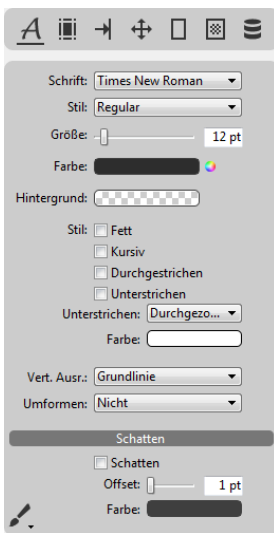
Im Benutzermodus zeigt das Widget standardmäßig eine Toolbar mit 8 Icons. Jeder Icon öffnet ein Steuer-Panell mit spezifischen Attributen. Sie sind folgendermaßen gegliedert:

-  **Schriftstil**
-  **Ränder & Ausrichtung**
-  **Tabulator**
-  **Einheit & Größe**
-  **Rahmen**
-  **Hintergrund**
-  **Information**
-  **Lesezeichen**

Es gibt auch eine Schaltfläche **Copy/Paste**  (in der linken unteren Ecke der meisten Panels) mit einem dynamischen Kontextmenü, das sich jeweils nach dem Inhalt des Panels richtet. So erscheint beispielsweise im Panel **Schriftstil** nach Kopieren einer formatierten Textauswahl im Kontextmenü der Eintrag "Stil kopieren":

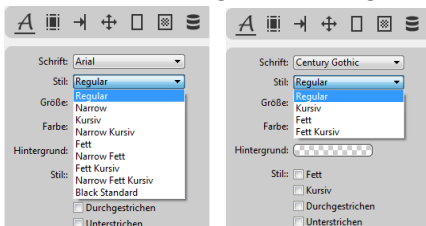


Schriftstil



Dieses Panel verwaltet die standardmäßigen Schriftstile und Eigenschaften für Text im 4D Write Pro Bereich.

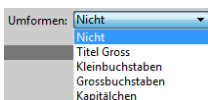
Die unter **Stil** verfügbaren Einträge können je nach gewählter Schrift variieren:




Hinweis: Die Schriftgröße ist immer in Punkt, unabhängig welche Einheit für das Dokument gesetzt wurde.

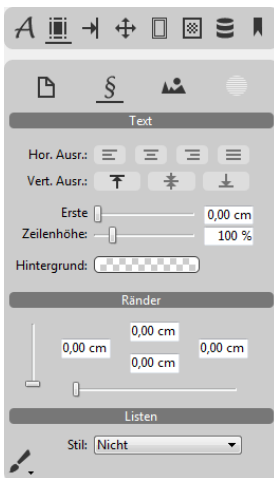
Das DropDown-Menü **Vert. Ausr.** bietet die Optionen Hochgestellt und Tiefgestellt für Text, das DropDown-Menü

Umformen bietet verschiedene Optionen für Groß- und Kleinschreibung:




Die Schaltfläche **Copy/Paste**  kopiert den auf den Text angewandten Stil. Haben Sie ausgewählten formatierten Text kopiert, wird im DropDown-Menü automatisch der Eintrag "Stil einsetzen" hinzugefügt, so dass Sie die Stilelemente an anderer Stelle wieder anwenden können. Das funktioniert jedoch nur, wenn die Formatierung auf die ganze Textauswahl angewendet wird.


Ränder und Ausrichtung




Dieses Panel bietet Optionen zur Ausrichtung von Text und setzt die Ränder für den 4D Write Pro Bereich.

Diese Einstellungen lassen sich für das ganze Dokument setzen, aber auch unabhängig für einen Absatz oder ein Bild im Text. Dafür dienen die Icons im oberen Bereich:

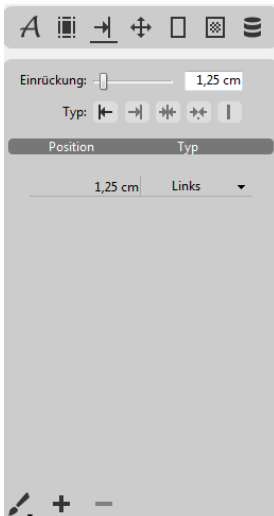
 für das Dokument als Ganzes

 für einen einzelnen Abschnitt

 für ein Bild innerhalb des Textes oder  für ein ausgewähltes verankertes Bild

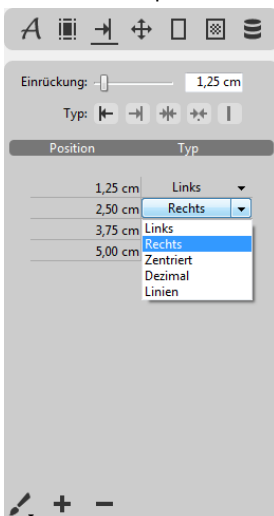
Über die Schaltfläche **Copy/Paste**  am Ende lassen sich die Einstellungen oder Ränder des ausgewählten Textes kopieren und einsetzen.

Tabulatoren






Dieses Panel steuert die Tabulatoren für Absätze im 4D Write Pro Bereich. Sie können den Wert für **Einrückung** entweder über den Regler setzen oder direkt einen Wert in den Eingabebereich eintragen. Er dient als Standardversatz zwischen allen danach hinzugefügten Tabulatoren. Wählen Sie über die Schaltflächen einen bestimmten **Typ**, wird dieser auf alle vorhandenen Tabs im Absatz angewandt.

Sie können einzelne Tab Stopps manuell ändern. Geben Sie dazu einen neuen Wert unter **Position** ein und wählen bei Bedarf im DropDown-Menü einen bestimmten **Typ**:




Hinweis: Änderungen unter **Einrückung** oder **Typ** im oberen Teil überschreibt alle Einstellungen, die im darunterliegenden Teil für einzelne Tabulatoren definiert wurden.

Über die Schaltfläche  am unteren Rand wird im Absatz ein neuer Standard Tabulator hinzugefügt. Um einen angelegten Tabulator zu löschen, wählen Sie ihn in der Liste aus und klicken auf die Schaltfläche .

Gesetzte Tabulatoren gelten für den aktuellen Absatz oder eine Auswahl an Absätzen. Über die Schaltfläche **Copy/Paste**  am Ende lassen sich die Einstellungen für Tabulatoren kopieren und einsetzen.

Einheit und Größe



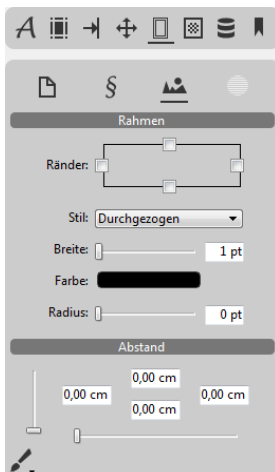
Dieses Panel definiert die standardmäßige Einheit für das 4D Write Pro Dokument und gibt die Ausmaße der Absätze und darin enthaltenen Bilder an. Die gesetzte Einheit gilt für das gesamte Dokument.

Hinweis: Unabhängig von der definierten Einheit für das Dokument wird für Schriftgröße (siehe Tafel), sowie Liniestärke und Radius für Rahmenecken (siehe Tafel) immer die Einheit Punkt verwendet.

Absätze können eine feste oder variable Breite haben, Bilder eine feste Größe oder eine Mindestbreite bzw. Mindesthöhe. Mit der Einstellung "auto" richtet sich die Größe nach dem Inhalt des Elements.


Im Bereich **Ersatztext** können Sie eine Meldung angeben, die erscheint, wenn sich das Bild nicht anzeigen lässt.


Rahmen



Dieses Panel verwaltet Rahmen und Füllen im 4D Write Pro Bereich.

Diese Einstellungen lassen sich für das ganze Dokument setzen, aber auch unabhängig für einen Absatz oder ein Bild im Text. Dafür dienen die Icons im oberen Bereich:


 für das Dokument als Ganzes

 für einen einzelnen Absatz

 für ein Bild innerhalb des Textes oder  für ein ausgewähltes verankertes Bild.


Hinweis: Die Rahmentypen Doppelt, Groove - Geritzt, Ridge - Aufliegend, Inset - Vertieft sind bei der Standardbreite 1 pt nicht sichtbar. Dazu muss der Rahmen breiter sein.

Radius definiert die Rundung der Rahmenecken. Diese Einstellung ist für die Rahmentypen - Geritzt, Ridge - Aufliegend oder Inset - Vertieft nicht möglich.


Die Schaltfläche **Copy/Paste**  kopiert die Einstellungen für einen Absatz oder Bild und setzt sie in einen anderen.

Hintergrund

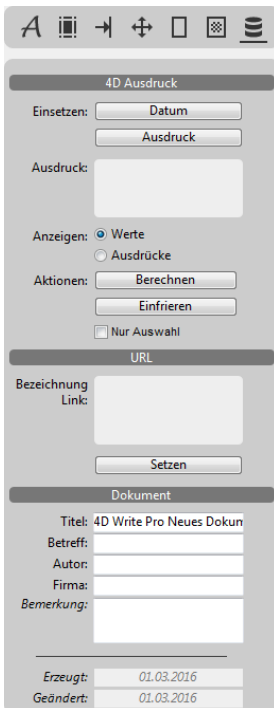


Dieses Panel steuert Hintergrundbilder für den 4D Write Pro Bereich. Sie können Bilder per Drag-and-Drop direkt in den Bereich **Bild** setzen oder eine URL eintragen. Mit der rechten Maustaste erscheint ein DropDown-Menü im Bereich **Bild**, in dem Sie verschiedene Muster für den Hintergrund wählen können. Sie können die Position und Größe des Bildes oder Muster setzen und auch eigene Einstellungen definieren. Mit dem Icon  können Sie ein gesetztes Bild oder Muster wieder entfernen.

Ein Hintergrundbild lässt sich für das gesamte Dokument () setzen bzw. für einzelne Absätze (). Das ist auch für ein Bild innerhalb des Textes möglich ().

Die Schaltfläche **Copy/Paste**  kopiert die Einstellungen für das Hintergrundbild von einem Absatz oder Bild und setzt sie in einen anderen.

Datenbank und Information



Dieses Panel steuert und fügt 4D Ausdrücke und URLs in den 4D Write Pro Bereich ein und bietet hilfreiche Angaben zum Dokument.

Es gibt folgende Möglichkeiten:

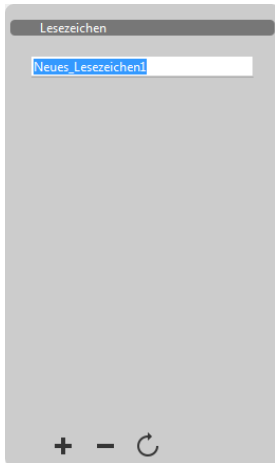
- **Datum:** Fügt das aktuelle Datum an der Cursorposition ein.
- **Ausdruck** (Schaltfläche): Öffnet den Formel-Editor, so dass Sie einen 4D Ausdruck erstellen oder einen vorhandenen laden können. Er wird an der aktuellen Cursorposition eingefügt.
- **Ausdruck** (Bereich): Zeigt die Referenz auf den gewählten 4D Ausdruck. Sie können spezifische Ausdrücke zu Dokument- oder Seitenattributen einfügen. Weitere Informationen dazu finden Sie im Abschnitt **Ausdrücke für Dokument oder Seite einfügen**.
- **Anzeigen Werte/Ausdrücke:** Gibt an, ob der Ausdruck als Referenz oder mit seinen aktuellen Werten angezeigt wird.
- **Berechnen:** Aktualisiert die Werte von 4D Ausdrücken im Dokument.
- **Einfrieren:** Wandelt den aktuellen 4D Ausdruck in Volltext um (lässt sich nicht rückgängig machen).
- **Nur Auswahl:** Führt die Aktionen **Berechnen** und **Einfrieren** nur für den ausgewählten Text im 4D Write Pro Bereich aus. Ist diese Option nicht markiert, gelten diese Aktionen für das gesamte Dokument.
- **Bezeichnung/Link:** Zeigt die Link-Adresse und Bezeichnung der ausgewählten URL.
- **Setzen:** Öffnet ein Dialogfenster, um eine URL einzugeben oder zu ändern (Link Adresse und Bezeichnung).

Hinweis: Der 4D Write Pro Bereich muss den Fokus haben, damit die Bereiche **4D Ausdruck** und **URL** dieser Steuertafel aktiv sind.

Dokument

Die hier eingetragenen Informationen werden mit dem Dokument gespeichert, erscheinen aber nicht. Der Titel lautet standardmäßig "4D Write Pro Neues Dokument". Die Datumsangaben für "Erzeugt" und "Geändert" sind vorgegeben und lassen sich nicht verändern.

Lesezeichen



Dieses Panel verwaltet Lesezeichen für Dokumente im 4D Write Pro Bereich. Lesezeichen sind dynamische Referenzen zu spezifischen Ausschnitten in diesem Dokument; wird der zugeordnete Ausschnitt verschoben, erweitert und verkleinert, bezieht sich das Lesezeichen weiterhin auf denselben Ausschnitt im Dokument..

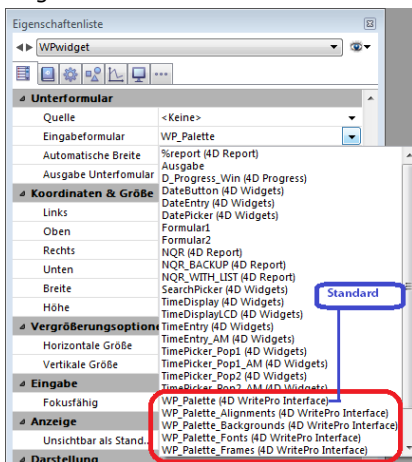
Um ein Lesezeichen zu erstellen, wählen Sie den entsprechenden Textausschnitt und klicken am unteren Rand auf die Schaltfläche **+**: In der Liste erscheint der Eintrag "Neues_Lesezeichen1". Sie können ein Lesezeichen jederzeit umbenennen; klicken Sie auf den Eintrag, können Sie den Namen ändern. Beachten Sie, dass nur alphanumerische Zeichen erlaubt sind.

Sie können Lesezeichen Ausschnitte auch ohne Umbenennen neu setzen: Wählen Sie einfach das gewünschte Lesezeichen in der Liste, dann markieren Sie den neuen Textausschnitt und klicken am unteren Rand auf die Schaltfläche **↻**. Der neue Ausschnitt ersetzt automatisch den bisher dem Lesezeichen zugeordneten Ausschnitt.

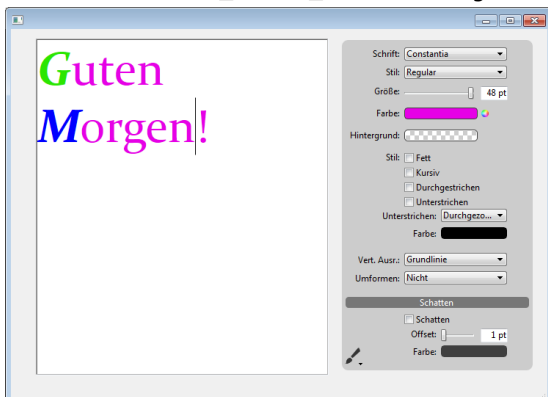
Die Liste zeigt Lesezeichen in derselben Reihenfolge wie sie im Dokument erscheinen. Wollen Sie ein Lesezeichen löschen, wählen Sie es in der Liste aus und klicken am unteren Rand auf die Schaltfläche **-**.

Steuer-Panels einzeln anzeigen

Das 4D Write Pro Formularobjekt enthält standardmäßig ein Unterformular 4D Write Pro Widget mit dem kompletten Satz Steuer-Panels. Standardmäßig ist "WP_Palette" als Eingabeformular vorgegeben. Sie können diese Panels aber auch einzeln verwenden. Wählen Sie dazu in der Eigenschaftensliste des Formulareditors den passenden Eintrag unter Eingabeformular:



Wählen Sie z.B. "WP_Palette_Fonts" als Eingabeformular, könnte Ihr Formular wie folgt aussehen:



Oberfläche der Steuer-Panels anpassen

Die Oberfläche der Steuer-Panels lässt sich verändern, um andere Varianten (skins) und Schriften zu verwenden. Dazu müssen Sie nur die Objektmethode der zugewiesenen 4D Write Pro Area verändern.

Stellen Sie sicher, dass die Variable des Objekts 4D Write Pro Area folgende zwingenden Attribute enthält (sie werden automatisch in die Standard Objektmethode eingefügt):

- *selection*: Wird von 4D Write Pro Befehlen wie **WP GET ATTRIBUTES** und **WP SET ATTRIBUTES** verwendet
- *areaName*: Wird von ST Befehlen verwendet.

Zusätzlich lässt sich ein optionales drittes Attribut (genannt "*skin*") hinzufügen, um die Darstellung des Panel individuell anzupassen. Der Wert dieses optionalen Attributs ist ein Objekt (im folgenden Beispiel \$WP_skin genannt) mit folgenden optionalen Attributen:

- *skinName*: Mögliche Werte "black", "dark", "grey", "light", "white" oder "night"
- Hintergrundfarbe: z.B. 0x00A0A0A0
- Separatorfarbe: z.B. 0x00D04060
- Schriftfarbe: z.B. 0x002080C0
- SeparatorSchriftfarbe: z.B. 0x00803000
- Schrift: Schriftfamilie, z.B. "Times"
- Schriftgröße: z.B. 12
- Scrollbar*: True oder False
 - * Die Scrollbar ist nur aktiviert und verwendbar, wenn das 4D Write Pro Formularobjekt vertikal anpassbar ist, d.h. in der Eigenschaftsliste unter der Gruppe Vergrößerungsoptionen die Option *Vertikale Größe* auf **Wachsen** gesetzt ist.

Weitere Informationen zu 4D Write Pro Attributen finden Sie im Abschnitt **4D Write Pro Attribute**.

Code-Beispiel zum Anpassen der Oberfläche für die Steuer-Panels des Assistenten:

```
C_OBJECT($WP_skin)
```

```
OB SET($WP_skin;"skinName";$skinName) // kann "black", "dark", "grey", "light", "white", "night" sein.
```

```
OB SET($WP_skin;"backgroundColor";0x00A0A0A0) //setzt Hintergrundfarbe für Panel
```

```
OB SET($WP_skin;"separatorColor";0x00D04060) //setzt Hintergrundfarbe für Trennbereiche
```

```
OB SET($WP_skin;"fontColor";0x002080C0) //setzt Schriftfarbe für Panel Text
```

```
OB SET($WP_skin;"separatorFontColor";0x00803000) //setzt Hintergrundfarbe für Trennbereiche Text
```

```
OB SET($WP_skin;"font";"Times") // setzt die verwendete Schrift
```

```
OB SET($WP_skin;"fontSize";13) // setzt die verwendete Schriftgröße
```

```
OB SET($WP_skin;"scrollbar";True)
```

```
// Jetzt müssen Sie diese individuelle Variante nur noch dem 4D Write Pro Objekt zuordnen.
```

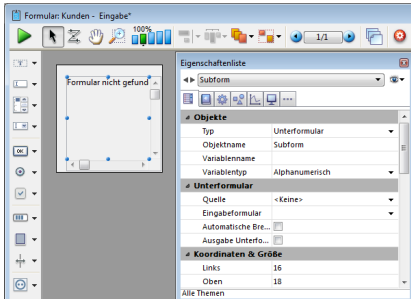
```
OB SET($WP_object;"skin";$WP_skin)
```

Plug-Ins und Unterformulare

Unterformular



Fügt einen Bereich Unterformular hinzu zur Anzeige eines Formulars aus einer anderen Anwendung oder eines Projekts, das gemeinsam mit einer Komponente benutzt wird.

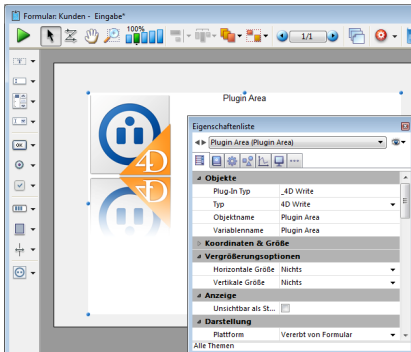


Weitere Informationen dazu finden Sie im Abschnitt **Überblick** unter Unterformulare und Widgets.

4D Write Bereich



Fügt den Plug-In Bereich 4D Write für Textverarbeitung hinzu.

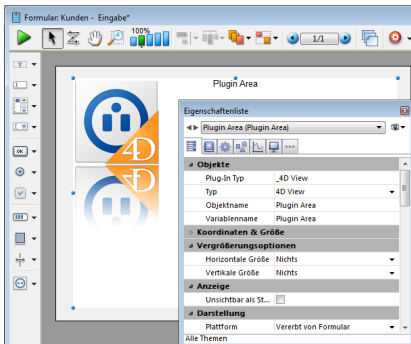


Weitere Informationen dazu finden Sie im Abschnitt **Plug-In Bereich**.

4D View Bereich



Fügt den Plug-in Bereich 4D View für Tabellenkalkulation hinzu.

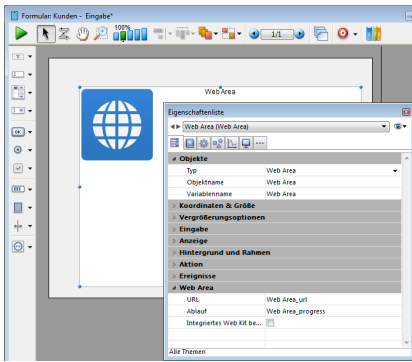


Weitere Informationen dazu finden Sie im Abschnitt **Plug-In Bereich**.

Web Area



Fügt einen Web Bereich zur Anzeige lokaler HTML Seiten oder Seiten aus dem Web hinzu. Der Inhalt lässt sich per Programmierung steuern.

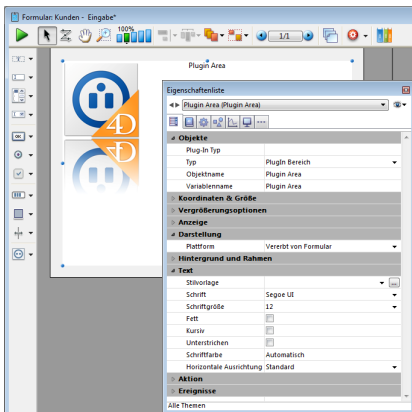


Weitere Informationen dazu finden Sie im Abschnitt **Web Areas**.

Plug-In



Fügt einen Plug-In Bereich zur Verwendung externer 4D Tools hinzu (z.B. 4D Write, 4D View, 4D Web Area), sowie Tools von Drittherstellern.



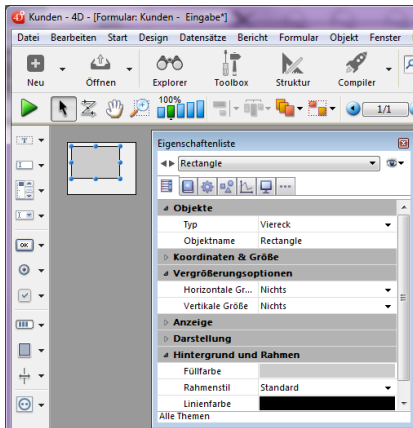
Weitere Informationen dazu finden Sie im Abschnitt **Plug-In Bereich**.

Statische Objekte

Viereck



Fügt ein Viereck hinzu. Sie können die Eigenschaften (Farbe, Linienstärke, Muster, etc.) über die Eigenschaftenliste oder per Programmierung festlegen.

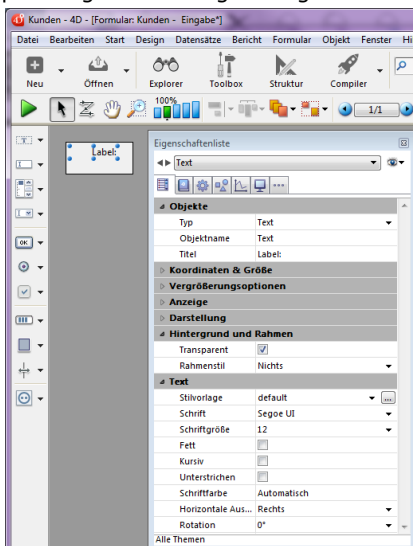


Weitere Informationen dazu finden Sie im Abschnitt [Objekteigenschaften setzen](#).

Etikett



Fügt ein Etikett hinzu. Sie können die Eigenschaften (Farbe, Linienstärke, Muster, etc.) über die Eigenschaftenliste oder per Programmierung festlegen.

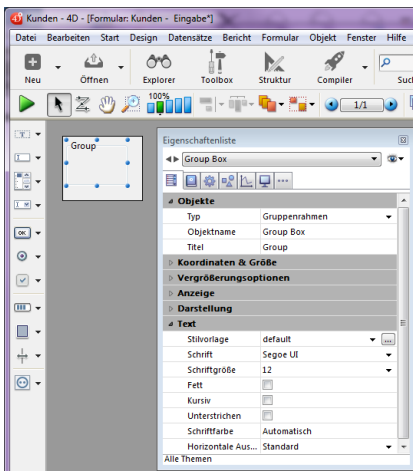


Weitere Informationen dazu finden Sie im Abschnitt [Objekteigenschaften setzen](#).

Gruppenrahmen



Fügt einen Gruppenrahmen hinzu. Sie können die Eigenschaften (Farbe, Linienstärke, Muster, etc.) über die Eigenschaftenliste oder per Programmierung festlegen.



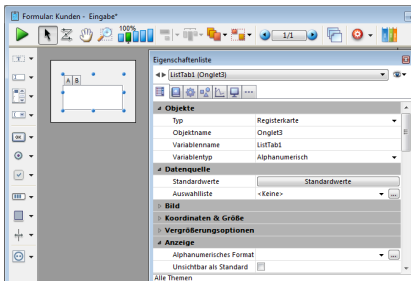
Weitere Informationen dazu finden Sie im Abschnitt **Objekteigenschaften setzen**.

Registerkarten und Splitter

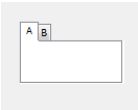
Registerkarte



Fügt eine Registerkarte hinzu mit Inhalt aus einem Array oder einer hierarchischen Liste. Sie können seine Eigenschaften über die Eigenschaftsliste verändern.



Im Formular erscheint die Registerkarte so:

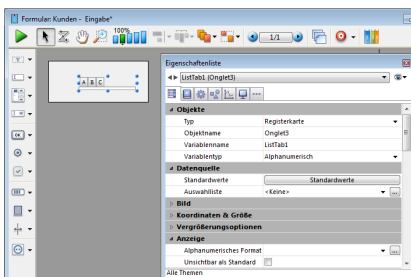


Weitere Informationen dazu finden Sie im Abschnitt [Registerkarten](#).

Registerkarte ohne Rahmen (nur Mac OS)



Fügt eine Registerkarte mit denselben Eigenschaften, jedoch ohne Rahmen hinzu, mit Inhalt aus einem Array oder einer hierarchischen Liste. Diese Variante ist nur auf Mac OS verfügbar.

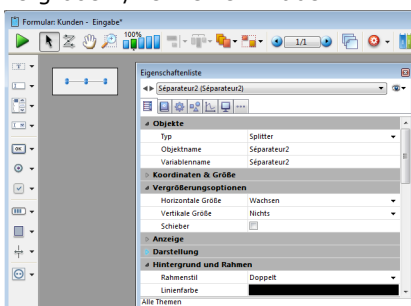


Weitere Informationen dazu finden Sie im Abschnitt [Registerkarten](#).

Horizontaler Splitter



Fügt einen horizontalen Splitter hinzu zum Unterteilen des ganzen Formulars oder Teilen davon in zwei Bereiche, die unabhängig vergrößerbar sind. Die vom Splitter abhängigen Objekte müssen passende Eigenschaften zum Vergrößern/Verkleinern haben.

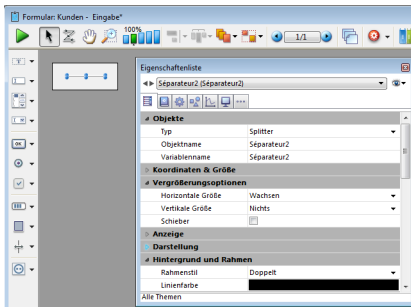


Weitere Informationen dazu finden Sie im Abschnitt [Splitter](#).

Vertikaler Splitter



Fügt einen horizontalen Splitter hinzu zum Unterteilen des ganzen Formulars oder Teilen davon in zwei Bereiche, die unabhängig vergrößerbar sind. Die vom Splitter abhängigen Objekte müssen passende Eigenschaften zum Vergrößern/Verkleinern haben.

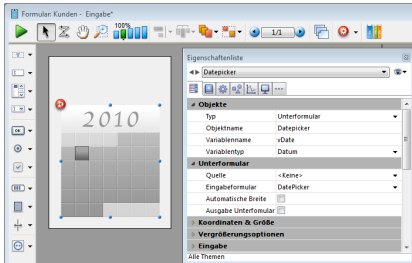


Weitere Informationen dazu finden Sie im Abschnitt [Splitter](#).

Date Picker



Fügt einen Kalender mit Datumsvariable und Objektmethode hinzu. Sie können ihn konfigurieren, um z.B. Feiertage oder Zeitbereiche anzuzeigen.



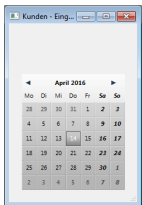
Sie können den Beispielcode in der Objektmethode über die Befehle für das Widget Date Picker verändern.

```

1 //Datepicker sample code
2
3 Case of
4
5   @ : (Form event=On_Load)
6
7     // Init the var itself
8     // this can be done anywhere else in your code
9     C_DATE(vDate)
10
11    // let's customise the datepicker if needed;
12    // all examples below are inactivated (if false)
13    C_BOOLEAN($Customise)
14    $Customise:=False
15
16    C_TEXT($FormName)
17    $FormName:=OBJECT Get name (Object current)
18
19    // the line below is useless until you use "DatePicker SET DEFAULT..." commands
20
21    // DatePicker RESET DEFAULT VALUES
22
23    // the example below shows how to set days off of a week
24    // just define a 7 rows boolean array and set days that are off to "true"
25    // Tip : use 40 constants for code lisibility
26
27    # If ($Customise)
28    ARRAY BOOLEAN($ WeekDaysOff;7) //lets say wednesday and sunday are days off
29    $ WeekDaysOff(Wednesday):=True
30    $ WeekDaysOff(Sunday):=True
31    DatePicker obj DATE OFF ($FormName;0->$ WeekDaysOff) //0 means weekly days off

```

Im Formular sieht der Kalender so aus:



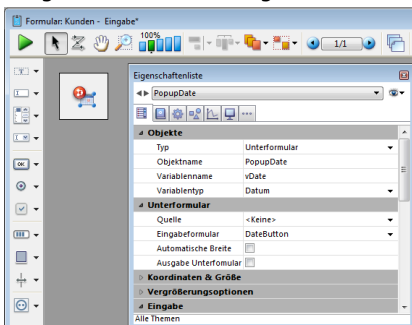
Weitere Informationen dazu finden Sie im Abschnitt **DatePicker**.

Hinweis: Date Picker, Pop-Up Datum und Datums-Eingabebereiche nutzen zu Beginn alle dieselbe Datumsvariable und zeigen bei Verwenden im gleichen Formular das jeweils zutreffende Datum.

Pop-Up Datum



Fügt einen Pop-Up Bereich mit Objektmethode hinzu, der einen Kalender zum Auswählen eines Datums anzeigt. Ausgewählte Datumsangaben werden direkt der verknüpften Datumsvariablen zu gewiesen.



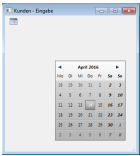
Sie können den Beispielcode in der Objektmethode mit den Befehlen für das Widget Date Picker verändern.

```

1 //Populate sample code
2
3
4
5
6 # (Form event=On_Load)
7 // Init the var itself
8 // this can be done anywhere else in your code
9
10 C_DATE:=Date
11
12 // let's customize the DatePicker if needed:
13 // All examples below are inactivated (if false)
14 C_BOOLEANS:=Customize
15 $Customize:=False
16
17 C_TEXT:=PrintName
18 #Function=MYOBJ Get_name(Object current)
19 // the line below is useless until you use "DatePicker SET DEFAULT." commands
20 // DatePicker RESET DEFAULT VALUES
21
22 // the example below shows how to set days off of a week
23 // just define a 7 row boolean array and set days that are off to "true"
24 // Tip: use 40 constants for code liability
25
26 # IF {Customize}
27 $DATE:=BOOLEANS(WeekDaysOff?) // lets say wednesday and saturday are days off
28 $WeekDaysOff(BOOLEANS)=False
29 $WeekDaysOff(Saturday)=True
30 $DatePicker SET DATE OFF (#Function()->$WeekDaysOff) // means weekly days off
31 End if
32
33 // the example below show how to define a min and a max enterable date
34
35

```

Im Formular sieht der Kalender so aus:



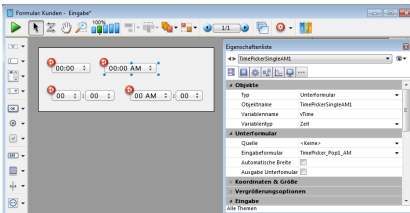
Weitere Informationen dazu finden Sie im Abschnitt **DatePicker**.

Hinweis: Date Picker, Pop-Up Datum und Datums-Eingabebereiche nutzen zu Beginn alle dieselbe Datumsvariable und zeigen bei Verwenden im gleichen Formular das jeweils zutreffende Datum.

Datumseingabebereich



Fügt einen Datumseingabebereich mit drei getrennten Feldern für Tag, Monat und Jahr und mit Objektmethode hinzu. Sie können auch definieren, dass nur ein bestimmter Datumsbereich eingebbar ist.



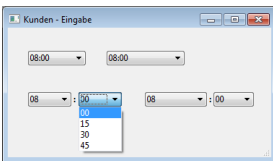
Sie können den Beispielcode in der Objektmethode über die Befehle für das Widget Date Picker verändern.

```

1 //TimePickerSample AM
2
3
4
5
6 # (Form event=On_Load)
7 // declare the variable here or anywhere else in your own code
8
9 C_TIME:=vTime
10
11 // then if needed, customize the time picker
12 // default values are Min = 8:00 Max = 20:00 Step = 0:15
13 // All examples below are inactivated (if false)
14 C_BOOLEANS:=Customize
15 $Customize:=False
16
17 # IF {Customize}
18 //use current object (if you don't the methods will change default values of time)
19 $ObjectName=COBJEKT Get_name(Object current)
20
21 TimePicker SET MIN TIME (#ObjectName;10:00:00)
22 TimePicker SET MAX TIME (#ObjectName;18:00:00)
23 TimePicker SET STEP (#ObjectName;700:20:00)
24
25 End if

```

Im Formular erscheint der Datums-Eingabebereich so:



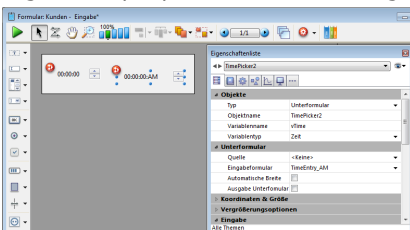
Weitere Informationen dazu finden Sie im Abschnitt **DatePicker**.

Hinweis: Date Picker, Pop-Up Datum und Datums-Eingabebereiche nutzen zu Beginn alle dieselbe Datumsvariable und zeigen bei Verwenden im gleichen Formular das jeweils zutreffende Datum.

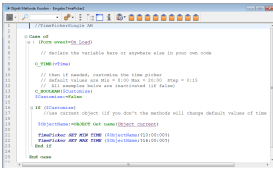
Time Picker



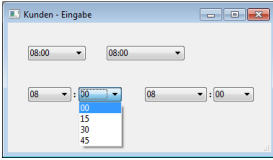
Fügt ein Pop-Up Bereich für Zeiteingabe hinzu. Sie können Mindest- und Maximumwerte sowie Zeitintervalle definieren.



Die Objektmethode enthält Beispielcode, den Sie über die Befehle für das Widget Time Picker verändern können.



Im Formular erscheinen die verschiedenen Zeitbereiche so:

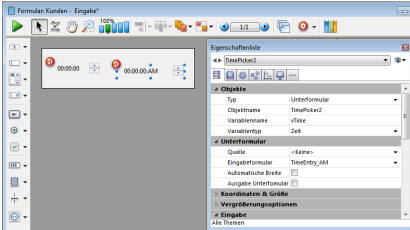


Weitere Informationen dazu finden Sie im Abschnitt **TimePicker**.

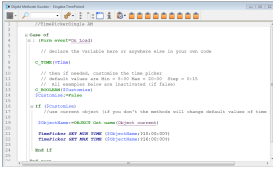
Zeiteingabebereich



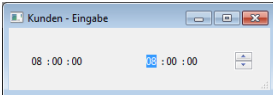
Fügt einen Zeiteingabebereich mit drei getrennten Feldern für Stunde, Minute und Sekunde ein. Sie können auch definieren, dass nur eine bestimmte Zeitspanne eingetragbar ist.



Die Objektmethode enthält Beispielcode, den Sie über die Befehle für das Widget Time Picker verändern können.



Im Formular erscheinen die Zeiteingabebereiche so:

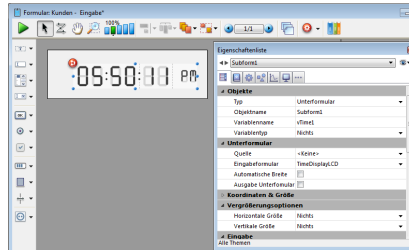
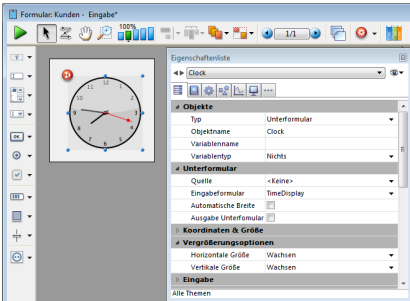


Weitere Informationen dazu finden Sie im Abschnitt **TimePicker**.

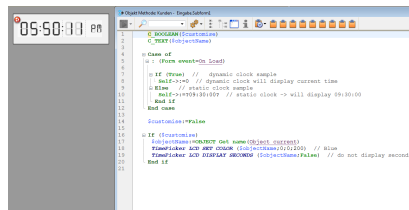
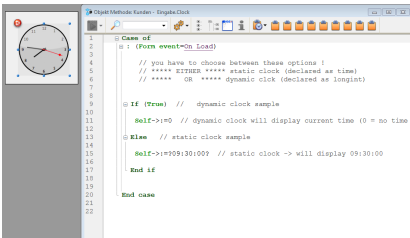
Zeitanzeigebereich



Fügt einen Zeitanzeigebereich (Uhr oder LCD) als statische Zeit oder laufende Uhr hinzu. Das richtet sich nach dem Typ der zugewiesenen Variable.

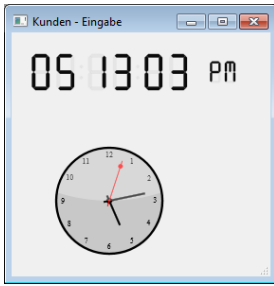


Die jeweilige Objektmethode enthält Beispielcode, den Sie über die Befehle des Widget Time Picker verändern können.



Deklarieren Sie die Variable in der Objektmethode als feste Zeit, erscheint sie im Anzeigebereich als statisches Objekt.




Für eine dynamische Zeit müssen Sie die Variable als Lange Ganzzahl deklarieren und den Wert auf "0" setzen. Dann zeigt die Uhr oder LCD die aktuelle Zeit.



Sie können auch andere Werte (positiv oder negativ) setzen, um einen Zeitversatz darzustellen. Die Berechnung erfolgt in Sekunden. Geben Sie beispielsweise "3600" an, wird die Anzeige eine Stunde vorgestellt, bei "-1800" eine halbe Stunde zurück.

Weitere Informationen dazu finden Sie im Abschnitt [TimePicker](#).

Unterformulare und Widgets

-  Überblick
-  Unterformulare als Liste
-  Unterformulare als Seite

Ein Unterformular ist ein Formular, das in ein anderes Formular eingebunden ist.

Es gibt zwei Haupttypen von Unterformularen:

- **Unterformulare als Liste**, hier geben Sie in der Regel Daten aus anderen Tabellen ein, zeigen sie an oder verändern sie,
- **Unterformulare als Seite**, hier können Sie spezifische vordefinierte Oberflächenobjekte verwenden, wie z.B. 4D Widgets.

Jede Art von Unterformular hat eigene Funktionsweisen und Eigenschaften.


Terminologie

Um die bei Unterformularen verwendeten Begriffe klar zu definieren, hier die wichtigsten Definitionen:

- **Unterformular**: Formular, das in ein anderes Formular, genannt Elternformular, eingebunden wird.
- **Elternformular**: Formular mit einem oder mehreren Unterformularen.
- **Unterformular Container**: Ein im Elternformular eingebundenes Objekt, das eine Instanz des Unterformulars anzeigt.
- **Unterformular Instanz**: Die Darstellung eines Unterformulars in einem Elternformular. Dieser Begriff ist wichtig, da sich in einem Elternformular verschiedene Instanzen des gleichen Unterformulars anzeigen lassen.
- **Listenformular**: Instanz des Unterformulars, angezeigt als Liste.
- **Detailformular**: Eingabeformular vom Typ Seite, das einem Unterformular vom Typ Liste zugeordnet ist und sich durch Doppelklick in die Liste öffnen lässt.

Unterformular erstellen

Sie können ein Unterformular auf drei Arten einem Formular hinzufügen:

- Im Formularassistenten mit der Seite Unterformular in den erweiterten Optionen (siehe Abschnitt **Formular über Formularassistent erstellen**)
- Im Formulareditor über das Symbol für Unterformular aus der seitlichen Objektleiste 
- Im Formulareditor per Drag-and-Drop über die **Seite Formulare** des Explorers

Hinweis: Ein Objekt vom Typ Unterformular (im Eingabeformular) wird auch erzeugt, wenn Sie aus der vorkonfigurierten Objektbibliothek ein Objekt vom Typ Widget hinzufügen (siehe Abschnitt **Vorkonfigurierte Objektbibliothek verwenden**).

Sie können die drei Arten bei Bedarf auch miteinander kombinieren.

Sie bestimmen den Formulartyp in der Eigenschaftsliste in der Gruppe "Unterformular" durch Markieren oder Nicht-Markieren der Option **Ausgabeformular**. Diese Operation zeigt jeweils weitere Optionen in der Eigenschaftsliste an, so dass sie zum gewählten Typ des Unterformulars passen.

Widgets

4D Widgets sind vordefinierte Objekte. Diese sog. Widgets sind mit oder ohne Programmierung verwendbar. Sie bieten Zugriff auf Standardfunktionalitäten, die sich leicht integrieren lassen. Es gibt folgende Widgets:

- **SearchPicker**: Suchbereich mit Standarddarstellung
- **DatePicker**: dynamischer Bereich für Datum
- **TimePicker**: dynamischer Bereich für Zeit

4D Widgets sind Unterformulare vom Typ Seite mit spezifischen Funktionen. Es gibt zwei Möglichkeiten, dies in Ihre Formulare einzufügen: Über das Werkzeug zum Erstellen von Unterformularen (siehe voriger Abschnitt) oder über die vorkonfigurierte Objektbibliothek von 4D (siehe **Vorkonfigurierte Objektbibliothek verwenden**).

Die Einsatzmöglichkeiten für Widgets werden ausführlich im Handbuch **4D Widgets** beschrieben.

Unterformulare als Liste

Ein Unterformular ist ein Listenformular aus einer anderen Tabelle, das in einem Detailformular angezeigt wird. Mit einem Unterformular als Liste können Sie Daten in anderen Tabellen eingeben, anzeigen und ändern. Sie verwenden normalerweise Unterformulare in Anwendungen, in der Sie Eine-zu-Viele-Verknüpfungen aufgebaut haben. Über ein Unterformular als Liste in einem Formular zu einer verknüpften Eine-Tabelle können Sie Daten aus einer verknüpften Viele-Tabelle anzeigen, eingeben und ändern. Sie können mehrere Unterformulare aus verschiedenen Tabellen im selben Formular haben. Sie können jedoch nicht zwei Unterformulare, die zur gleichen Tabelle gehören, auf dieselbe Seite des Formulars setzen.

Eine Anwendung zum Verwalten von Kontakten kann zum Beispiel ein Unterformular als Liste enthalten, das alle Telefonnummern für einen bestimmten Kontakt anzeigt. Obwohl die Telefonnummern in der Anzeige der Kontakte erscheinen, wird diese Information in einer verknüpften Tabelle gespeichert. Über eine Eine-zu-Viele-Verknüpfung in der Struktur lassen sich beliebig viele Telefonnummern pro Kontakt speichern. Mit automatischen Verknüpfungen können Sie die Dateneingabe direkt in die verknüpfte Viele-Tabelle unterstützen, ohne programmieren zu müssen.

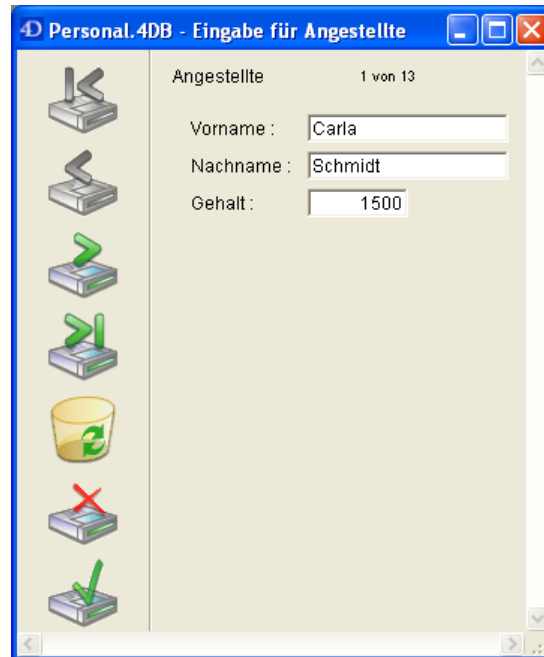
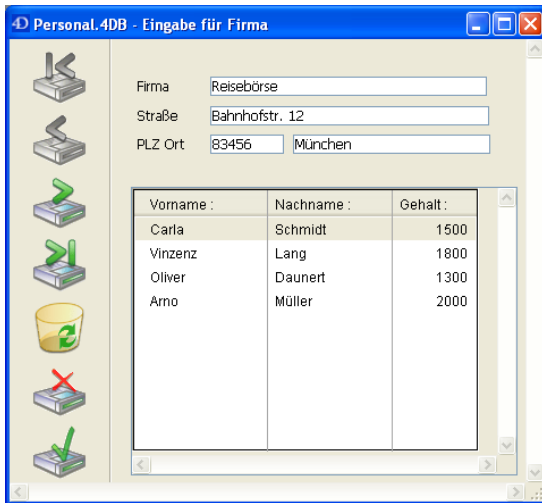
Obwohl Unterformulare als Listen im allgemeinen Viele-Tabellen zugeordnet sind, kann ein Unterformular Instanz die Datensätze aus jeder anderen Tabelle der Anwendung anzeigen.

Sie können ein Unterformular als Liste mit dem Formularassistenten erzeugen.

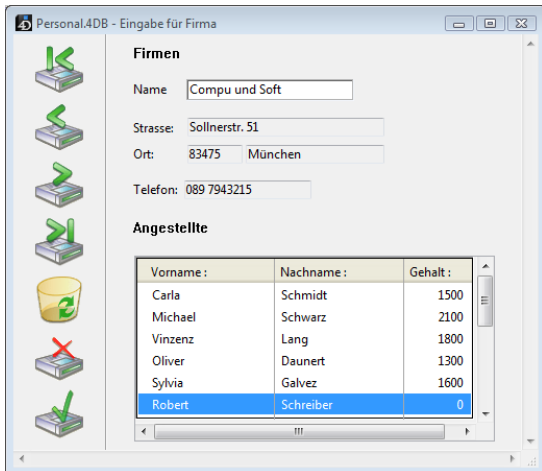
Sie können ein Unterformular mit dem Formularassistenten erzeugen oder über den Formulareditor ein bestehendes Formular hinzufügen. Sie müssen das Listenformular, das sie als Unterformular verwenden wollen, zuerst erstellen.

Vorname :	Nachname :	Position :

In Unterformulare als Liste lassen sich Daten auf zwei Arten eingeben: Der Benutzer kann Daten direkt im Unterformular eingeben oder sie in einem zugeordneten Eingabeformular eingeben. Hierbei wird das Formular, das als Unterformular verwendet wird, als *Listenformular* bezeichnet; das Eingabeformular als *Detailformular*.




Sie können dem Benutzer ermöglichen, Daten über das Listenformular einzugeben oder auf eine Zeile im Listenformular zu klicken, um das Detailformular zu öffnen.



Unterformular als Liste erstellen

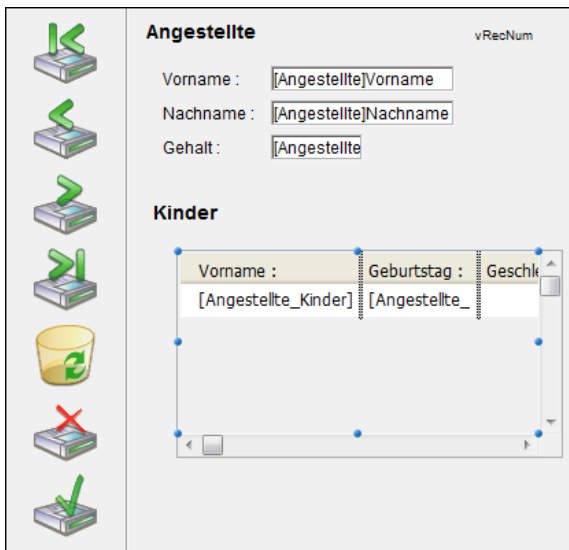
Um ein Listenformular zu definieren, müssen Sie das Unterformular, welches Sie im Elternformular verwenden wollen, zuerst erstellen und einrichten. Weitere Informationen zu Listenformularen finden Sie im Kapitel **Ausgabeformulare und Berichte**.

Im Elternformular erstellen Sie ein Objekt Unterformular über das Werkzeug Unterformular  der seitlichen Objektleiste (Variation der letzten Schaltfläche).

Hinweis: Sie können das Listenformular auch per Drag-and-Drop Operationen aus der **Seite Formulare** des Explorers anlegen.

Stellen Sie sicher, dass in der Eigenschaftsliste unter der Gruppe „Unterformular“ die Option **Ausgabeformular** markiert ist, wählen Sie unter **Quelle** die Quelltable sowie unter **Ausgabeformular** das passende Formular.

Im Elternformular können Sie den Unterformular Container beliebig in der Größe und Position verändern.



Schaltflächen für Datensätze in Unterformularen hinzufügen

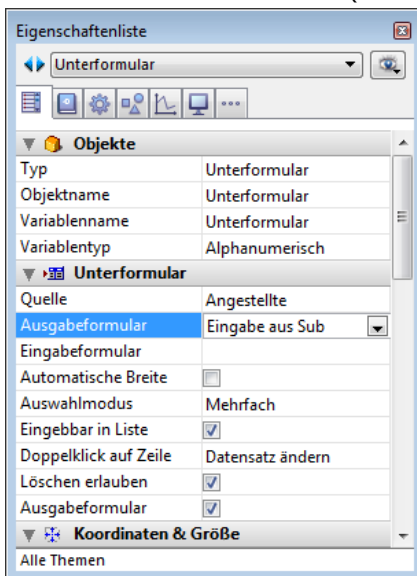
Sie können in einem Unterformular eigene Schaltflächen zum Steuern der Dateneingabe hinzufügen. Jeder Schaltflächentyp (invertierte, unsichtbare Schaltfläche, etc.) ist verwendbar. Sie können die Aktion dieser Schaltflächen über Programmierung (siehe Handbuch *4D Programmiersprache*, Kapitel **Eingabe**) oder über **Standardaktionen** definieren.

4D bietet drei Standardaktionen zum Verwalten der Grundfunktionen für Unterdatensätze: **Unterdatensatz öffnen**, **Unterdatensatz löschen** und **Unterdatensatz hinzufügen**. Enthält das Formular mehrere Unterformular Instanzen, gilt die Aktion für das Unterformular mit Fokus.

Weitere Informationen dazu finden Sie in den Abschnitten **Schaltflächen** und **Standardaktionen**.

Eigenschaften für Unterformulare als Liste

Sie können verschiedene Eigenschaften für Unterformulare als Liste einrichten und so automatische Funktionalitäten und für Benutzer erlaubte Aktionen (Auswahl und Dateneingabe) steuern.



Variablenname und Variablentyp

Diese Eigenschaften verwenden Sie in der Regel für Unterformulare als Seite. Weitere Informationen dazu finden Sie im Abschnitt **Unterformulare als Seite**.

Quelle

Damit können Sie die Art der Quelle für das Unterformular auswählen. Für Unterformulare als Liste wählen Sie die Tabelle, zu der das Unterformular gehört.

Eingabeformular

Damit können Sie dem Unterformular als Liste ein Eingabeformular zuordnen. Es dient zum Eingeben und Ansehen der Datensätze. Es enthält in der Regel mehr Informationen als im Unterformular als Liste angezeigt werden. Das Eingabeformular muss natürlich zur selben Tabelle wie das Unterformular gehören. Sie verwenden normalerweise ein Ausgabeformular als Listenformular und ein Eingabeformular als Detailformular. Definieren Sie kein Formular für den Listenmodus, verwendet 4D automatisch das standardmäßige Eingabeformular der Tabelle.

Je nach Konfiguration der Unterformulars kann der Benutzer das Detailformular durch Doppelklick auf einen Datensatz oder über die Befehle zum Hinzufügen und Bearbeiten der Datensätze im Unterformular anzeigen.

Hinweis: Sie können einem Unterformular als Liste ein Detailformular zuordnen, indem sie es bei gedrückter **Großschreibttaste** aus dem Explorer in den Unterformular Container ziehen.

Automatische Breite

Sie können einen Unterformularbereich in einem Formular wie jedes andere Objekt in der Größe ändern. 4D kann die Größe auch automatisch anpassen, so dass alle darin enthaltenen Datenfelder angezeigt werden. Dazu markieren Sie in der Eigenschaftensliste für das Unterformular die Option **Automatische Breite**.

Hinweis: Erstellen Sie einen Unterformularbereich direkt über das Symbol im Explorer, setzt 4D die Breite des Unterformulars automatisch.

Auswahlmodus

Für Unterformulare als Liste gibt es drei Auswahlmodi für den Benutzer: **Nichts**, **Einfach** und **Mehrfach**.

- **Nichts**

In diesem Modus lässt sich kein Datensatz auswählen. Ein Klick in die Liste hat keine Auswirkung, außer die Option **Eingebbar in Liste** ist markiert. Die Tasten zum Navigieren ermöglichen nur das Blättern in der Liste, das Formularereignis On Selection Change wird nicht erzeugt.

- **Einfach**

In diesem Modus lässt sich immer nur ein Datensatz auswählen. Klicken Sie auf einen Datensatz, wird er ausgewählt und zum aktuellen Datensatz. Klicken Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** auf einen Datensatz, wechseln Sie seinen Status (ausgewählt oder nicht ausgewählt).

Über die Pfeiltasten **Nach oben** und **Nach unten** wählen Sie den vorigen/nächsten Datensatz in der Liste. Mit den anderen Tasten navigieren Sie in der Liste. Das Formularereignis On Selection Change wird jedes Mal generiert, wenn der Datensatz geändert wird.

- **Mehrfach**

In diesem Modus lassen sich mehrere Datensätze gleichzeitig auswählen. Die ausgewählten Unterdatensätze werden mit dem Befehl **GET HIGHLIGHTED RECORDS** zurückgegeben. Klicken Sie auf einen Datensatz, wird er ausgewählt. Dies verändert jedoch nicht den aktuellen Datensatz. Klicken Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** auf einen Datensatz, wechseln Sie seinen Status (ausgewählt oder nicht ausgewählt). Mit der Kombination **Strg-Taste+Großschreibttaste+Klick** unter Windows, **Befehlstaste+Großschreibttaste+Klick** auf Mac OS können Sie mehrfache Auswahlen treffen.

Über die Pfeiltasten **Nach oben** und **Nach unten** wählen Sie den vorigen/nächsten Datensatz in der Liste. Mit den anderen Tasten navigieren Sie in der Liste. Das Formularereignis On Selection Change wird jedes Mal generiert, wenn der ausgewählte Datensatz geändert wird.

Eingebbar in Liste

Ist ein Unterformular als Liste im Modus **Eingebbar in Liste**, kann der Benutzer die Daten im Datensatz direkt in der Liste ändern, ohne das dazugehörige Eingabeformular zu verwenden. Er muss lediglich zweimal auf das Feld für die Bearbeitung klicken (Lassen Sie genügend Zeit zwischen den beiden Klicks, damit Sie keinen Doppelklick auslösen). Diese Option ist standardmäßig für alle Unterformulare als Liste aktiviert.

Vorname :	Nachname :	Gehalt :
Carla	Schmidt	1500
Michael	Schwarz	2100
Vinzenz	Lang	1800
Oliver	Daunert	1300
Sylvia	Galvez	1600
Robert	Schreiber	0

Ist diese Option nicht markiert, ist die Eingabe nur über das zugeordnete Eingabeformular möglich.

Aktionen bei Doppelklick

Sie können festlegen, wie sich ein Unterformular bei Doppelklick des Benutzers verhalten soll. In Anwendungen, die mit einer früheren Version von 4D erstellt wurden, können Sie auch eine Reaktion bei Doppelklick auf eine leere Zeile definieren (Option Kompatibilität).

- **Doppelklick auf Zeile:** Auszuführende Aktion bei Doppelklick auf einen Datensatz des Unterformulars. Es gibt folgende Optionen:
 - **Nichts:** Doppelklick ignorieren.
 - **Datensatz bearbeiten:** Ändert den Datensatz im Unterformular in den Bearbeitungsmodus. Die Änderungen werden direkt in der Liste ausgeführt, wenn die Option *Eingebbar in Liste* markiert ist. Sonst wird sie im Detailformular ausgeführt, das mit dem Unterformular verknüpft ist.
 - **Datensatz anzeigen:** Zeigt die Daten eines Datensatzes im Detailformular, das mit dem Unterformular verknüpft ist (nur Lesen).

- **Doppelklick auf Leerzeile:** Sie können definieren, welche Aktion bei Doppelklick auf eine Leerzeile im Unterformular ausgeführt wird. Es gibt folgende Optionen:
 - **Nichts:** Doppelklick ignorieren.
 - **Datensatz hinzufügen:** Erstellt einen neuen Datensatz im Unterformular und wechselt in den Bearbeitungsmodus. Er wird direkt in der Liste erstellt, wenn die Option *Eingebbar in Liste* markiert ist. Sonst wird er im Detailformular ausgeführt, das mit dem Unterformular verknüpft ist.

Erlaube Löschen

Der Benutzer kann standardmäßig Datensätze in einem Unterformular über die Tasten **Del** oder **Rückschritt** löschen. Da dies jedoch die Standardoperationen in der Oberfläche einiger Anwendungen beeinträchtigen kann – z.B. wenn sie auf Schaltflächen basieren – können Sie es über die Option **Erlaube Löschen** verhindern.

Ist diese Option inaktiv, kann der Benutzer über die Löschtasten keine Datensätze in Unterformularen löschen.

Fokusfähig

Auf Objekte im Unterformular lässt sich die Eigenschaft **Fokusfähig** (Gruppe Eingabe) anwenden. Ein Unterformular mit Fokus kann der Benutzer über die Navigationstasten steuern, bei einer mehrzeiligen Auswahl das Menü **Alle auswählen** verwenden, etc.

Erhält oder verliert ein Unterformular den Fokus, wird die Formularemethode des Hauptformulars über die Ereignisse On getting focus oder On losing focus aufgerufen. In diesem Fall gibt die 4D Funktion **OBJECT Get pointer** oder **Focus object** einen Zeiger auf die Tabelle des Unterformulars zurück.

Wie bei allen fokusfähigen Objekten lässt sich über die Eigenschaft **Fokusrahmen ausblenden** die grafische Anzeige des Fokus ausblenden.

Drucken

Da es mehr Datensätze geben kann, als der Unterformularbereich aufnehmen kann, bietet 4D unter der Gruppe „Drucken“ drei Optionen an, die das Drucken von Datensätzen in Unterformularen steuern:

- **Variabel** (Standardeinstellung)
Mit dieser Option erweitert oder verringert 4D den Unterformularbereich, um alle Datensätze zu drucken.
- **Fest (abgeschnitten)**
Mit dieser Option druckt 4D nur die Datensätze, die im Unterformularbereich erscheinen. Das Formular wird nur einmal gedruckt, nicht gedruckte Datensätze werden ignoriert.
- **Fest (mehrere Datensätze)**
Mit dieser Option wird die Ausgangsgröße des Unterformularbereichs beibehalten, 4D druckt das Formular jedoch mehrmals, um alle Datensätze auszudrucken.

Hinweise:

- Sie können keine Objekte neben den Rahmen des Objekts Unterformular Container legen, da jedes Objekt vor oder nach diesem Rahmen für jede Zeile des Unterformulars wiederholt wird.
- Sie können immer nur eine Unterformular Instanz aus derselben Tabelle auf derselben Formularseite drucken.
- Die Funktionen **Print object** und **Print form** sind nicht kompatibel mit den Druckoptionen für variable Größe.
- Mit dem Befehl **OBJECT SET PRINT VARIABLE FRAME** lässt sich die Option **Drucke variable Größe** per Programmierung setzen.
- Weitere Informationen zum Drucken von Objekten mit variabler Größe finden Sie im Abschnitt **WA OPEN BACK URL**.

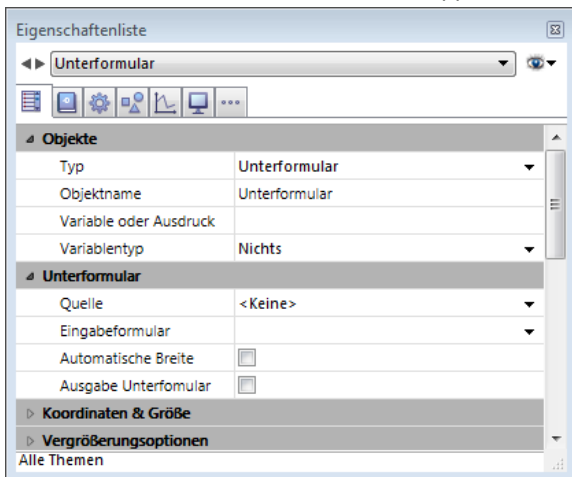
Unterformulare als Seite

Unterformulare als Seite können die Daten des aktuellen Unterdatensatzes oder, je nach Kontext (Variablen, Bilder, etc.), jede Art von dauerhaftem Wert anzeigen. Ein Hauptvorteil dieser Unterformulare ist, dass sie ergänzende Funktionalitäten haben und direkt mit dem Elternformular (Widgets) interagieren können. Unterformulare als Seite haben auch eigene Eigenschaften und Ereignisse; sie lassen sich komplett per Programmierung steuern.

Hinweis: Sie können Komponenten generieren, die über Unterformulare zusätzliche Funktionalitäten bieten. Weitere Informationen dazu finden Sie im Abschnitt **4D Komponenten entwickeln und installieren**.

Unterformular als Seite erstellen

Um ein Unterformular als Seite zu aktivieren, deaktivieren Sie einfach in der Eigenschaftensliste unter der Gruppe **Unterformular** die Option **Ausgabe Unterformular**. Dann erscheinen die Eigenschaften für Unterformulare als Liste nicht mehr, wie z.B. Auswahlmodus, Doppelklick auf Zeile, u.ä:



Das Unterformular als Seite verwendet das Eingabeformular, welches unter der Eigenschaft "Eingabeformular" angegeben ist. Es kann, anders als das Unterformular als Liste, von derselben Tabelle stammen wie das Elternformular. Sie können auch ein Projektformular verwenden. Bei der Ausführung hat das Unterformular als Seite die gleichen Anzeigemerkmale wie ein Eingabeformular. Die Funktionalitäten von Ausgabeformularen, hauptsächlich Markierer, sind nicht aktiv.

Eigenschaften von Unterformularen als Seite

Sie verwalten Unterformulare als Seite über spezifische Eigenschaften, welche die Einbindung und Interaktion mit dem Elternformular vereinfachen.

Variablenname

Sie können einem Objekt Unterformular eine Variable zuordnen. Sie hat standardmäßig keinen Namen, so kann Ihr Code über einen Zeiger darauf zugreifen. Weitere Informationen dazu finden Sie im Abschnitt **Dynamische Variablen** des Handbuchs *4D Programmiersprache*.

Sie hat einen Typ (siehe nächster Abschnitt) und lässt sich als Standardvariable im Elternformular darstellen. Das Ändern dieser Variablen löst Formularereignisse aus, so dass Sie die Werte in Elternformular und Unterformular aufeinander abstimmen können:

- Das Formularereignis [On Data Change](#) gibt im Unterformular Container an, dass der Wert der Variablen geändert wurde.
- Das Formularereignis [On bound variable change](#) gibt dem Unterformular über dessen Formularmethode an, dass die Variable im Elternformular geändert wurde.

Variablentyp

Damit können Sie den Typ der Variablen setzen, die dem Objekt Unterformular zugewiesen ist. Der Variablentyp bestimmt die Art der Werte, die über diese Variable zwischen Elternformular und Unterformular ausgetauscht werden. Standardmäßig ist als Typ **Nichts** vorgegeben. Behalten Sie diese Option bei, müssen Sie den Typ explizit über Code setzen, damit die Anwendung im kompilierten Modus arbeitet. Weitere Informationen dazu finden Sie im Abschnitt **Dynamische Variablen** des Handbuchs *4D Programmiersprache*.

Wählen Sie den Typ Objekt (mit Variablenname), können Sie über die Funktion **Form** die Eigenschaften dieses Objekts im Unterformular erhalten oder setzen (siehe unten **An Unterformular gebundenes Objekt verwenden**).

Quelle

Damit können Sie verschiedene Arten von Quellen auswählen:

- **<Keine>**: Wählen Sie diesen Typ, wenn Sie ein Projektformular oder eine Komponente als Unterformular verwenden wollen. Diese Unterformulare sind nur im Seitenmodus verwendbar: dazu muss in der Eigenschaftensliste

unter der Gruppe "Unterformular" die Option **Ausgabeformular** deaktiviert sein.

Damit ein Komponentenformular in der Eigenschaftenliste im DropDown-Menü "Eingabeformular" erscheint, muss es in der Komponente veröffentlicht sein. Weitere Informationen dazu finden Sie im Abschnitt **Unterformular veröffentlichen (Komponente)**.

- **Tabellenname**: Wählen Sie diesen Quelltyp, wenn Sie Tabellenformulare verwenden wollen.

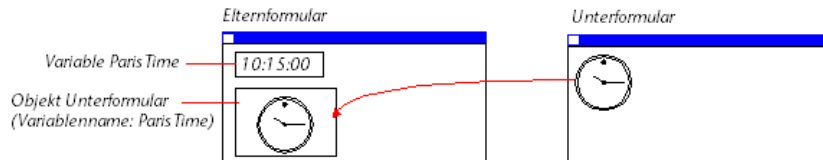
Objektmethode und Ereignisse

Ein Objekt Unterformular kann eine Objektmethode haben, um seine Funktionsweise zu steuern, insbesondere wenn Ereignisse ausgelöst werden. Weitere Informationen zu Ereignissen bei Unterformularen als Seite finden Sie unter der 4D Funktion **Form event**.

Beachten Sie, dass die Interaktion zwischen dem Inhalt des Objekts Unterformular und dem Elternformular über spezifische Mechanismen verwaltet werden muss (siehe **Formularübergreifende Programmierung**).

Zugewiesene Variable verwalten

Die dem Unterformular zugewiesene Variable ermöglicht, beide Kontexte (Formular und Unterformular) miteinander zu verbinden, um komplexen Oberflächen den Endschliff zu geben. Nehmen wir zum Beispiel ein Unterformular mit einer dynamischen Uhr, die in ein Elternformular eingefügt wird, das wiederum eine einstellbare Variable vom Typ Zeit enthält:



Beide Objekte (Zeitvariable und Unterformular Container) haben denselben Variablenamen. In diesem Fall synchronisiert 4D beide Werte automatisch, wenn das Elternformular geöffnet wird. Ist der Wert der Variablen an verschiedenen Stellen gesetzt, verwendet 4D den zuletzt geladenen Wert. Beim Laden gilt folgende Reihenfolge:

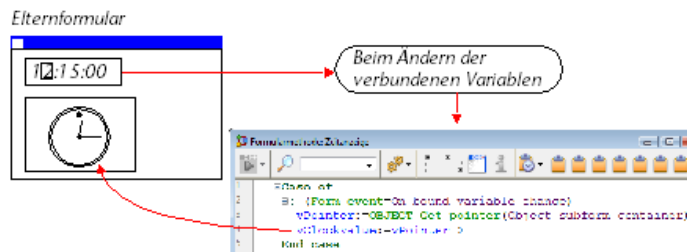
- 1- Objektmethode des Unterformulars
- 2- Formularmethode des Unterformulars
- 3- Objektmethode des Elternformulars
- 4- Formularmethode des Elternformulars

Beim Ausführen des Elternformulars muss der Entwickler über passende Formularereignisse für die Synchronisierung der Variablen sorgen. Zwei Arten von Interaktionen können auftreten: Formular zu Unterformular und Unterformular zu Formular.

Inhalt des Unterformulars updaten

Fall1: Der Wert der Variablen des Elternformulars wird geändert, diese Änderung muss sich im Unterformular widerspiegeln. In unserem Beispiel ändert sich die Zeit der Variablen ParisTime auf 12:15:00, entweder weil der Benutzer das eingegeben hat oder weil die Zeit dynamisch aktualisiert wird, z.B. über die 4D Funktion **Current time**.

In diesem Fall müssen Sie das Formularereignis **On bound variable change** verwenden. Sie müssen es in den Eigenschaften des Unterformulars markieren. Es wird in der Formularmethode des Unterformulars erzeugt.



Das Formularereignis **On bound variable change** wird erzeugt:

- sobald der Variablen des Elternformulars ein Wert zugewiesen wird, selbst wenn derselbe Wert erneut zugewiesen wird.
- wenn das Unterformular zur aktuellen Formularseite oder zur Seite 0 gehört.

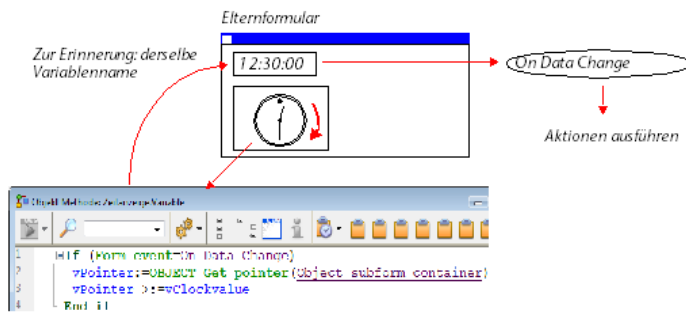
Wir empfehlen, wie im obigen Beispiel, die 4D Funktion **OBJECT Get pointer** zu verwenden, da sie statt der Variablen einen Zeiger auf den Unterformular Container übergibt. Denn in dasselbe Elternformular lassen sich mehrere Unterformulare einfügen. So enthält z.B. ein Fenster mit verschiedenen Zeitzonen mehrere Uhren. In diesem Fall lässt sich nur über einen Zeiger feststellen, welcher Unterformular Container am Ursprung des Ereignisses ist.

Inhalt des Elternformulars updaten

Fall 2: Der Inhalt des Unterformulars wird geändert und diese Änderung muss sich im Elternformular widerspiegeln. Stellen Sie sich in unserem Beispiel vor, dass der Benutzer im Unterformular die Zeiger der Uhr manuell bewegen kann.

In diesem Fall müssen Sie den Wert des Objekts der Variablen des Unterformular Containers im Elternformular zuweisen. Wie im vorigen Beispiel empfehlen wir, die 4D Funktion **OBJECT Get pointer** mit dem Selektor **Object subform container** zu verwenden, der einen Zeiger auf den Unterformular Container zurückgibt.

Weisen Sie den Wert der Variablen zu, wird das Formularereignis **On Data Change** in der Objektmethode des Unterformular Containers im Elternformular generiert, so dass Sie jede Art von Aktion ausführen können. Dieses Ereignis muss in den Eigenschaften des Unterformular Containers ausgewählt sein.



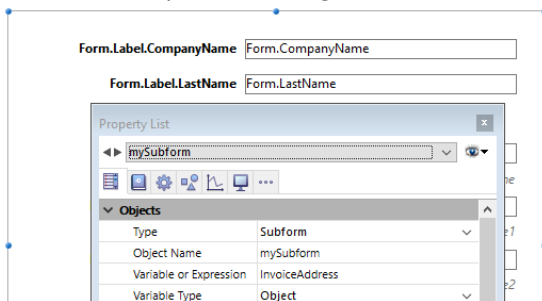
Objektmethode der Variablen vClockvalue im Unterformular

Hinweis: Bewegen Sie die Zeiger der Uhr manuell, wird das Formulereignis On Data Change ebenfalls in der Objektmethode der Variablen *clockValue* im Unterformular generiert.

An Unterformular gebundenes Objekt verwenden

4D bindet automatisch ein *Objekt* (**C_OBJECT**) an jedes Unterformular. Der Inhalt dieses Objekts kann im Kontext des Unterformulars gelesen bzw. geändert werden, so dass Sie Werte in einem lokalen Kontext gemeinsam nutzen können. Das Objekt lässt sich automatisch erstellen oder ist die Variable des Elterncontainers, wenn sie explizit benannt und als Objekt typisiert ist (siehe unten). In allen Fällen wird das Objekt von der Funktion **Form** zurückgegeben, die sich direkt im Unterformular aufrufen lässt (ein Zeiger wird ignoriert). Da Objekte immer über Referenz übergeben werden, wird eine Änderung, die der Benutzer im Unterformular macht, automatisch im Objekt selbst gespeichert.

Zum Beispiel werden in Ihrem Unterformular Feldbezeichnungen im verbundenen Objekt gespeichert, so dass Sie verschiedene Sprachen anzeigen können:

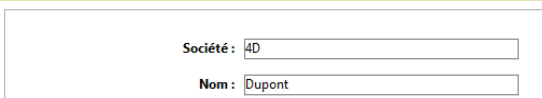


Um die Bezeichnungen vom Unterformular zu ändern, weisen Sie einfach dem Objekt *InvoiceAddress* Werte zu:

```

C_OBJECT($lang)
$lang:=New object
If(<>lang="fr")
    $lang.CompanyName:="Société :"
    $lang.LastName:="Nom :"
Else
    $lang.CompanyName:="Company:"
    $lang.LastName:="Name:"
End if
InvoiceAddress.Label:=$lang

```



Weitere Informationen dazu finden Sie unter der Funktion **Form**.

Formularübergreifende Programmierung

Die Kommunikation zwischen dem Elternformular und den Instanzen des Unterformulars erfordern u.U. mehr als den Austausch eines Werts über die zugewiesene Variable. Sie wollen die Variablen in den Unterformularen gemäß den Aktionen updaten, die im Elternformular ausgeführt werden und umgekehrt. Nehmen wir wieder das Beispiel Unterformular mit dynamischer Uhr. Wir wollen nun eine oder mehrere Alarmzeiten pro Uhr setzen.

Dafür wurden in 4D folgende Funktionalitäten geschaffen:

- Verwendung des Parameters "Unterformular" mit der 4D Funktion **OBJECT Get name** um das Objekt im Unterformular anzugeben und die 4D Funktion **OBJECT Get pointer**.
- Aufrufen des Objekts Container aus dem Unterformular über den 4D Befehl **CALL SUBFORM CONTAINER**,
- Ausführen einer Methode im Kontext des Unterformulars über den 4D Befehl **EXECUTE METHOD IN SUBFORM**.

Funktionen **Object get pointer** und **Object get name**

Die 4D Funktion **OBJECT Get pointer** akzeptiert zusätzlich zum Selektor Object subform container einen Parameter, der angibt, in welchem Unterformular nach dem Objekt gesucht werden soll, dessen Name im zweiten Parameter angegeben ist. Diese Syntax lässt sich nur verwenden, wenn der Selektor Object named übergeben ist.

Beispiel:

Die folgende Anweisung:

```
$ptr:=OBJECT Get pointer(Object_named;"MyButton";"MySubForm")
```

... findet einen Zeiger auf die Variable "MyButton", die im Unterformular "MySubForm" liegt. Über diese Syntax lässt sich vom Elternformular aus auf jedes Objekt zugreifen, das im Unterformular gefunden wird.

Beachten Sie auch den 4D Befehl **OBJECT Get name** über den Sie den Namen des Objekts mit Fokus wiederfinden können.

Befehl CALL SUBFORM CONTAINER

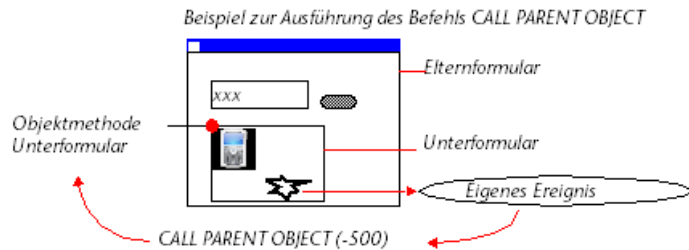
Über den 4D Befehl **CALL SUBFORM CONTAINER** kann eine Unterformular Instanz ein Ereignis an das Objekt Unterformular-Container senden, welches es dann im Kontext des Elternformulars bearbeiten kann. Das Ereignis wird in der Objektmethode des Containers empfangen. Es kann am Ursprung jedes Ereignisses sein, das vom Unterformular gefunden wird (Klick, Drag-and-Drop, etc.).

Der Code des Ereignisses ist nicht begrenzt, er kann z.B. 20000 oder -100). Sie können Code von vorhandenen Ereignissen, z.B. 3 für *On Validate*, oder einen eigenen Code verwenden. Im ersten Fall können Sie nur Ereignisse verwenden, die Sie in der Eigenschaftenliste für Unterformular Container markiert haben. Im zweiten Fall darf der Code keinem vorhandenen Formularereignis entsprechen. Wir empfehlen, einen negativen Wert zu verwenden, um sicherzustellen, dass 4D diesen Code nicht in zukünftigen Versionen verwendet.

Die ausführliche Beschreibung dieses Befehls finden Sie unter **CALL SUBFORM CONTAINER**.

Befehl EXECUTE METHOD IN SUBFORM

Über den Befehl **EXECUTE METHOD IN SUBFORM** kann ein Formular oder eins seiner Objekte die Ausführung einer Methode im Kontext der Unterformular Instanz anfordern, was Zugriff auf Variablen, Objekte, etc. des Unterformulars gibt. Diese Methode kann auch Parameter empfangen. Nachfolgendes Schema zeigt die Funktionsweise:










Die ausführliche Beschreibung dieses Befehls finden Sie unter **EXECUTE METHOD IN SUBFORM**.

Befehl GOTO OBJECT

Der 4D Befehl **GOTO OBJECT** sucht nach dem Zielobjekt im Elternformular, auch wenn es über ein Unterformular ausgeführt wird.

Ausgabeformulare und Berichte

-  Überblick
-  Ausgabeformulare
-  Formulare für gedruckte Berichte
-  Steuerlinien für Ausgabe verwenden
-  Beispielbericht
-  Serienbriefe erstellen
-  Etiketten erstellen

Ausgabeformulare verwenden Sie für zwei Aufgaben: Zum Auflisten von Datensätzen auf dem Bildschirm und zum Drucken.

In bestimmten Fällen können Sie schneller einen Bericht mit dem Schnellberichteditor erstellen. Der Formulareditor bietet Ihnen jedoch bessere Möglichkeiten für das endgültige Aussehen Ihres Berichtes. Weitere Informationen dazu finden Sie in Kapitel **Schnellberichte**.

Hinweis: Nur Tabellenformulare lassen sich als Ausgabeformular verwenden. Projektformulare eignen sich nur zur Verwendung als Detailformular.

Bereiche des Ausgabeformulars

Ausgabeformulare sind in folgende Bereiche mit unterschiedlichen Eigenschaften gegliedert:

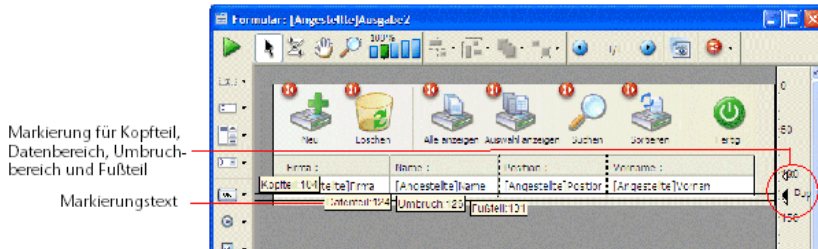
- **Kopfteil:** Enthält den Berichtstitel und Spaltenüberschriften, sowie Schaltflächen oder Objekte zum Verwalten des Formulars
- **Datenbereich:** Enthält den Inhalt des Berichtes
- **Fußteil:** Enthält Schaltflächen oder Summenberechnungen, basierend auf allen Datensätzen im Bericht
- **Umbruchbereich:** Enthält Text oder Grafiken, die nach der Liste von Datensätzen erscheinen und Summenberechnungen, basierend auf allen Datensätzen oder Untergruppen von Datensätzen

Erstellen Sie ein Listenformular mit dem Formularassistenten, gibt dieser die Bereiche automatisch für Sie vor. Er setzt den Formulartitel, d.h. den Tabellennamen und die Namen der Datenfelder in den Kopfteil und eine oder mehrere Schaltflächen (der genaue Inhalt des Kopfteil-Bereichs richtet sich nach den Vorgaben, die Sie im Formularassistenten gewählt haben). Die von Ihnen ausgewählten Datenfelder erscheinen im Datenbereich nebeneinander. Der Formularassistent erstellt einen kleinen Umbruchbereich, schreibt in diesen aber nichts hinein.

Wenn Sie das Formular im Formulareditor öffnen, können Sie die Größe jedes Bereichs und seinen Inhalt verändern, im Umbruchbereich Objekte hinzufügen sowie weitere Umbruchbereiche zur Berechnung von Summen erstellen.

Die einzelnen Bereiche werden durch Ausgabesteuerlinien begrenzt. Ziehen Sie diese Steuerlinien bei gedrückter Maustaste nach oben oder unten, um die Bereiche zu vergrößern oder zu verkleinern.

Die folgende Abbildung zeigt ein Ausgabeformular, das mit dem Formularassistenten erstellt wurde (XP Vorlage):



Hinweis: Sie können Markierung und Markierungstext im Formulareditor ein- und ausblenden. Weitere Informationen dazu finden Sie im Kapitel **Formulareditor**.

Die horizontalen Linien unterteilen den Bericht in die Bereiche Kopfteil, Datenbereich, Umbruch und Fußteil. Der Bereich vom Kopf des Formulars bis zur Kopfteil-Linie ist der Kopfteilbereich, der Bereich zwischen Kopfteil-Linie und Datenbereich-Linie ist der Datenbereich. Der Fußteilbereich reicht von der Umbruchlinie, bezeichnet mit U0, bis zur Fußteilbereichs-Linie. Sie können die Größe jedes Bereichs verändern, wenn Sie bei gedrückter Maustaste die Markierung für den entsprechenden Bereich vertikal ziehen.

Wird dieses Formular ausgeführt, sieht es wie folgt aus:

Firma :	Name :	Position :	Vorname :
4D Deutschland GmbH	Müller	Technik	Tom
4D Deutschland GmbH	Huber	Vertrieb	Josef
4D Deutschland GmbH	Schuster	Presse	Karl
4D Deutschland GmbH	Kisch	Buchhaltung	Johanna
4D Deutschland GmbH	Schmitt	Technik	Barbara
4D Deutschland GmbH	Redlin	Ingenieur	Martin
4D Deutschland GmbH	Hansen	Vertrieb	Henry
4D Deutschland GmbH	Knoll	Vertrieb	Johann

Der Datenbereich wird dynamisch erweitert, wenn das Fenster angepasst wird, während Kopfteil und Fußteil ihre feste Größe behalten. Ein Formular, das Datensätze auf dem Bildschirm anzeigt, kann im Kopfteil weitere klickbare, nicht eingebare Objekte enthalten wie Schaltflächen, hierarchische Listen, etc..

In einem gedruckten Bericht enthält der Kopfteil meistens das Datum, die Uhrzeit, einen Titel und Spaltenbezeichnungen. Die Datensätze erscheinen im Datenbereich. Im Umbruchbereich kann eine berechnete Summe erscheinen. Im Fußteil steht oft die Seitenzahl.

Die folgende Abbildung zeigt die unterschiedlichen Bereiche in einem gedruckten Bericht:

Kopfteilbereich		Vertrieb		
Datum: 13.07.13		Uhrzeit: 11:17		
		Produkt	Funktion	Umsatz
Datenbereich	Vorteil	Milch	Zugab	10.000,00
	Zugab	Milch	Teigab	25.000,00
	Teigab	Milch	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
	Zugab	Teigab	Zugab	10.000,00
Umsatzbericht				Gesamtumsatz: 350.000,00
Fußteilbereich		Umsatzbericht		Seite 1

Für Zwischensummen und andere Berechnungen kann der Bericht weitere Umbruchbereiche enthalten. Ein Bericht kann auch zusätzliche Kopfteilbereiche im Bericht selbst enthalten, die zur Festlegung von Untergruppen dienen. Ein Beispiel dazu finden Sie im Abschnitt **Steuerlinien für Ausgabe verwenden**.

Standardfunktionen

Ein Ausgabeformular listet Datensätze auf. Obwohl sich jedes Formular als Ausgabeformular verwenden lässt, haben die meisten Ausgabeformulare diese Eigenschaften:

- Jede Zeile ist ein Datensatz
- Jede Spalte ist ein Feld oder eine Variable.
- Über jeder Spalte steht ein Spaltentitel. Über die Splitter zwischen den Titelbereichen lassen sich die Spalten größer oder kleiner ziehen
- Kopf- und Fußteil können Schaltflächen, PopUp-Menüs o.ä. enthalten.

In der Liste scrollen

Ein Ausgabeformular hat standardmäßig eine waagrechte und eine senkrechte Bildlaufleiste. Sie können Datensätze durchblättern, indem Sie auf die Pfeile der Leisten, auf die Bildlaufleisten selbst klicken oder bei gedrückter Maustaste den Bildlaufanzeiger bewegen. Sie können auch die Sondertasten **Bild Pfeil nach oben** bzw. **Bild Pfeil nach unten** verwenden, um seitenweise zu blättern oder **Pos1** bzw. **Ende**, um an den Anfang bzw. das Ende der Liste zu springen. Enthält Ihr Ausgabeformular mehr Datenfelder, als im Fenster angezeigt werden können, können Sie auch den Bildschirmausschnitt mit der waagrechten Bildlaufleiste am unteren Fensterrand verschieben.

Spalten anpassen

Sie können die Spalten im Ausgabeformular größer oder kleiner machen — außer diese Möglichkeit wurde im Formulareditor entfernt. Setzen Sie dazu den Mauszeiger auf den Spaltentrenner; der Cursor wird zu einem doppelseitigen Pfeil, so dass Sie die Spaltenbreite verändern können. Durch Ziehen mit gedrückter Maustaste können Sie die Spalte größer oder kleiner ziehen:



Datensätze markieren

Sie markieren einen Datensatz im Ausgabeformular durch Anklicken. Sie können einen Datensatz, zwei oder mehrere zusammenhängende bzw. nicht zusammenhängende Datensätze markieren. Sie können dann beispielsweise Operationen ausführen wie Löschen oder als Auswahl anzeigen.

- Um einen Datensatz zu markieren, klicken Sie auf einen Datensatz im Ausgabeformular oder drücken Sie die Pfeiltaste nach oben bzw. nach unten. Der angeklickte Datensatz wird markiert. Über die Pfeiltasten wird der erste bzw. der letzte Datensatz der Liste markiert.
- Um mehrere zusammenhängende Datensätze zu markieren, klicken Sie auf den ersten Datensatz, den Sie markieren wollen. Halten Sie die **Umschalttaste** gedrückt und klicken auf den letzten gewünschten Datensatz (drücken Sie die **Pfeiltaste nach oben bzw. nach unten**). Alle Datensätze zwischen dem ersten und dem letzten angeklickten Datensatz werden markiert.
- Um mehrere nicht-zusammenhängende Datensätze zu markieren, klicken Sie auf den ersten Datensatz, den Sie markieren wollen. Halten Sie unter Windows die **Strg-Taste**, auf Macintosh die **Befehlstaste** gedrückt und klicken einen bzw. mehrere weitere Datensätze an. Jeder angeklickte Datensatz wird markiert.
- Um alle Datensätze der aktuellen Auswahl im Ausgabeformular zu markieren, wählen Sie im Menü **Bearbeiten** den Befehl **Alles auswählen** oder die standardmäßige Tastenkombination **Strg+A** unter Windows, **Befehl+A** auf Mac OS.

Um alle Datensätze der Tabelle zu markieren, wählen Sie im Menü **Datensätze** den Befehl **Alle anzeigen**, bevor Sie im Menü **Bearbeiten** den Befehl **Alles auswählen** aufrufen.

Erforderliche Farbeinstellung für Formularobjekte, die markiert werden

In Übereinstimmung mit Oberflächenstandards müssen zum Erkennen von Markieren für Textfarbe und Hintergrundfarbe in Objekten, die im Hauptteil des Formularbereichs liegen, folgende Einstellungen angewendet werden:

- **Automatisch** für die Textfarbe jedes Objekts
- **Transparent** für die Hintergrundfarbe jedes Objekts
- **Automatisch** für das Rechteck, das unter dem Hauptteil des Formularbereichs liegt

Diese Einstellungen werden in den Standard Ausgabeformularen automatisch angewandt.

Datensätze hinzufügen und ändern

Sie können Datensätze direkt im Ausgabeformular hinzufügen und ändern. Das Ausgabeformular ist besonders hilfreich, um zusammenhängende Datensätze zu ändern, da sie gleichzeitig am Bildschirm zu sehen sind.

Sie können allerdings nur Datenfelder der aktuellen Tabelle ändern oder eingeben. Es ist nicht möglich, Daten in Variablen, Datenfeldern anderer Tabellen oder Unterformularen einzugeben oder zu ändern.

Im Ausgabeformular können Sie:

- Durch Anklicken Datenfelder auswählen und bearbeiten (Sie müssen zweimal klicken, wenn der Datensatz noch nicht ausgewählt ist),
- Mit der **Tabulator-, Eingabetaste** oder der **Zeilenschaltung** von einem Datenfeld bzw. einem Datensatz zum nächsten springen
- Über den Menübefehl **Datensätze>Neuer Datensatz in Liste** einen neuen Datensatz hinzufügen

Drücken Sie die **Tabulator-, Eingabetaste** oder **Zeilenschaltung** oder klicken Sie auf ein anderes Feld, werden die Änderungen gesichert.

Leerzeilen verwalten

4D zeigt im Listenmodus bei Bedarf nach dem letzten Datensatz Leerzeilen an, so dass das gesamte Fenster ausgefüllt ist:

Vorname :	Nachname :	Position :	Webseite :
Michael	Meier	Administrator	http://www.de.4d.com
Alexander	Baumann	Angestellte	www.meinewebsite.de
Franz	Braun	Angestellter	
Susi	Meier	Lager	
Birgitt	Vogel	Vertreib	
Annette	Schwab	Marketing	
Michaela	Pesch	Dokumentation	
Anne	Krimm	Buchführung	
Klaus	Ben	Angestellte	
Richard	Beck	Angestellte	

Erscheint dieses Fenster, wird das Formularereignis On Display Detail auch für jede Leerzeile der Liste erzeugt. In diesem Fall gibt es keinen aktuellen Datensatz: Die 4D Funktion **Record number** gibt -1 zurück und die 4D Funktion **Selected record number** gibt keinen signifikanten Wert zurück. Über die 4D Funktion **Displayed line number** erfahren Sie die Nummer der aktuell angezeigten Zeile.

Ausgabeformulare anpassen

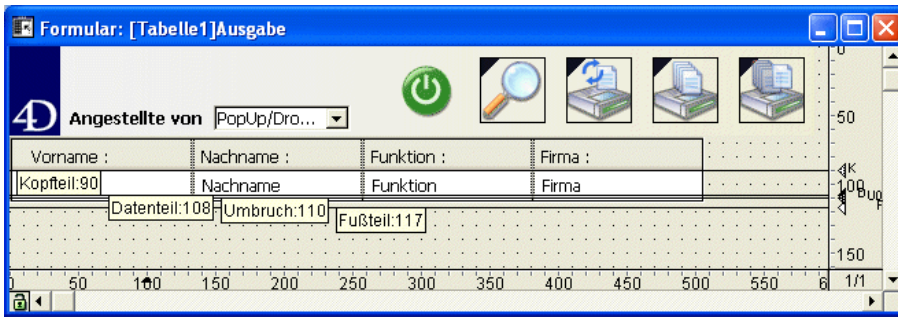
Ein mit dem Formularassistenten erstelltes Listenformular ist zum Auflisten der Datensätze gut geeignet. Bei Bedarf können Sie leicht Veränderungen vornehmen:

- Sie ersetzen Datenfelder mit Variablen und fügen Methoden hinzu
- Sie benutzen Optionen für die Oberfläche, Schriftattribute, Füll-, Rahmen- oder Farboptionen, um die Darstellung des Formulars zu verbessern
- Sie verändern die Breite der Datenfelder oder Variablen
- Sie stellen die Steuerlinien für die Ausgabe ein
- Sie fügen im Kopfteilbereich ein Bild ein
- Sie fügen im Kopfteil oder Fußteil Objektvariablen hinzu.

Die folgende Abbildung zeigt ein gestaltetes Ausgabeformular zur Personalverwaltung:

Vorname :	Nachname :	Funktion :	Firma :
Tom	Banana Soft	Mechaniker	4D
Johanna	M.A.D.E.C	Vertreib	4D
Henry	Biro	Vertrieb	4D
Barbara	Schmitt	Vertreib	4D

Der Kopfteil enthält ein Bild sowie eine DropDown-Liste zum Auswählen einer Firma. Auf dem Bildschirm erscheinen dann die dazugehörigen Angestellten. Im Formulareditor sieht dieses Formular folgendermaßen aus:

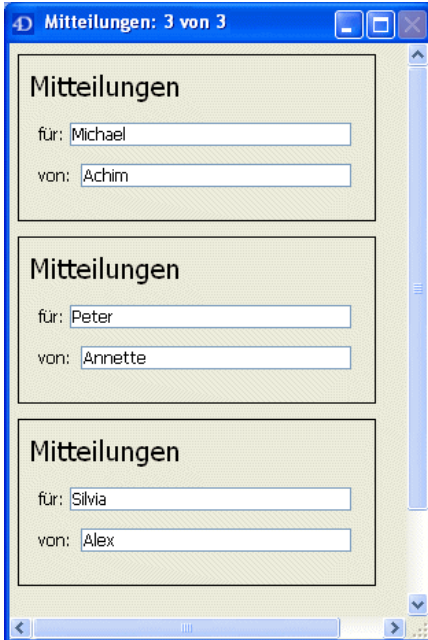


Sie können die Steuerlinien versetzen, z.B. den Kopfteil vergrößern, um ihn an das Bild anzupassen.

Pro Datensatz mehrere Zeilen anzeigen

In 4D können Sie für jeden Datensatz mehrere Zeilen verwenden. Dazu erweitern Sie den Datenbereich, so dass mehr als eine Zeile Platz hat. Dann empfiehlt es sich auch, Datenfelder und Datensätze durch grafische Elemente, wie Linien und Kästchen, zu trennen.

Hier ein Beispiel für ein Ausgabeformular zum Verwalten von Telefonanrufen. Es verwendet zwei Zeilen pro Datensatz:



Objekte sichtbar

Die Eigenschaft **Sichtbar** ist für alle Objekte im Formulareditor verfügbar, wenn das Formular als Liste angezeigt wird. Sie erscheint in der Eigenschaftsliste unter der Gruppe "Anzeige". Es gibt drei Möglichkeiten:

- Immer sichtbar
- Wenn Datensatz ausgewählt
- Wenn Datensatz nicht ausgewählt

Diese Eigenschaft gilt nur für Zeichen von Objekten im Hauptteil eines Listenformulars. Sie gibt an, ob das Objekt gezeichnet wird, wenn der Datensatz in Bearbeitung gewählt bzw. nicht gewählt ist.

So können Sie ausgewählte Datensätze nicht nur mit Farben, sondern auch mit anderen visuellen Attributen darstellen:



4D berücksichtigt diese Eigenschaft nicht, wenn das Objekt über den 4D Befehl **OBJECT SET VISIBLE** ausgeblendet ist; in diesem Fall bleibt das Objekt unsichtbar, unabhängig ob die Datensätze ausgewählt oder nicht ausgewählt sind.

Erscheinen nach dem letzten Datensatz Leerzeilen (siehe oberen Abschnitt "Leerzeilen verwalten"), zeichnet 4D die Objekte mit der Eigenschaft **Immer sichtbar** oder **Wenn Datensatz nicht ausgewählt** auch in jeder Leerzeile.

Formulare für gedruckte Berichte

In 4D können Sie einen Bericht entweder mit einem Formular oder mit dem Schnellberichteditor (siehe [Schnellberichte](#)) erstellen.

Es gibt verschiedene Vorteile, ein Formular für einen Bericht zu verwenden: Sie können grafische Elemente einsetzen und entsprechend setzen, Methoden zum Ausführen von Berechnungen anwenden und Kopfteile anlegen, um Unterabschnitte in einem Bericht mit Umbrüchen entsprechend hervorzuheben. Formularberichte sind über den Benutzerformular-Editor auch vom Benutzer anpassbar (siehe [Benutzerformulare](#)).

Ein Formular zum Erstellen von Berichten eignet sich besonders für folgende Fälle:

- Format ohne Spalten
- Unterformulare anzeigen
- eingebettete Grafiken anzeigen
- Spezifische grafische Elemente, wie z.B. Haarlinien, anzeigen

Im Gegensatz zu Formularen, die auf dem Bildschirm angezeigt werden, können gedruckte Berichte am Ende den Umbruchbereich und für Hinweise, die am Ende jeder Seite erscheinen sollen, den Fußteilbereich verwenden.

Hinweis: Der Auswahltyp **Listboxen** ist besonders geeignet, um Datensatzlisten anzuzeigen bzw. zu drucken.

Berichtstypen

Gedruckte Berichte mit Spalten

Gedruckte Berichte enthalten üblicherweise Informationen in einzelnen Spalten. Sie veröffentlichen beispielsweise Listen mit Telefonnummern, Preisen, Ergebnissen, Spezifikationen oder Artikeln.

Für einen Bericht in Tabellenform wählen Sie im Formularassistenten als Formulartyp **Listenformular zum Drucken**.

Eine gedruckte Liste enthält, ähnlich wie auf dem Bildschirm angezeigte Datensätze, Spalten mit Informationen. Sie kann auch einen Spaltentitel sowie grafische Elemente enthalten, um den Bericht anschaulicher zu machen. Der Umbruchbereich wird einmal am Ende des Berichtes gedruckt, hier lassen sich Gesamtsummen berechnen. Beispiele hierzu sehen Sie in den nachfolgenden Absätzen.

Berichte mit einem Datensatz pro Seite

Sie wollen immer nur einen Datensatz pro Seite ausdrucken. Sie wollen z.B. eine Datenbank für Rechnungen einsetzen, um für Ihre Datensätze von jeder Rechnung eine Kopie zu drucken. Sie können auch einen Serienbrief drucken, wo pro Seite jeweils nur bestimmte Felder geändert werden (siehe [Serienbriefe erstellen](#)).

Dafür wählen Sie im Formularassistenten als Formulartyp **Detail-Formular zum Drucken**.

Ziehen Sie die Steuerlinie **H** für den Kopfteilbereich an den oberen Rand der Seite und ordnen die Datenfelder und andere Elemente des Bericht darunter an.

Zeigt Ihr Formular Datensätze in einem Unterformular an, stellen Sie sicher, dass das Unterformular mit festem Druckrahmen gedruckt wird, damit die Datensätze nicht über mehrere Seiten reichen. Weitere Informationen dazu finden Sie im Abschnitt [Unterformulare als Liste](#) unter **Drucken**.

Ziehen Sie die Steuerlinien für den Datenbereich D, Umbruchteil U0 und den Fußteilbereich F an den unteren Rand der Seite, so dass nur ein Datensatz pro Seite gedruckt wird.

Unterformulare verwenden

Ein typisches Beispiel sind Rechnungen. Eine Rechnung enthält in der Regel einen Namen, eine Rechnungsanschrift, Lieferanschrift, eine Auflistung von Teilen oder durchgeführten Dienstleistungen und eine Summe.

Eine Datenbank für Rechnungen enthält ein Ausgabeformular zum Drucken von einer Rechnung pro Seite. Eine Rechnung holt sich z.B. Informationen aus zwei Tabellen: Eine Tabelle [Auftrag] mit Kundeninformationen, wie Rechnungs- und Lieferanschrift und eine Tabelle [Zeileneinträge] mit Zeileneinträgen. Die Summe des Auftrages wird berechnet und in der Tabelle [Auftrag] gespeichert.

Das Rechnungsformular wird in der Tabelle [Auftrag] erstellt und enthält einen Unterformular-Bereich für die Zeileneinträge. Dieser Unterformular-Bereich wird beim Drucken erweitert, um alle Zeileneinträge zu drucken, auch wenn die Rechnung dadurch eine zweite Seite benötigt. Weitere Informationen dazu finden Sie im Abschnitt [Unterformulare als Liste](#) unter **Drucken**.

Der Datenbereich erstreckt sich über die ganze Seite. Für jeden Auftrag wird eine Rechnung gedruckt. Im Bereich für Rechnungszeilen werden so viele Zeileneinträge, wie benötigt, aufgelistet.

Berichte mit Textfeld

Bei vielen Anwendungen kann der Benutzer in ein Textfeld oder eine Variable Hinweise oder Beschreibungen eingeben. In ähnlicher Weise können Berichte Fotos oder Grafiken aus Bildfeldern oder Variablen enthalten.

Diese Elemente können im Detail-Bereich als erweiterte Bereiche gedruckt werden. Sie können beim Drucken entsprechend erweitert werden, so dass alle Daten Platz haben. Weitere Informationen dazu finden Sie im Abschnitt [Drucke variable Größe](#).

Hinweis: Dieses Prinzip gilt auch für 4D Write Pro Bereiche.

Eigene Adressetiketten

Für besondere Adressetiketten können Sie ein eigenes Ausgabeformular anlegen, in dem Sie grafische Elemente, alle verfügbaren Schriftarten und Variablen benutzen können.

Hinweis: Oft lassen sich Adressetiketten leichter und schneller mit dem integrierten **Etiketteneditor** drucken. Weitere Informationen dazu finden Sie im Abschnitt **Über den Befehl PRINT LABEL**.

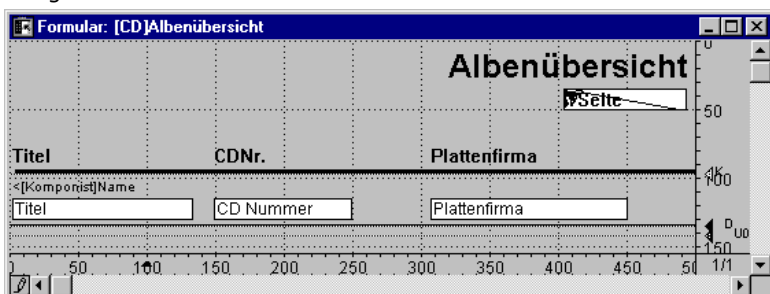
Umbrüche verwenden

4D kann Berichte drucken, mit Umbrüchen und Umbruch-Kopfteilen arbeiten. Ein Umbruch entsteht, wenn Sie die Datensätze sortieren.

Nehmen wir an, Sie haben eine CD-Sammlung, die Sie mit einer 4D Anwendung verwalten und Sie möchten eine Liste ausdrucken, wo die Informationen nach Künstlern sortiert sind. Wenn Sie die Datensätze nach Künstlern sortieren, gehören die Datensätze in bestimmte Gruppen. Der Umbruch erfolgt jeweils, wenn der letzte Datensatz einer Gruppe gedruckt ist. Die folgende Abbildung zeigt den Bericht in der Vorschau auf dem Bildschirm:

Albenübersicht		
Titel	CDNr.	Plattenfirma
THE PLANETS	023	Deutsche Grammophon
THE PLANETS, SUITE DE BALLET, OP. 10	250	Enigma Classics
Harry Belafonte		
ALL TIME GREATEST HITS VOL.1	025	BMG Music
PARADISE IN GAZANKULU	028	Capitol Records
Hector Berlioz		
SYMPHONIE FANTASTIQUE	192	Enigma Records
Hiroshima		
ANOTHER PLACE	036	
Huey Lewis And The News		
FORE!	031	Chrysalis Records
INXS		
KICK	203	Atlantic Recording Corporation
Jean-Luc Ponty		
COSMIC MESSENGER	029	Atlantic Recording Corporation
Jerry Goodman		
ARIEL	073	Private Music, Inc.
Joaquin Rodrigo		
CONCIERTO DE ARANJUEZ	233	Enigma Classics
Joe Sample		
SPELLBOUND	179	Warner Bros. Records Inc.
Johann Sebastian Bach		
BACHBUSTERS	080	TELARC DIGITAL
BRANDENBURG CONCERTOS NOS. 1, 2, & 3	125	Enigma Records
BRANDENBURG CONCERTOS NOS. 4, 5 & 6	163	Enigma Records
ORGAN FAVOURITES	248	Enigma Classics
Johann Strauss, Jr.		
STRAUSS FESTIVAL VOLUME 1: FAMOUS WALTZES, PC096		Enigma Records
STRAUSS FESTIVAL VOLUME 2	159	Enigma Records
Johannes Brahms		
HUNGARIAN DANCES NOS. 1-21	243	Enigma Classics
HUNGARIAN DANCES NOS. 1-21	146	Enigma Records
SYMPHONY NO.4 ACADEMIC FESTIVAL OVERTURE	127	Enigma Records
Joni Mitchell		
COURT AND SPARK	108	Asylum Records
Joseph Haydn		

4D bietet Möglichkeiten, um die Informationen attraktiver darzustellen. Die folgende Abbildung zeigt dieses Formular im Designmodus:



Hinweis: Wenn Sie einen Bericht mit Umbrüchebenen und Kopfteilen erstellen wollen, müssen Sie zuerst den Umbruchvorgang aktivieren. Weitere Informationen dazu finden Sie im unteren Abschnitt "Umbruchvorgang starten".

Der Umbruch-Kopfteil wird einmal vor jeder Gruppe von Datensätzen gedruckt. Ein Umbruch wird einmal vor der Gruppe von Datensätzen gedruckt. In der Abbildung oben wird der Umbruch als "Umbrüchebene 1" bezeichnet und der Umbruch-Kopfteil als "Kopfteil Umbrüchebene 1", weil der Umbruch als Ergebnis der ersten Sortierung der Datenfelder erfolgt.

Sie können bis zu neun Umbrüchebenen einsetzen. Wenn Sie die 4D Funktion **Subtotal** verwenden, um den Umbruchvorgang auszulösen, müssen Sie die Sortierung über ein Datenfeld mehr vornehmen, als die Anzahl der Umbrüche beträgt. In diesem Fall müssen Sie über zwei Datenfelder sortieren, wenn Sie eine Umbrüchebene einsetzen. Verwenden Sie drei Umbrüchebenen, müssen Sie über vier Datenfelder sortieren.

In einem Berichtformular benötigen zusätzliche Umbrüchebenen und Kopfteil-Umbrüchebenen zusätzliche Bereiche. Das erreichen Sie durch Hinzufügen eigener Steuerlinien für die Ausgabe. Weitere Informationen dazu finden Sie im Abschnitt "Zusätzliche Steuerlinien erstellen" unter **Steuerlinien für Ausgabe verwenden**.

Dieser Abschnitt beschreibt, wie Sie Berichte mit Umbrüchen erstellen. Ein ausführliches Beispiel finden Sie im Abschnitt **Beispielbericht**.

Umbruchvorgang starten

Damit 4D Bereiche für den Umbruch-Kopfteil druckt, Zwischensummen addiert oder andere Aspekte des Umbruchvorganges durchführt, müssen Sie zuerst den Umbruchvorgang im Berichtformular starten. Gehen Sie folgendermaßen vor:

- Setzen Sie die 4D Funktion **Subtotal** in ein Objekt oder eine Formularmethode
- Führen Sie die 4D Befehle **BREAK LEVEL** und **ACCUMULATE** vor dem Ausdrucken des Berichtes aus

Verwenden Sie die 4D Funktion **Subtotal** müssen Sie die Datensätze über ein Sortierfeld mehr als die Anzahl der Umbrüchebenen sortieren. Wenn Sie z.B. in Ihrem Bericht zwei Umbrüchebenen verwenden, müssen Sie über drei Datenfelder sortieren.

Berichte mit Zwischensummen

Dieser Abschnitt beschreibt im einzelnen, wie sich die Sortierfolge auf Berichte auswirkt und wie Sie zusätzliche Umbruchbereiche benutzen, um Zwischensummen zu erstellen.

4D kann automatisch Gesamt- und Zwischensummen berechnen und drucken. Die folgende Abbildung zeigt einen Bericht mit berechneten Zwischensummen pro Kunde und einer Gesamtsumme am Berichtende:

Zic Industrie Zac		Verkaufsbericht		
Kunden	Produkt	Datum	Preis	
American Data	Schulung 76	05.06.03	1.000,00	
American Data	STS 0000	15.03.03	3.000,00	
American Data	STS Service	15.04.03	1.500,00	
Summe für American Data			5.500,00	
Horizon	ATN 700	14.09.03	2.000,00	
Horizon	ATN 850	18.10.03	5.000,00	
Horizon	STS 0000	15.11.03	3.000,00	
Horizon	STS 3000	07.05.03	1.800,00	
Summe für Horizon			11.800,00	
James System	ATN 500	22.06.03	1.700,00	
Summe für James System			1.700,00	
Omni Data	ATN 850	05.10.03	5.000,00	
Omni Data	ATN 850	30.01.03	5.000,00	
Omni Data	Schulung 76	07.07.03	1.000,00	
Omni Data	Schulung 76	05.02.03	1.000,00	
Omni Data	STS 0000	14.02.03	3.000,00	
Omni Data	STS 3000	25.09.03	1.800,00	
Omni Data	STS 3000	01.10.03	1.800,00	
Omni Data	STS 3000	25.04.03	1.800,00	
Omni Data	STS 3000	22.06.03	1.800,00	
Omni Data	STS 4000	17.05.03	3.400,00	
Omni Data	STS 4000	05.08.03	3.400,00	
Omni Data	STS 4000	14.07.03	3.400,00	
Summe für Omni Data			32.400,00	
Thomas Info System	ATN 700	27.01.03	2.000,00	
Thomas Info System	STS 2000	22.06.03	1.400,00	
Summe für Thomas Info System			3.400,00	
Gesamt			54.800,00	

Diese Datensätze sind nach Kunden und Produkt sortiert. Nachdem die Datensätze jedes Kunden gedruckt sind, wird die Zwischensumme für jeden Kunden berechnet und gedruckt. Nachdem die Datensätze des gesamten Berichtes gedruckt sind, wird die Gesamtsumme berechnet und gedruckt.

4D weiß, wann die Zwischensumme zu berechnen und zu drucken ist. Es wurde angewiesen, diesen Vorgang immer dann auszuführen, wenn der Wert im ersten Datenfeld der Sortierung wechselt, d.h. wenn ein Umbruch erfolgt. Die Zwischensumme des Kunden wird in Umbrüchebene 1 berechnet, denn sie basiert auf einem Wechsel in der ersten Sortierebene, in diesem Fall im Datenfeld Kunden. Die Gesamtsumme wird in Umbrüchebene 0 berechnet. Ein Umbruch der Ebene 0 beinhaltet alle Datensätze und erfolgt am Ende des Berichtes.

Die folgende Abbildung zeigt eine andere Berechnung. Das Beispiel verwendet die gleichen Datensätze, hat aber eine andere Sortierfolge und erstellt einen anderen Umbruch. Hier sind die Datensätze nach Produkt und Kunden sortiert. Die Zwischensummen werden berechnet, wenn der Wert im Datenfeld Produkt wechselt. Dies ist weiterhin Umbrüchebene 1, der Umbruch erfolgt jedoch in einem anderen Datenfeld:

Zic Industrie Zac		Verkaufsbericht		
Kunden	Produkt	Datum	Preis	
James System	ATN 500	22.06.03	1.700,00	
Summe für ATN 500			1.700,00	
Horizon	ATN 700	14.09.03	2.000,00	
Thomas Info System	ATN 700	27.01.03	2.000,00	
Summe für ATN 700			4.000,00	
Horizon	ATN 850	18.10.03	5.000,00	
Omni Data	ATN 850	05.10.03	5.000,00	
Omni Data	ATN 850	30.01.03	5.000,00	
Summe für ATN 850			15.000,00	
American Data	Schulung V6	03.06.03	1.000,00	
Omni Data	Schulung V6	07.07.03	1.000,00	
Omni Data	Schulung V6	05.02.03	1.000,00	
Summe für Schulung V6			3.000,00	
American Data	ETS 1000	17.03.03	2.000,00	
Horizon	ETS 1000	17.11.03	3.000,00	
Omni Data	ETS 1000	14.02.03	3.000,00	
Summe für ETS 1000			9.000,00	
Thomas Info System	ETS 2000	22.06.03	1.400,00	
Summe für ETS 2000			1.400,00	
Horizon	ETS 3000	07.05.03	1.800,00	
Omni Data	ETS 3000	25.09.03	1.800,00	
Omni Data	ETS 3000	01.10.03	1.800,00	
Omni Data	ETS 3000	25.04.03	1.800,00	
Omni Data	ETS 3000	22.06.03	1.800,00	
Summe für ETS 3000			9.000,00	
Omni Data	ETS 4000	17.05.03	3.400,00	
Omni Data	ETS 4000	03.08.03	3.400,00	
Omni Data	ETS 4000	14.07.03	3.400,00	
Summe für ETS 4000			10.200,00	
American Data	ETS Service	17.04.03	1.500,00	
Summe für ETS Service			1.500,00	
Gesamt			54.800,00	

Zusätzliche Umbrüchebenen

Sie können zusätzliche Summenberechnungen durchführen, indem Sie eine weitere Sortierebene und eine weitere Umbrüchebene hinzufügen. Die folgende Abbildung zeigt Datensätze des Verkaufs, sortiert nach Kunden, Produkt und Verkäufer. Die Summenberechnungen zeigen zwei Sätze von Zwischensummen: eine Zwischensumme pro Kunde und, innerhalb eines Kunden, zusätzlich Zwischensummen pro Produkt. Abschließend berechnet dieser Bericht die Gesamtsumme für die gesamte Firma. Dies sind Beispiele für Berechnungen, durchgeführt auf Umbrüchebene 2, auf Umbrüchebene 1 und auf Umbrüchebene 0.

Zic Industrie Zac		Verkaufsbericht		
Kunden	Produkt	Datum	Preis	
Omni Data	ATN 850	30.01.03	5.000,00	
Omni Data	ATN 850	05.10.03	5.000,00	
Summe für ATN 850			10.000,00	
Omni Data	ETS 1000	14.02.03	3.000,00	
Summe für ETS 1000			3.000,00	
Omni Data	ETS 3000	22.06.03	1.800,00	
Omni Data	ETS 3000	25.04.03	1.800,00	
Omni Data	ETS 3000	01.10.03	1.800,00	
Omni Data	ETS 3000	25.09.03	1.800,00	
Summe für ETS 3000			7.200,00	
Omni Data	ETS 4000	14.07.03	3.400,00	
Omni Data	ETS 4000	17.05.03	3.400,00	
Summe für ETS 4000			6.800,00	
Omni Data	Schulung V6	05.02.03	1.000,00	
Omni Data	Schulung V6	07.07.03	1.000,00	
Summe für Schulung V6			2.000,00	
Summe für Omni Data			29.000,00	
Thomas Info System	ATN 700	27.01.03	2.000,00	
Thomas Info System	ATN 700	22.06.03	2.000,00	
Summe für ATN 700			4.000,00	
Thomas Info System	ETS 4000	08.09.03	3.400,00	
Summe für ETS 4000			3.400,00	
Summe für Thomas Info System			7.400,00	
Gesamt			36.400,00	

Die Berechnungen der Zwischensumme werden nur für die Gruppe von Datensätzen durchgeführt, die vor dem Umbruch stehen. Eine Zwischensumme wird z.B. für jedes Produkt berechnet, welches an diesen Kunden verkauft wurde. Die Zwischensumme für den Kunden wird für alle Produkte berechnet, die an diesen Kunden verkauft wurden.

Hinweis: Weitere Informationen dazu finden Sie im Abschnitt "Zusätzliche Steuerlinien erstellen" unter **Steuerlinien für Ausgabe verwenden**.

Summenberichte

Sie können Berichte erstellen, in denen nur zusammenfassende Informationen gedruckt werden. Solch ein Bericht zeigt nur die Zwischensummen und Gesamtsummen mit erklärendem Text. Die folgende Abbildung zeigt hierzu ein Beispiel:

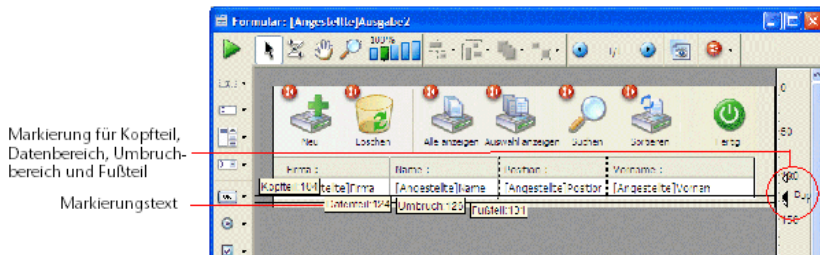
Zic Zac Industries	
Verkaufsbericht Summen	
Summe für ATN 850	15.000,00
Summe für STS 1000	9.000,00
Summe für STS 3000	9.000,00
Summe für STS 4000	10.200,00
Summe für Schulung V6	3.000,00
Summe für Omni Data Service	46.200,00
Summe für ATN 700	4.000,00
Summe für STS 4000	10.200,00
Summe für Thomas Info Systems	14.200,00
Total	60.400,00
1. Dezember 2003	

In diesem Bericht sind die Datensätze nach Kunden, Produkt und Datum sortiert und die Berechnungen erfolgen, während die Umbrüche entsprechend der Sortierfolge erstellt werden. Der Datenbereich für jeden Datensatz wird nicht gedruckt. Die Datensätze werden lediglich für die Werte der Berechnungen benutzt. Beachten Sie, dass 4D für die Berechnung dieser Zahlen beim Drucken des Berichtes vom ersten zum letzten Datensatz geht. Sie erhalten den Summenbericht, wenn Sie die Steuerlinie für den Datenbereich auf die Steuerlinie für den Kopfteilbereich setzen und somit keinen Platz zum Drucken von Details lassen.

Sie können mit 4D in Listen zusätzliche Berechnungen durchführen, wie Zwischensummen, Durchschnitts-, Maximal- und Minimalwerte, im Fußteilbereich pro gedruckter Seite Gesamtsummen sowie Durchschnittswerte. Dazu verwenden Sie die Summenfunktionen von 4D (siehe **Berechnungen**).

Steuerlinien für Ausgabe verwenden

Sie steuern den Kopfteilbereich, den Daten- und Umbruchbereich und den Fußteilbereich über die Steuerlinien für die Ausgabe im Formulareditor. Ziehen Sie diese Steuerlinien nach oben oder unten, um einen Bereich zu vergrößern oder zu verkleinern. Jedes Objekt, das Sie in einen dieser Bereiche setzen, wird an der entsprechenden Stelle angezeigt oder gedruckt.



Steuerlinien und Bereiche

Nachfolgend wird beschrieben, wie diese Bereiche arbeiten, wenn das Formular im Anwendungsmodus bzw. über die integrierten Funktionen des Designmodus angezeigt oder gedruckt wird.

- **Kopfteilbereich:** Der Kopfteil erscheint oben auf jeder Bildschirmseite und wird auf jeder Seite eines Berichtes oben gedruckt. Er liegt über der Steuerlinie H. Sie können ihn größer oder kleiner ziehen. Hier setzen Sie Spaltennamen, Hinweise, zusätzliche Informationen oder eine Grafik, wie z.B. ein Firmenlogo. Sie können auch aktive Objekte im Kopfteil von Ausgabeformularen verwenden, die als Unterformulare, im Anzeigefenster der Datensätze oder über die 4D Befehle **DISPLAY SELECTION** und **MODIFY SELECTION** angezeigt werden.

Alle aktiven Objekte sind möglich. Das gilt für:

- Schaltfläche, 3D Schaltfläche, invertierte Schaltfläche, Bildschaltfläche
- Combo Box, PopUp-Menü/DropDown-Liste, hierarchisches PopUp-Menü oder vom Typ Bild
- Rollbarer Bereich, hierarchische Liste, Listbox
- Optionsfeld, 3D Optionsfeld, Optionsbild
- Kontrollkästchen, 3D Kontrollkästchen
- Laufbalken, Lineal, Halbkreissskala

Sie können den eingefügten Schaltflächen **Standardaktionen** wie **Bestätigen**, **Abbrechen** (Listen angezeigt über die 4D Befehle **DISPLAY SELECTION** und **MODIFY SELECTION**) oder **Automatischer Splitter** zuweisen.

Folgende Formularereignisse sind möglich: On Load, On Clicked, On Header, On Printing Footer, On Double Clicked, On Drop, On Drag Over, On Unload. Beachten Sie, dass die Formularmethode mit dem Ereignis On Header nach Aufruf der Objektmethoden dieses Bereichs aufgerufen wird.

- **Datenbereich:** Der Datenbereich wird für jeden Datensatz einmal auf dem Bildschirm dargestellt und in einem Bericht für jeden Datensatz einmal gedruckt. Der Datenbereich liegt zwischen der Steuerlinie H des Kopfteilbereiches und der Steuerlinie D des Datenbereiches. Sie können diesen Bereich über die Steuerlinie D größer oder kleiner ziehen. Alles, was Sie in den Datenbereich setzen, wird einmal pro Datensatz angezeigt oder gedruckt. In der Regel setzen Sie in diesen Bereich Datenfelder oder Variablen, so dass diese Information in jedem Datensatz angezeigt oder gedruckt wird. Es sind aber auch andere Elemente möglich.
- **Umbruchbereiche:** Die Umbruchbereiche erscheinen einmal am Ende einer Liste von Datensätzen und werden, nachdem die Datensätze eines Berichtes gedruckt sind, einmal ausgedruckt. Der Umbruchbereich liegt zwischen der Steuerlinie D für den Datenbereich und der Steuerlinie U0 für den Umbruch. Ihr Bericht kann weitere Umbruchbereiche enthalten. Sie können diesen Bereich über die Steuerlinie U0 größer oder kleiner ziehen. Im Umbruchbereich zeigen Sie Informationen, die nicht zu den Datensätzen gehören, wie Anweisungen, aktuelles Datum oder Uhrzeit, eine Linie oder andere grafische Elemente, die den Bildschirmbereich abschließen. In einem gedruckten Bericht benutzen Sie den Umbruchbereich für Berechnungen oder zum Drucken der Gesamtsumme.
- **Fußteilbereich:** Der Fußteilbereich wird auf dem Bildschirm unter der Liste mit den Datensätzen angezeigt. Er wird immer auf jeder Seite eines Berichtes gedruckt. Der Fußteilbereich liegt zwischen der Steuerlinie des Umbruchbereiches U0 und der Steuerlinie F für den Fußteilbereich. Sie können diesen Bereich über die Steuerlinie F größer oder kleiner ziehen. In den Fußteil setzen Sie Grafiken, Seitenzahlen, aktuelles Datum oder Text, der auf jeder Seite eines Berichtes am unteren Rand erscheinen soll. In Ausgabeformularen auf dem Bildschirm enthält der Fußteilbereich Schaltflächen für Benutzeroptionen, wie z.B. Suchen oder Sortieren, Ausdrucken von Berichten oder aktuellen Bericht ablegen. Alle aktiven Objekte sind möglich.

Sobald Sie ein beliebiges Formular zur Ausgabe verwenden, entweder auf dem Bildschirm oder zum Drucken, werden die Ausgabe-Steuerlinien berücksichtigt und an den zugewiesenen Stellen angezeigt bzw. ausgedruckt. Diese Linien werden auch wirksam, wenn Sie ein Formular als Listenformular in einem Unterformularbereich verwenden.

Wenn Sie ein Formular zur Eingabe benutzen, haben die Ausgabe-Steuerlinien keine Bedeutung.

Methoden, die mit Objekten in diesen Bereichen verbunden sind, werden ausgeführt, sobald diese Bereiche gedruckt oder angezeigt werden, solange die entsprechenden Ereignisse aktiviert sind. Eine Objektmethode im Kopfteilbereich wird z.B. ausgeführt, wenn das Ereignis On Header auftritt.

Für zusätzliche Umbruchbereiche und Kopfteilebereiche eines Berichtes können Sie weitere Steuerlinien einfügen. Hier drucken Sie Zwischensummen und andere Berechnungen eines Berichtes oder zeigen Informationen an. Weitere Informationen dazu finden Sie im unteren Abschnitt **Zusätzliche Steuerlinien erstellen**.

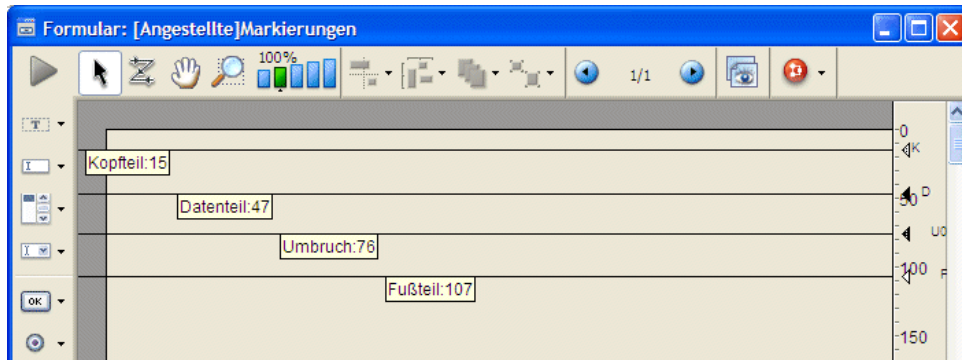
Steuerlinien für Ausgabe bewegen

Sie bestimmen die Größe der einzelnen Bereiche, d.h. Kopfteil-, Daten-, Umbruchteil- und Fußteilbereich, durch Bewegen der Steuerlinien.

Die Steuerlinien für die Ausgabe erscheinen im Formular als gestrichelte Linien. Jede Steuerlinie zeigt im senkrechten Lineal eine Marke in Form eines Dreiecks mit einem Buchstaben oder eine Bezeichnung unter der Linie. Wollen Sie eine Steuerlinie verschieben, klicken Sie auf das Dreieck oder die Bezeichnung und ziehen diese an die neue Position. Die Bezeichnung wird standardmäßig angezeigt, sie können diese bei Bedarf ausblenden. Dann erscheint der Text nur, wenn Sie auf das Dreieck klicken. Weitere Informationen dazu finden Sie im Abschnitt "Elemente im Formulareditor ein-/ausblenden" unter **Formulareditor**).

Die Bezeichnung gibt Name und Position der Marke in Bezug auf den Ursprung des Formulars an. Verschieben Sie die Marke, erscheint sofort der Wert der neuen Position. Über die Bezeichnung können Sie Steuerlinien auch bewegen, wenn die Lineale nicht angezeigt werden.

Die folgende Abbildung zeigt Marken und ihre Bezeichnungen:



Um eine Steuerlinie zu bewegen, ziehen Sie die Marke oder die Bezeichnung nach oben oder unten.

- Ziehen Sie eine Markierung bei gedrückter **Umschalttaste**, werden alle Steuerlinien unterhalb der Markierung bewegt. Wollen Sie z.B. sämtliche Steuerlinien zusammen ziehen, halten Sie die **Umschalttaste** gedrückt und ziehen die Markierung des Kopfteilbereiches. Wollen Sie alle Steuerlinien mit Ausnahme der Kopfteillinie bewegen, ziehen Sie die Markierung des Datenbereichs mit gedrückter **Umschalttaste**.
- Wollen Sie Objekte unter der Steuerlinie verschieben oder Objekte über der Steuerlinie vergrößern und gleichzeitig mit der Linie verschieben, wählen Sie die betreffenden Objekte vor dem Verschieben aus.

Sie können die Reihenfolge der Steuerlinien nicht verändern. Versuchen Sie z.B., die Steuerlinie für den Fußteil über die Steuerlinie für den Umbruchbereich hinaus zu ziehen, stoppt die Operation, sobald die Linie für den Umbruchbereich erreicht ist.

Sie können jedoch Bezeichnungen und Steuerlinien übereinanderlegen. Sobald Sie eine Bezeichnung über eine andere legen, wird dieser Bereich auf Null reduziert und erscheint nicht im Bericht. Wollen Sie z.B. im Umbruchbereich nichts drucken, ziehen Sie diese Steuerlinie bei gedrückter Maustaste auf die Linie für den Datenbereich. Dann weist 4D keinen Bereich für den Umbruch zu und Sie können den gesamten Platz auf der Seite für Ihren Bericht nutzen.

Wollen Sie im Datenbereich nichts drucken, ziehen Sie bei gedrückter Maustaste die Steuerlinie für den Datenbereich auf die Linie für den Kopfteilbereich. Benötigen Sie keinen Kopfteilbereich, ziehen Sie diese Steuerlinie bei gedrückter Maustaste im Formular ganz nach oben auf den Punkt 0.

Warnung: Aktive Objekte (Datenfelder oder Variablen), die im Datenteil von Formularen liegen, dürfen nicht in Kopf- bzw. Fußteil hineinragen, denn dann werden sie beim Ausführen des Formulars nicht angezeigt.

Zusätzliche Steuerlinien erstellen

Die Berichtbeispiele in diesem Abschnitt enthalten Umbrüchebenen und Umbruch-Kopfteile. Um hierfür Bereiche zu haben, erstellen Sie zusätzliche Steuerlinien.

Der Formulareditor beginnt immer mit den Original-Steuerlinien mit den Bezeichnungen K, D, U0.

U0 steht für "Umbruch auf Ebene 0". Ebene Null bezieht alle Datensätze mit ein; der Umbruch erfolgt, nachdem alle Datensätze gedruckt sind. Zusätzliche Steuerlinien für Umbrüche sind mit Nummern versehen. So steht die Steuerlinie **U1** Steuerlinie für "Umbruch auf Ebene 1". Ein Umbruch der Ebene 1 erfolgt, nachdem alle Datensätze, die nach dem ersten sortierten Datenfeld gruppiert sind, gedruckt wurden:

Bezeichnung	Erläuterung	Gedruckt nach Gruppen erzeugt durch:
B1	Umbruch bei Ebene 1	Erstes sortiertes Datenfeld
B2	Umbruch bei Ebene 2	Zweites sortiertes Datenfeld
B3	Umbruch bei Ebene 3	Drittes sortiertes Datenfeld

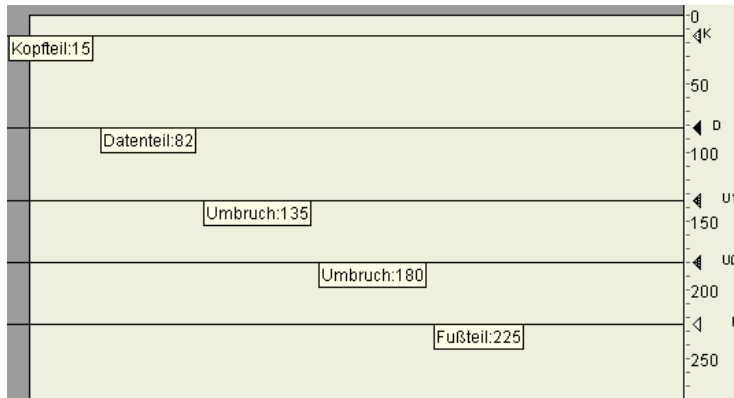
K steht für "Kopfteil", der oben auf jeder Seite gedruckt wird. Zusätzliche Steuerlinien für Kopfteile hängen mit den Umbrüchen zusammen. **K1** steht für "Kopfteil auf Ebene 1". Ein Kopfteil der Ebene 1 wird genau vor den Datensätzen gedruckt, die über das erste sortierte Datenfeld gruppiert sind.

Bezeichnung	Erläuterung	Gedruckt nach Gruppen erzeugt durch:
H1	Kopfteil bei Ebene 1	Erstes sortiertes Datenfeld
H2	Kopfteil bei Ebene 2	Zweites sortiertes Datenfeld
H3	Kopfteil bei Ebene 3	Drittes sortiertes Datenfeld

Sie erstellen zusätzliche Steuerlinien, indem Sie unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** gedrückt halten, während Sie auf die entsprechende Markierung klicken. Eine Umbruch-Steuerlinie verwenden Sie, um einen Umbruch für die entsprechende Ebene zu erstellen. Eine Steuerlinie für den Umbruch-Kopfteil verwenden Sie, um für die entsprechende Ebene einen Umbruch-Kopfteil zu erstellen. Die neue Linie liegt hinter der vorhandenen Steuerlinie; um diese sichtbar zu machen, müssen Sie die vorhandene Steuerlinie bei gedrückter Maustaste verschieben.

Verwenden Sie die Funktion **Subtotal** zum Auslösen der Umbruchbearbeitung, müssen Sie die Datensätze über ein Sortierfeld mehr als die Anzahl der Umbrüchebenen sortieren. Wenn Sie z.B. in Ihrem Bericht zwei Umbrüchebenen verwenden, müssen Sie über drei Datenfelder sortieren. Wenn Sie in einem der Umbruchbereiche nichts ausdrucken müssen, können Sie seine Größe auf Null reduzieren. Dazu ziehen Sie seine Markierung auf eine der anderen Steuerlinien. Haben Sie mehr Sortierebenen als Umbruchbereiche, wird der letzte Umbruchbereich beim Druck wiederholt.

Die folgende Abbildung zeigt zusätzliche Steuerlinien:



Steuerlinien entfernen

Um selbst angelegte Steuerlinien für Umbruch oder Umbruch-Kopfteil zu löschen, klicken Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** auf die entsprechende Steuerlinie bzw. die Bezeichnung dieser Steuerlinie.

4D löscht die Steuerlinie und nummeriert bei Bedarf die verbleibenden Linien um.
Die Original-Steuerlinien K, D, U0 und F können Sie nicht löschen.

Beispielbericht

Dieser Abschnitt beschreibt einen Beispielbericht und zeigt, wie der fertige Bericht im Formulareditor mit einem Formular verknüpft wird, sowie die Methoden, die das Drucken steuern. Der fertige Bericht sieht folgendermaßen aus:

Verkaufsbericht im Detail				02/12/2003 7:44 PM
American Data				
<i>Produkt</i>	<i>Deliefert</i>	<i>Kommentar</i>	<i>Preis</i>	
ATN 700	14.9.03	Konfiguriert für schnellen Zugriff	12.450	
		Summe für ATN 700	12.450	
STS 1000	17.3.03	Benötigt Zusatzleistung für die Version	22.450	
	17.4.03	Plan für weitere Versionen	3.300	
		Summe für STS 1000	24.360	
		Summe für American Data	38.200	
Omni Data Service				
<i>Produkt</i>	<i>Deliefert</i>	<i>Kommentar</i>	<i>Preis</i>	
STS 1000	14.2.03	Testversionen	24.360	
		Summe für STS 1000		
STS 3000	25.4.03	Benötigt Zusatzleistung für die Version	71.025	
	22.6.03	Verwendet die Version nun als Standard	53.252	
	25.9.03	Dritte Version	47.250	
		Summe für STS 3000	171.527	
STS 4000	17.5.03	Benötigt 4000 für spezielle Anwendung	92.450	
	14.7.03	Spezielle Konfiguration	95.420	
		Summe für STS 4000	187.870	
		Summe für Omni Data Service	383.757	
Thomas Info				
<i>Produkt</i>	<i>Deliefert</i>	<i>Kommentar</i>	<i>Preis</i>	
ATN 700	27.1.03	Erstlieferung ATN 700	12.780	
	22.6.03	Wird weitere ATN 700 einsetzen	24.745	
		Summe für ATN 700	37.525	
STS 4000	3.8.03	Benötigt 4000 für neue Datenzentrale	89.740	
		Summe für STS 4000	89.740	
		Summe für Thomas Info	127.265	
		Total	549.222	

Lightwave Computer Corporation 1

Der Kopfteilbereich enthält Datum und Uhrzeit, sowie den Berichtstitel. Die Umbruch-Kopfteile enthalten den Kundennamen und Spaltenüberschriften für die Informationen, die während des ersten Umbruchs geschrieben werden. Die Datenbereiche enthalten die direkt aus den Datensätzen entnommenen Daten. Umbruchebe 2 enthält jeweils die Zwischensumme der Produkte pro Kunde. Umbruchebe 1 enthält die Zwischensumme pro Kunde, Umbruchebe 0 enthält die Gesamtsumme des Berichtes. Der Fußteilbereich enthält die Seitenzahl. Die Sortierung des Berichts enthält eine Ebene mehr als Umbruchebe vorhanden sind. Die Datenfelder sind sortiert nach Kunde, Produkt und Datum.

Formular zum Bericht

Die folgende Abbildung zeigt das Formular für den oben angezeigten Bericht:

The screenshot shows a window titled 'Formular: [Angestellte]Verkaufsbericht'. The main area is a grid with the following structure:

- Header:** A large box for the report title 'Verkaufsbericht' and a box for the date 'Datum' and time 'Zeit'.
- Customer Section:** A box for 'Kunde'.
- Data Table:** A table with columns: 'Produkt', 'Kaufdatum', 'Bemerkung', 'Kaufpreis'.
- Summary Rows:**
 - Summe für <[Verkauf]Produkt> (labeled 'Summe-Prod')
 - Summe für <[Verkauf]Kunde> (labeled 'Summe-Kund')
 - Gesamt (labeled 'Gesamtsumm')
- Footer:** A box for 'B & G Informatik' and a box for 'Seite'.

The right-hand pane contains various controls like arrows and checkboxes, and a ruler at the bottom shows a scale from 0 to 550.

Jede Steuerlinie im Formular definiert das untere Ende eines Bereiches. Alle im Formular gesetzten Informationen werden an der richtigen Stelle im Bericht gedruckt. Der Kopfteil enthält die Elemente, die am Anfang jeder Seite gedruckt werden, der Datenbereich die Elemente, die für jeden Datensatz gedruckt werden, usw.. Die folgende Tabelle zeigt die Bedeutung der verschiedenen Steuerlinien:

Ebene	Erklärung	Auswirkung
K	Kopfteil	Einmal am Anfang jeder Seite gedruckt
K1	Kopfteil Ebene 1	Einmal gedruckt vor jedem Umbruch der Ebene 1
K2	Kopfteil Ebene 2	Einmal gedruckt vor jedem Umbruch der Ebene 2
D	Datenbereich	Einmal gedruckt für jeden ausgewählten Datensatz
U2	Umbruchbereich Ebene 2	Einmal gedruckt bei jedem Umbruch der Ebene 2, d.h. wenn der Wert im zweiten Datenfeld der Sortierung wechselt
U1	Umbruchbereich Ebene 1	Einmal gedruckt bei jedem Umbruch der Ebene 1, d.h. wenn der Wert im ersten Datenfeld der Sortierung wechselt.
U0	Umbruchbereich Ebene 0	Einmal gedruckt am Ende des Berichtes
F	Fußteilmereich	Einmal gedruckt am Ende jeder Seite

Objektmethoden des Berichtes

Die nicht-eingebbaren Objekte im Kopfteil-, Umbruch- und Fußteilmereich werden über Objektmethoden gesteuert.

Hinweis: Nachfolgenden Code können Sie nur bei Objekt- und Formularmethoden verwenden. Für Projektmethoden ist er nicht verwendbar.

Das Datum wird mit dem nicht eingebbaren Objekt *vDatum* vom System übernommen und in den Kopfteilbereich eingefügt mit der Methode:

```
vDatum:=Current date
```

Die Uhrzeit wird mit dem nicht eingebbaren Objekt *vZeit* vom System übernommen und in den Kopfteilbereich eingefügt mit der Methode:

```
vZeit:=Current time
```

Die Zwischensumme für die Verkaufszahlen im Umbruchbereich der Ebene 2 wird in dem Objekt *vSummeProd* berechnet und angezeigt mit der Methode:

```
vSummeProd:=Subtotal(Verkauf)
```

Die Zwischensumme für die Verkaufszahlen im Umbruchbereich der Ebene 1 wird im Objekt *vSummeKund* berechnet und angezeigt mit der Methode:

```
vSummeKund:=Subtotal(Verkauf)
```

Die Gesamtsumme der Verkaufszahlen im Umbruchbereich der Ebene 0 wird mit dem Objekt *vGesamtsumme* berechnet und angezeigt mit der Methode:

```
vGesamtsumme:=Subtotal(Verkauf)
```

Beachten Sie, dass alle drei Objekte die gleiche Berechnung durchführen und trotzdem zu unterschiedlichen Ergebnissen führen. Da sie in unterschiedlichen Umbruchbereichen angeordnet sind, werden sie zu unterschiedlichen Zeiten durchgeführt und führen ihre Berechnungen für unterschiedliche Gruppen von Datensätzen durch. Weitere Informationen dazu finden Sie im Abschnitt "Berichte mit Umbrüchen" unter **Formulare für gedruckte Berichte**.

```
vSeite:="Seite"+String(FORM Get current page)
```

Die 4D Funktion **FORM Get current page** gibt die Seitennummer zurück.

■ Serienbriefe erstellen

Für Serienbriefe benutzen Sie ein Ausgabeformular, das in einem statischen Textbereich Datenfelder oder Variablen enthält. Sie erstellen Dokumente, die – bis auf Namen und Adressen und dynamischen Berechnungen – das gleiche Aussehen haben.

Sie benutzen Serienbriefe z.B. um über Sonderangebote oder geschäftliche Entwicklungen, wie z.B. die Verlagerung von Aktivitäten oder wichtige personelle Veränderungen zu informieren. Eine andere typische Anwendung ist eine Mitteilung an den Kunden, dass noch Rechnungen offen sind. Für die genaue Berechnung der offenen Posten verwenden Sie eine Variable und eine Methode.

Zur Erstellung eines Serienbrief-Formulars verwenden Sie im Formularassistenten in der Dropdown-Liste Formulartyp die Option **Detail-Formular zum Drucken**. Im Formulareditor erstellen Sie einen Textbereich, der den statischen Text und die Datenfelder oder Variablen enthält, die sich in jedem Bericht ändern. Beim Druck werden die Werte der Datenfelder oder Variablen in den Text eingefügt.

Die folgende Abbildung zeigt einen Text mit Datenfeldern und Variablen.

Information from fields

Text

Variable calculates bonus

Display format

Um ein Ausgabeformular für Serienbriefe zu erstellen, fügen Sie im Detail-Bereich einen oder mehrere statische Textbereiche hinzu.

Schreiben Sie den Brief und fügen Sie dort, wo Sie Werte von Datenfeldern oder Variablen einsetzen wollen, zwischen den Zeichen

< (kleiner als) und > (größer als) das Datenfeld und den Variablennamen ein.

Sie können hier ein Datenfeld aus einer beliebigen Tabelle der Datenbank einsetzen. Datenfelder aus der Haupttabelle benötigen keine Angabe der Tabelle; diese können Sie eingeben als <Datenfeld>. Datenfelder aus anderen Tabellen müssen den Tabellennamen enthalten; diese geben Sie ein als <[Tabellenname] Datenfeld>. Wenn das Formular gedruckt wird, wird in <Datenfeld> der Inhalt des Datenfeldes eingesetzt.

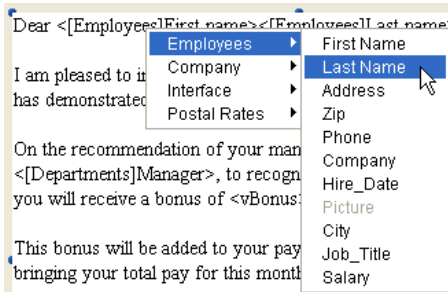
Um die Bezeichnung der Tabelle einzufügen, geben Sie <?[Tabellenname]> oder <?[2]> (Erstellungsnummer der Tabelle) ein. Um die Bezeichnung des Datenfeldes einzufügen, geben Sie <?[Tabellenname]Datenfeldname> oder <?[2]3> (Erstellungsnummer der Tabelle und des Feldes), oder <?3> zum Einfügen eines Feldes der aktuellen Tabelle ein. Weitere Informationen dazu finden Sie im Abschnitt **Referenzen in statischem Text verwenden**.

Für Variablen müssen Sie einen Wert über eine Objekt- oder Formularymethode zuweisen.

Sie können festlegen, wie ein eingebundenes Datenfeld bzw. eine Variable dargestellt wird. Dazu setzen Sie nach dem Namen des Datenfeldes oder der Variablen ein Semikolon, gefolgt von einem Anzeigeformat (siehe **GET LIST ITEM PARAMETER**). Der oben gezeigte Serienbrief zeigt ein Anzeigeformat für die Variable *vbonus*. Die eingebundene Variable <vbonus;#.#0,00+> berechnet die Höhe des Bonus für die betreffende Person und zeigt ihn im Format Euro an.

4D bietet zum Einsetzen von Datenfeldern in Textbereiche ein Tastenkürzel, über das Sie den Namen des Datenfeldes aus einem Dropdown-Menü wählen können. Dazu klicken Sie mit dem Text-Tool, um die Einfügemarke an die gewünschte Stelle für das Datenfeld zu setzen. Setzen Sie den Mauszeiger in den Textbereich und wählen mit gedrückter Maustaste unter Windows die **Alt-Taste**, auf Mac OS die **Wahl Taste**.

4D zeigt ein PopUp-Menü mit Datenfeldern aus der Haupttabelle, aus dem Sie das gewünschte Datenfeld wählen. Um ein Datenfeld aus einer anderen Tabelle der Datenbank zu wählen, wählen Sie bei gedrückter Maustaste unter Windows die Tastenkombination **Umschalttaste+Alt**, auf Mac OS **Wahl Taste+Umschalttaste**. 4D zeigt ein hierarchisches Menü von Tabellen und Datenfeldern der Datenbank.



4D setzt <Datenfeld> oder <[Tabelle]Datenfeld> am Einfügekpunkt im Textbereich.

Wenn der Bericht gedruckt wird, erscheinen die Werte der Datenfelder und der eingebetteten Variablen im Brief:

01/24/05

Dear Jeffrey Muldoon,

I am pleased to inform you that the Sales department has demonstrated a remarkable increase in productivity this year.

On the recommendation of your manager, Ms. Rollings, to recognize and reward your contribution, you will receive a bonus of \$900.

This bonus will be added to your payslip at the end of this month, bringing your total pay for this month to \$4,700.00.

Congratulations and keep up the good work.

Sincerely,

Jeannette Simpson,
Chief Executive Officer

01/24/05

Dear Alan Hull,

I am pleased to inform you that the Marketing department has demonstrated a remarkable increase in productivity this year.

On the recommendation of your manager, Mr. Trump, to recognize and reward your contribution, you will receive a bonus of \$800.

This bonus will be added to your payslip at the end of this month, bringing your total pay for this month to \$4,980.00.

Congratulations and keep up the good work.

Sincerely,

Jeannette Simpson,
Chief Executive Officer

Etiketten erstellen

Sie erzeugen Etiketten entweder mit dem **Etiketteneditor** oder mit einem eigenen Listenformular. Meist reicht der Etiketteneditor aus. In manchen Fällen ist jedoch ein Formular besser geeignet, denn hier können Sie in Etiketten Variablen einsetzen und alle grafischen Hilfsmittel des Formulareditors nutzen.

Haben Sie im Formulareditor ein Formular für Etiketten erstellt, können Sie es auf zwei Arten verwenden:

- Über den 4D Befehl **PRINT LABEL**
Über die Anweisung **PRINT LABEL**([Tabelle]) wird die aktuelle Auswahl von *Tabelle* im aktuellen Listenformular gedruckt. In diesem Fall verwendet 4D die Marker für Breite und Höhe dieses Formulars zum Einrichten des Etikettenformats.
- Damit können Sie die erweiterten Funktionen des Formulareditors und die Layouteinstellungen des Etiketteneditors nutzen.
Der Etiketteneditor berücksichtigt nur die absolute Position der Objekte im Formular sowie gesetzte Ränder. Die Position der Marker wird ignoriert.

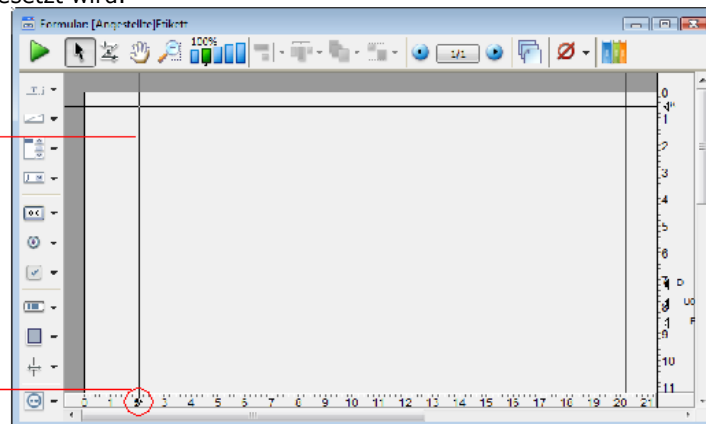
Über den Befehl **PRINT LABEL**

Um ein Listenformular als Etikett anzulegen und über den 4D Befehl **PRINT LABEL** zu drucken:

1. Setzen Sie die Etikettenbreite, indem Sie im horizontalen Lineal des Formulareditors die Markierung für Breite ziehen.
Die Breitenmarkierung bestimmt, wie viele Etiketten der 4D Befehl **PRINT LABEL** auf einer Seite druckt. Die Einstellung der Breite muss mit der Breite Ihrer Etiketten übereinstimmen. Berücksichtigen Sie hierbei auch den linken und rechten Rand für Ihre Etiketten.
Sie können diese Ränder bestimmen, indem Sie den Abstand zwischen den Etiketten berechnen und diese Zahl durch 2 dividieren. Sind Ihre Etiketten z.B. 2,25 Zoll breit mit einem Rand von 0,125 Zoll (1/8 Zoll) auf beiden Seiten des Etiketts, sollten Sie den Marker für die Breite auf 2,625 Zoll (2 3/8 Zoll) setzen, um sicherzustellen, dass der Text korrekt in die Etiketten gesetzt wird.

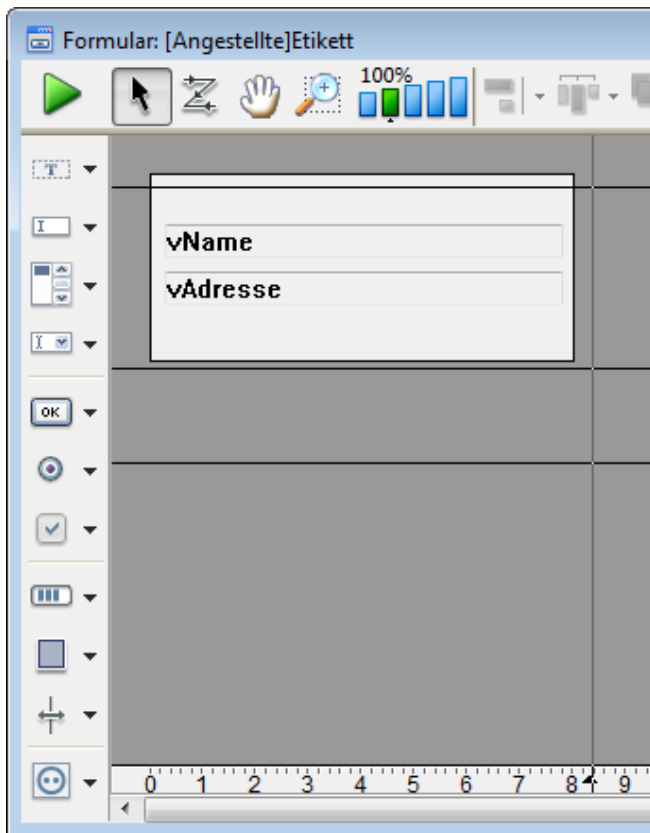
Die Breitenlinie erscheint, wenn Sie auf die Breitenmarkierung klicken

Breitenmarkierung



Hinweis: Um die Einstellung der Breitenmarkierung zu erleichtern, können Sie die Maßeinheiten des Lineals ändern. Hierzu wählen Sie im Menü **Formular** den Befehl **Lineal festlegen**.

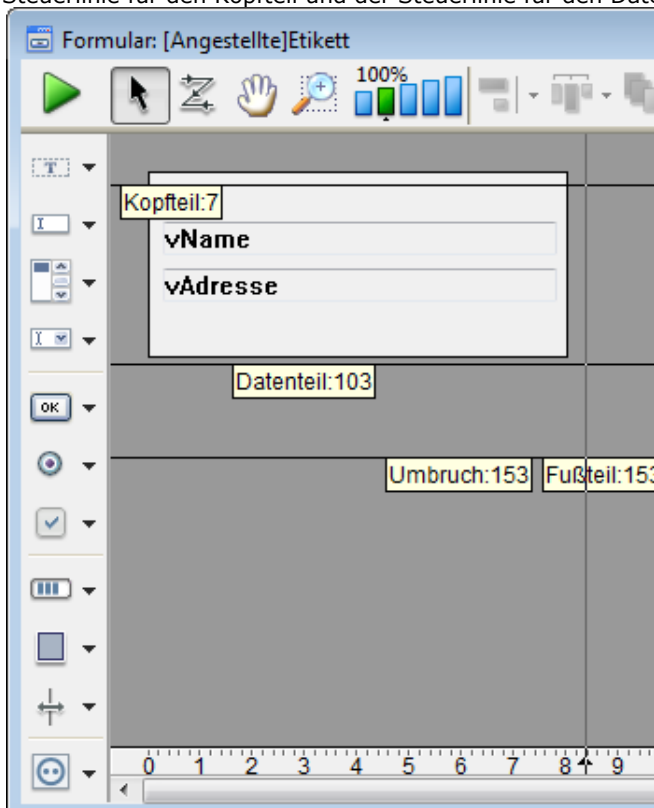
2. Setzen Sie die Objekte in das Etikett.
Etiketten können Datenfelder, aktive Objekte (denen Sie bei Bedarf Objektmethoden zuordnen können), grafische Elemente und statischen Text enthalten. Ein Unterformular kann dagegen nicht im Etikett gedruckt werden.



Dieses Etikett enthält Variablen, d.h. aktive Objekte, deren Werte den Objektmethoden zugeordnet sind. Die Methode für die Variable `vName` verbindet z.B. den Vor- und Nachnamen jeder Person und setzt zwischen beide Namen ein Leerzeichen:

```
vName:=[Kunden]Vorname+" "+[Kunden]Nachname
```

3. Setzen Sie die Objekte im Formular vor die Breitenmarkierung.
Die Etikettenelemente müssen links vor der Markierung liegen und zwischen linkem Rand des Formulars und der Markierung zentriert werden, damit der Abstand auf jeder Seite des Etiketts gleich ist.
4. Stellen Sie die Steuerlinien so ein, dass sich die Steuerlinie für den Kopfteil über dem Etikett und die Steuerlinien für die Bereiche Detail, Umbruch und Fußteil unter dem Etikett befinden.
Die Steuerlinie für den Kopfteil stellen Sie auf Null, die Steuerlinie für den Datenbereich auf die Höhe des Etiketts. Um den Text innerhalb des Etiketts zu zentrieren, zentrieren Sie die Formularelemente zwischen der Steuerlinie für den Kopfteil und der Steuerlinie für den Datenbereich.



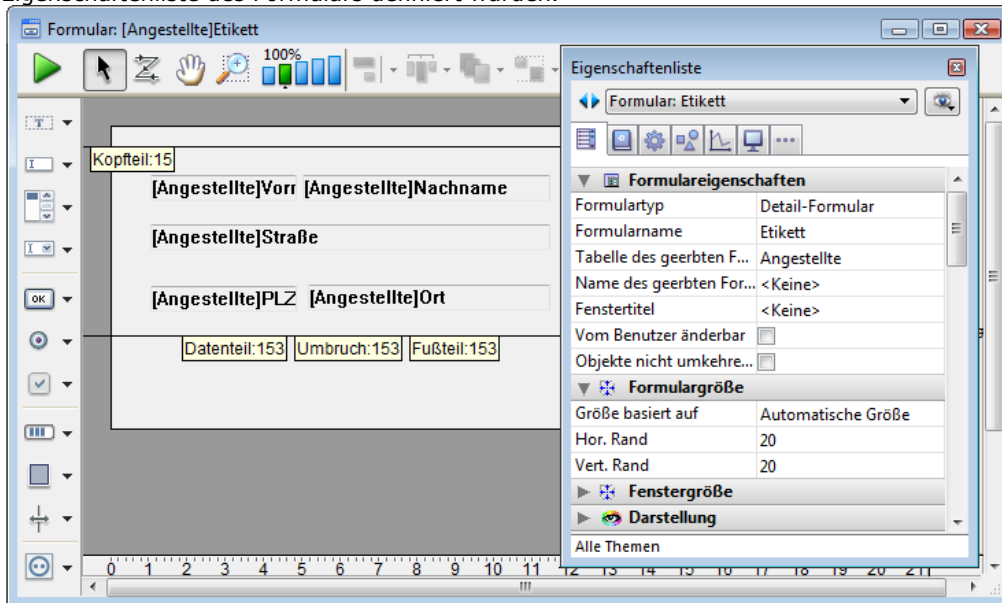
Wenn Sie die Etiketten ausdrucken, erscheint auf den Etiketten alles, was sich zwischen diesen beiden Steuerlinien befindet.

5. Sichern Sie das Formular und drucken Sie die Auswahl über den 4D Befehl **PRINT LABEL**.

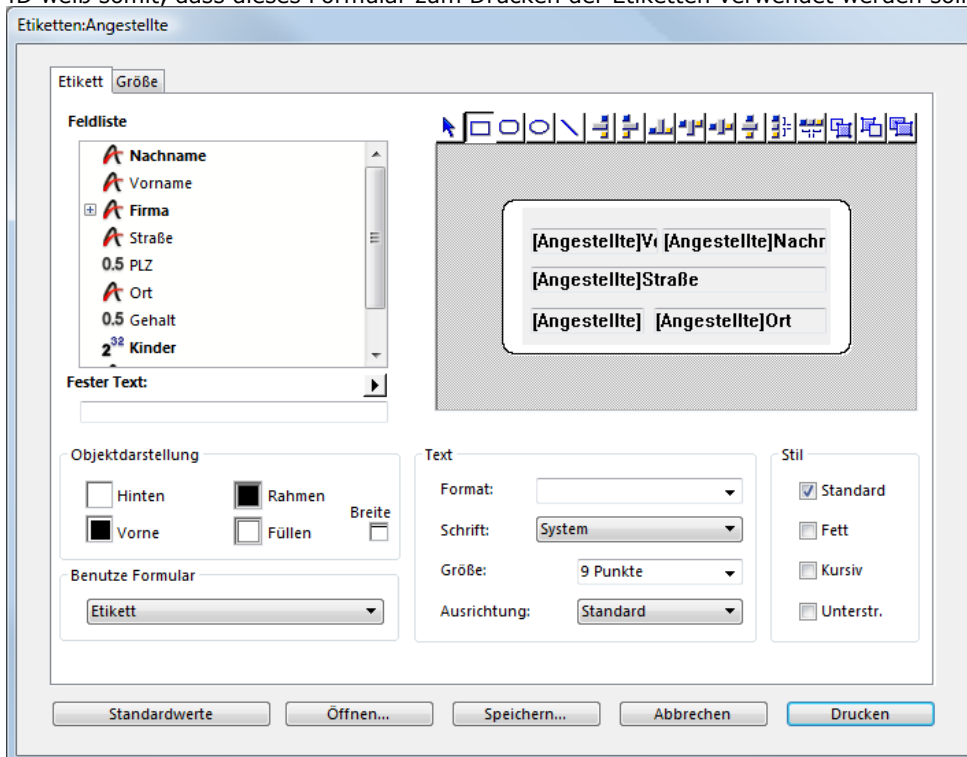
Über den Etiketteneditor

Um ein Listenformular als Etikett anzulegen und über den Etiketteneditor zu drucken:

1. Fügen Sie im Formular die Objekte für das Etikett ein.
2. Setzen Sie jedes Element im Etikett nach folgenden Regeln:
 - Linker und oberer Rand des Etiketts richten sich nach der Position der Elemente in Bezug auf die Koordinaten (0,0) des Formulars.
 - Rechter und unterer Rand jedes Etiketts richten sich nach den Werten **Hor. Rand** und **Vert. Rand**, die in der Eigenschaftsliste des Formulars definiert wurden.








3. Wählen Sie im Fenster **Liste der Tabellen** das Etikettenformular, damit es das aktuelle Ausgabeformular der Tabelle wird. Ist die Tabellenliste nicht geöffnet, wählen Sie im Menü **Datensätze** den Befehl **Tabellenliste**.
4. Wählen Sie im Menü **Bericht** den Befehl **Etiketten**. Auf dem Bildschirm erscheint der Etiketteneditor.
5. Wählen Sie in der Dropdown-Liste „Benutze Formular“ das zuvor angelegte Etikettenformular. 4D weiß somit, dass dieses Formular zum Drucken der Etiketten verwendet werden soll.



6. Sie können weitere Einstellungen festlegen, z.B. die Randbereiche der Etiketten oder eine bestimmte Schriftart für den Text. Weitere Informationen dazu finden Sie im Kapitel **Etiketteneditor**.
7. Klicken Sie auf die Schaltfläche **Drucken**. 4D druckt die aktuelle Auswahl von Datensätzen in Etikettenformular.

Benutzerformulare

-  Überblick über Benutzerformulare (Anwendungsmodus)
-  Benutzerformulareditor
-  Objekte ansehen und bearbeiten
-  Eingabereihenfolge
-  Objektbibliothek verwenden

Überblick über Benutzerformulare (Anwendungsmodus)

In 4D kann der Entwickler Benutzern die Möglichkeit geben, Formulare in der Datenbank zu verändern, um z.B. eigene Berichte zu erstellen, ein spezifisches Logo hinzuzufügen oder ganz allgemein, Formulare an eigene Bedürfnisse anzupassen. Wird das „Benutzerformular“ gesichert, ersetzt es das ursprüngliche Format auf dem Rechner des Benutzers und kann wie jedes andere Formular in 4D verwendet werden.

Änderungen sind hauptsächlich für grafische Aspekte möglich. So kann der Benutzer nicht Objektmethoden, Variablen oder Datenfelder hinzufügen. Dagegen kann er die Eingabereihenfolge verändern oder aktive Objekte aus der Objektbibliothek hinzufügen. Die zugelassenen Operationen legt der Entwickler zuvor im Formulareditor im Designmodus fest.

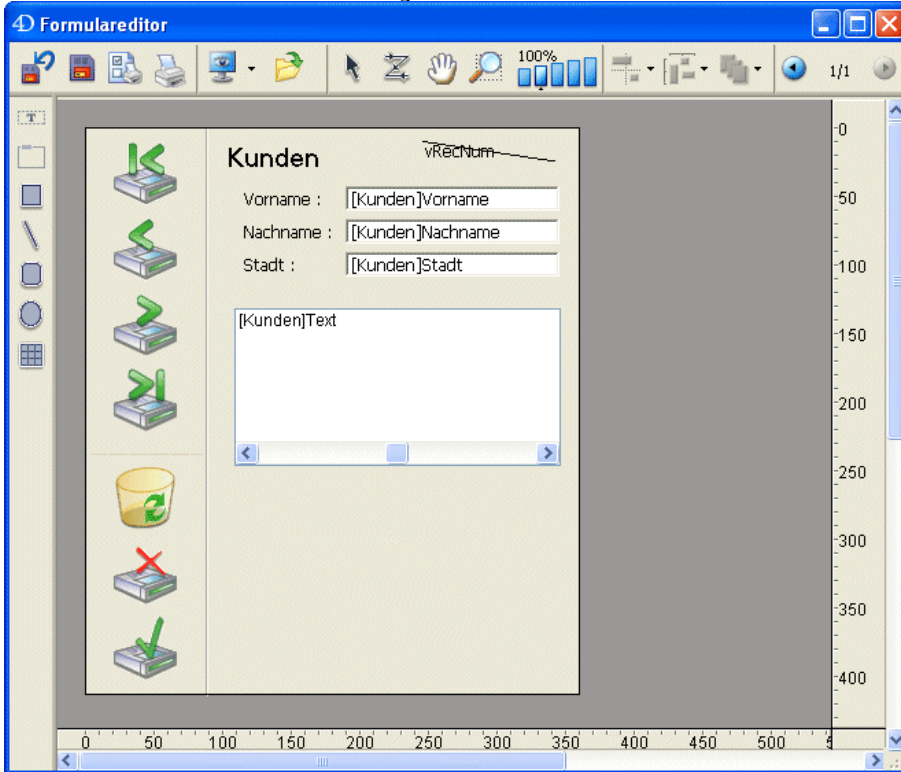
Weitere Informationen zum Einrichten dieses Systems durch den Entwickler finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Benutzerformulare**.

Dieses Kapitel erläutert die Funktionsweise des Editors für Benutzerformulare in der Anwendungsumgebung.

Benutzerformulareditor

Der Benutzerformulareditor erscheint für ein Formular, wenn der 4D Befehl **EDIT FORM** ausgeführt wird und für das Formular, das als Parameter übergeben ist, die Eigenschaft **Vom Benutzer editierbar** markiert ist (siehe Abschnitt "Vom Benutzer editierbar" unter **Formulareigenschaften**). Dieser Editor wird in der Regel über eine Konfiguration der Anwendung oder einen eigenen Befehl aufgerufen.

Der Benutzerformulareditor sieht folgendermaßen aus:



Er verwendet die Standardoberfläche des 4D Formulareditors mit den beiden Werkzeugleisten für Tools und Objekte. Darüberhinaus hat er auch eigene spezifische Funktionen.












Mit diesem Editor ist nur das Menü **Bearbeiten** der Menüleiste verfügbar. Auch eine vereinfachte Eigenschaftsliste sowie ein Kontextmenü sind verfügbar.

Das Auswählen und Bewegen von Objekten – sofern zugelassen – funktioniert wie im standardmäßigen Formulareditor von 4D. Weitere Informationen dazu finden Sie im Kapitel **Formulareditor**.

Werkzeugleiste

Die Werkzeugleiste enthält folgende Optionen:

- **Zurückgehen**: Kehrt zur zuletzt gesicherten Version des Formulars zurück. Klicken Sie auf diese Schaltfläche, erscheint eine Meldung, um die Operation zu bestätigen oder abubrechen.
- **Sichern**: Sichert die Änderungen im Benutzerformular.
- **Druckeinstellungen** und **Drucken**: Über diese Schaltflächen können Sie das Drucken des Benutzerformulars einrichten und starten.
- **Anzeige**: Damit öffnen Sie ein PopUp-Menü, über das Sie Elemente anzeigen oder ausblenden können:
 - **Seite 0**: Blendet die Objekte von Seite 0 in der aktuellen Seite des Formulars ein/aus. Damit können Sie im Formular unterscheiden zwischen den Objekten der aktuellen Seite und den Objekten der Seite 0. In einem mehrseitigen Formular enthält Seite 0 Objekte, die auf allen Seiten erscheinen.
 - **Papier**: Blendet die Ränder der Druckseite ein/aus. Sie erscheinen als graue Linien. Diese Option hat keine Auswirkung, wenn die Option **Begrenzung** ausgewählt ist. Ist das Formular kleiner als die Druckseite, liegen die Seitenränder außerhalb des Formularbereichs und erscheinen deshalb nicht.
 - **Lineal**: Blendet im Formulareditor die Lineale ein bzw. aus.
 - **Markierung**: Blendet die Steuerlinien und zugeordneten Marker ein bzw. aus. Sie zeigen die Begrenzung der verschiedenen Formularbereiche an, also Kopfteil, Datenteil und Fußteil. Marker werden hauptsächlich in Ausgabeformularen verwendet.
 - **Begrenzung**: Blendet die Formularbegrenzungen ein oder aus. Ist diese Option gewählt, erscheint das Formular im Formulareditor mit seinen endgültigen Maßen.

-  **Bibliothek:** Damit laden Sie – sofern vorhanden – die Bibliothek mit Objekten, die im Formular verwendbar sind (siehe **Eigene Objektbibliothek anlegen und einsetzen**).
-  **Auswahl:** Damit wählen, bewegen und passen Sie Formularobjekte in der Größe an.
-  **Eingabefolge:** Wechselt in den Modus "Eingabefolge", wo Sie die aktuelle Eingabereihenfolge im Formular ansehen und verändern können (siehe **Eingabereihenfolge der Daten ändern**)
-  **Bewegen:** Wechselt in den Modus "Bewegen", wodurch Sie per Drag&Drop im Fenster rasch zu jedem Teil des Fensters gelangen. Der Cursor wird zum Symbol Hand. Dieser Modus ist besonders hilfreich, wenn Sie im Formular zoomen.
-  **Zoom:** Sie können die Anzeige des Formulars vergrößern oder verkleinern (standardmäßig 100%). Dazu klicken Sie in der Werkzeugleiste am oberen Rand entweder auf das Vergrößerungsglas oder direkt auf eines der danebenliegenden Kästchen. Klicken Sie auf das Vergrößerungsglas, wird der Cursor zum Symbol . Zum Vergrößern um eine Stufe klicken Sie in das Formular. Zum Verkleinern um eine Stufe klicken Sie mit gedrückter **Umschalttaste** in das Formular. Der Cursor wird dann zum Symbol .
-  **Ausrichten:** Damit öffnen Sie ein DropDown-Menü, um Objekte auszurichten. Es wird erst aktiv, wenn mindestens zwei Objekte im Formular ausgewählt sind.
-  **Verteilen:** Damit öffnen Sie ein DropDown-Menü, um Objekte im Formular horizontal oder vertikal anzuordnen. Es wird erst aktiv, wenn mindestens drei Objekte im Formular ausgewählt sind.
-  **Ebene:** Damit öffnen Sie ein DropDown-Menü, um die Ebene der Objekte im Formular zu verändern.
-  **Seite anzeigen:** Hier blättern Sie in den Seiten des Formulars und löschen bzw. fügen Seiten hinzu. Sie können über die Pfeile in den Seiten blättern oder in die Mitte klicken und die entsprechende Seite im DropDown-Menü wählen. Der Benutzerformulareditor erlaubt nicht, Seiten hinzuzufügen.

Objektleiste

Über die Objektleiste des Benutzerformulareditors können Sie verschiedene grafische Objekte im Formular hinzufügen: **Text** (dieses Objekt muss gewählt werden, um den Inhalt von Objekten mit statischem Text zu ändern), **Gruppenrahmen, Viereck, Linie, Abgerundetes Viereck, Oval, Matrix.**




Um ein Objekt im Formular zu erstellen, wählen Sie das entsprechende Symbol in der Objektleiste aus und zeichnen dann das Objekt im Formular.

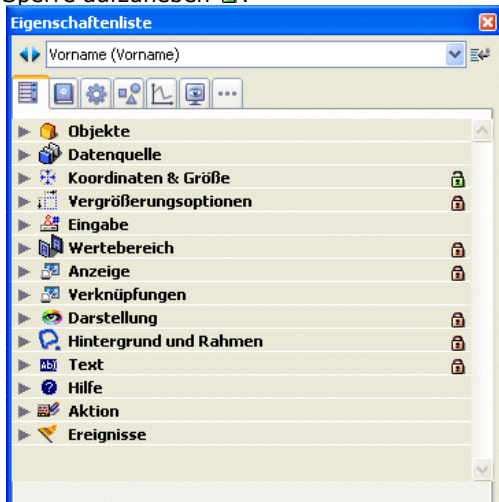
Objekte ansehen und bearbeiten

Der Benutzer kann die Eigenschaften von Objekten im Formular über eine vereinfachte Eigenschaftenliste ansehen und bearbeiten. Die Liste erscheint, wenn Sie auf das Objekt doppelklicken oder im Kontextmenü (rechte Maustaste) den Eintrag **Eigenschaftenliste** wählen.



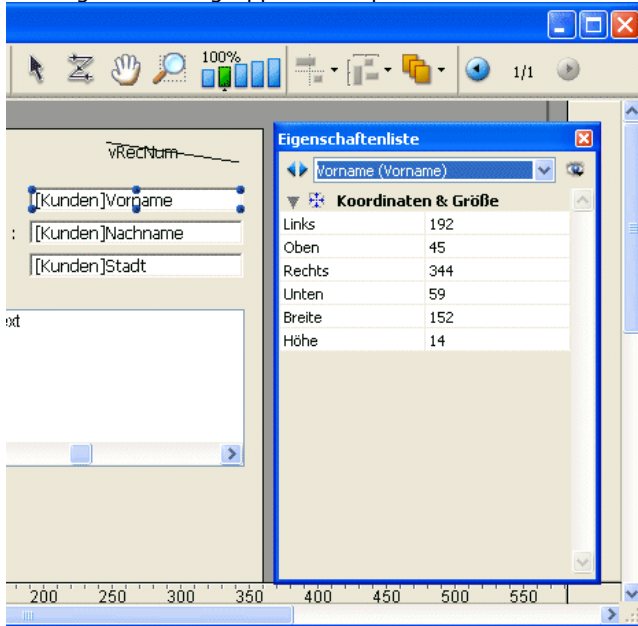
In der Eigenschaftenliste erscheinen nur die Eigenschaften, welche der Benutzer bearbeiten kann.

- Für grafische Objekte, die der Benutzer hinzugefügt hat, sind alle Eigenschaften für diesen Objekttyp verfügbar.
- Für Objekte aus dem Originalformular müssen zwei Bedingungen zutreffen, damit eine Eigenschaft im Formulareditor für Benutzer editierbar ist:
 - Die Eigenschaft muss zu einer Eigenschaftengruppe gehören, die für das Objekt oder Formular editierbar gemacht werden kann. So können z.B. Eigenschaften zum Verwalten von Formularereignissen oder Drag-and-Drop nie vom Benutzer bearbeitet werden.
Im Formulareditor der Designumgebung sind editierbare Eigenschaftengruppen in der Eigenschaftenliste mit dem Symbol eines offenen Vorhängeschlosses gekennzeichnet. Weitere Informationen dazu finden Sie im Abschnitt "Vom Benutzer editierbar" unter **Formulareigenschaften**.
 - Der Entwickler muss für das Objekt oder Formular die Eigenschaftengruppe explizit auf editierbar gesetzt haben. Dazu muss er in der Eigenschaftenliste in der Designumgebung auf das Vorhängeschloss klicken, um die Sperre aufzuheben .



Eigenschaftenliste in der Designumgebung

Eine Eigenschaftengruppe ohne Sperre ist im Formulareditor für Benutzer editierbar:



Das gilt auch für andere Aktionen im Benutzerformular. Ist z.B. die Gruppe "Koordinaten & Größe" ohne Sperre, kann der Benutzer das Objekt mit der Maus oder über Tastatur bewegen.

Menü Bearbeiten

Wird in der Designumgebung der Benutzerformulareditor geöffnet, erscheint nur das Menü **Bearbeiten**. Bei selbst gestalteten Oberflächen in der Anwendungsumgebung erscheint das Menü **Bearbeiten** nur, wenn die darin enthaltenen Befehle über Standardaktionen verwaltet werden.

Alle Befehle zum Bearbeiten (**Ausschneiden**, **Kopieren**, **Einfügen**, **Löschen**, **Alles auswählen**) funktionieren ohne Einschränkung für Objekte, die der Benutzer hinzugefügt hat.

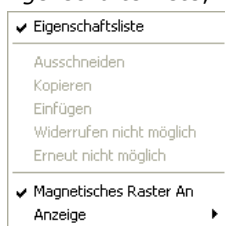
Zu Wahrung der Konsistenz hat der Benutzer dagegen nie die Erlaubnis, ein Objekt aus dem Originalformular zu entfernen. Verwendet er die Befehle **Ausschneiden**, **Löschen** oder die Tasten **Del** bzw. **Rückschritt**, erscheint die Meldung, dass der Benutzer diese Aktion nicht ausführen kann.

Kontextmenü

Das Kontextmenü im Formulareditor für Benutzer bietet einige Einträge. Der Inhalt dieses Menüs lässt sich nicht steuern.

Außerhalb eines Objekts klicken

Klickt der Benutzer außerhalb eines Objekts, erscheinen die standardmäßigen Anzegebefehle des Formulareditors: *Eigenschaftensliste*, *Magnetisches Raster an* und *Anzeige der Editorwerkzeuge*:



- **Magnetisches Raster An:** Das magnetische Raster bietet eine visuelle Hilfe zum Anordnen und Ausrichten von Objekten im Formular. Es arbeitet folgendermaßen: Legen Sie ein Objekt per Drag&Drop in das Formular, zeigt 4D seine mögliche Positionierung in Bezug auf andere Formularobjekte:
 - Stimmen Oberkante oder Mitte von zwei Objekten überein, werden die horizontal möglichen Positionen vorgeschlagen
 - Stimmen die Seitenkanten von zwei Objekten überein, werden die vertikal möglichen Positionen vorgeschlagen4D legt das Objekt an die jeweilige Position und zeigt mittels einer roten Linie die Art der Ausrichtung an:
 - Oben bündig ausrichten
 - Zentriert ausrichten4D gibt bei der Verteilung von Objekten einen für Oberflächen üblichen Abstand an. Wieder kennzeichnen rote Linien die gewählten Abstände.
Hinweis: Sie können das Magnetrastrer manuell ein- oder ausschalten, wenn Sie bei einem ausgewählten Objekt unter Windows die **Strg-Taste**, auf Mac OS die **Befehlstaste** drücken.
- **Anzeige:** Dieses Menü hat dieselben Einträge wie die Schaltfläche **Anzeige** in der Werkzeugleiste des Editors (siehe **Benutzerformulareditor**).

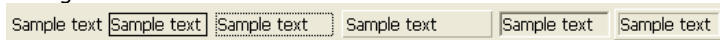
Auf ein Objekt klicken

Klickt der Benutzer auf ein Objekt, erscheinen grafische Befehle, wie Farbe, Ausrichtung, etc..



Der Zugriff auf diese Befehle lässt sich auf die Objekte im Originalformular beschränken. Diese Einstellung legt der Entwickler im Formulareditor fest.

- **Farbe:** Zeigt zwei Paletten zum Ändern der Vordergrund- und Hintergrundfarben eines Objekts an.
- **Ausrichtung:** Bietet Befehle zum Ausrichten und Verteilen von Objekten. Die Befehle zum Ausrichten werden erst aktiv, wenn mindestens zwei „bewegbare“ Objekte ausgewählt sind. Die Befehle zum Verteilen werden erst aktiv, wenn mindestens drei „bewegbare“ Objekte ausgewählt sind.
- **Rahmen Linienstil:** Damit lässt sich ein Randstil für ausgewählte Objekte festlegen. Hier sind Beispiele für die verfügbaren Stilarten:



Transparent / Normal / Gepunktet / Erhaben / Vertieft / Doppelt

- **Automatische Größe:** Damit lässt sich die Größe von Textobjekten und Bildern anpassen, um unter Beibehaltung der grafischen Merkmale (Schrift, Größe, Stil) eine optimale Anzeige im Formular zu erzielen oder um sie in ihrer Originalgröße anzuzeigen (Bilder). Diese Operation lässt sich auch erreichen, wenn Sie unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** in die rechte untere Ecke des Objekts klicken.
- **Ebene:** Ändert die Ebene von Objekten.
- **Ähnliche Objekte auswählen:** Damit lassen sich alle Objekte auswählen, die vom selben Typ wie das gewählte Objekt sind.


Eingabereihenfolge

Die Eingabefolge ist die Reihenfolge, in der Datenfelder, Unterformulare und andere tabfähige Objekte ausgewählt werden, wenn Sie mit der Tabulatortaste durch ein Formular springen.

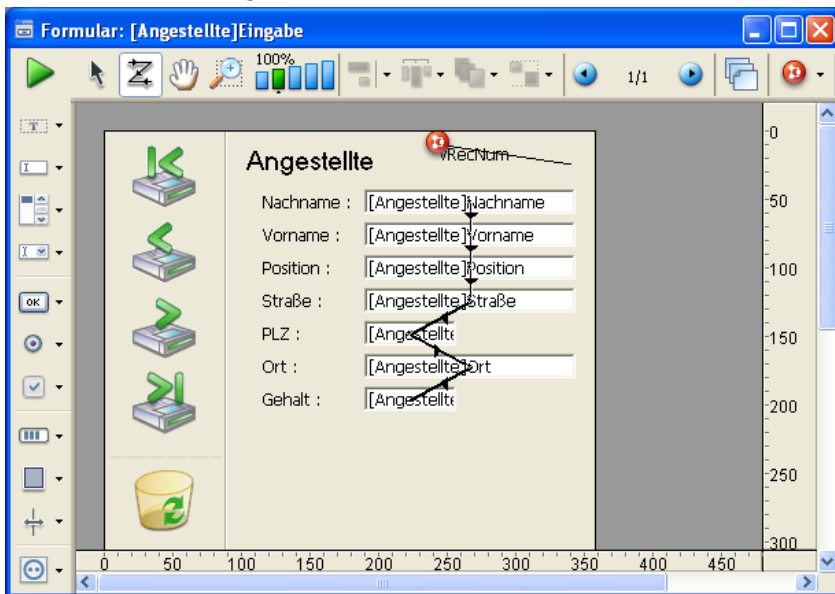
Hinweis: Weitere Informationen über die Standardreihenfolge in Formularen finden Sie im Abschnitt **Eingabereihenfolge der Daten ändern**.

In einigen Fällen ist es erforderlich, eine eigene Eingabefolge festzulegen, z.B. wenn Sie nachträglich Objekte hinzufügen und sie in eine logischere Reihenfolge im Formular bringen wollen.

Eingabefolge anzeigen und ändern

Um die aktuelle Eingabefolge im Formular anzusehen, klicken Sie in der Werkzeugleiste des Benutzerformulareditors auf die Schaltfläche: 

Das Formular sieht folgendermaßen aus:




- Um die Eingabefolge zu ändern, klicken Sie mit dem Mauszeiger auf ein Objekt im Formular und ziehen bei gedrückter Maustaste zu dem Objekt, das Sie als nächstes in die Eingabefolge aufnehmen wollen. 4D ändert die Eingabefolge entsprechend.
- Ist die neue Eingabefolge abgeschlossen, klicken Sie in der Werkzeugleiste erneut auf die Schaltfläche **Eingabefolge**.

Erstes Objekt in der Eingabefolge festlegen

Alle eingebbaren Objekte gehören zu Dateneingabefolge. Um das erste Objekt der Eingabefolge zu definieren, muss der Modus **Eingabefolge** deaktiviert sein.


Um ein Objekt an den Anfang der Eingabefolge zu stellen:

1. Markieren Sie das Objekt, welches das erste in der Eingabefolge sein soll, und setzen es nach hinten.
Sie können im Kontextmenü den Eintrag **Ebene>Nach hinten bringen** wählen oder in der Werkzeugleiste das entsprechende Icon wählen. Das Objekt wird vorübergehend hinter alle anderen Formularelemente gesetzt.
2. Klicken Sie in der Werkzeugleiste auf das Symbol für **Eingabefolge** .
Das ausgewählte Objekt wird erstes Objekt in der Eingabefolge, das bisherige erste Objekt wird zweites. Sie können jetzt bei gedrückter Maustaste von Objekt zu Objekt ziehen, um die gewünschte Eingabefolge herzustellen.
3. Haben Sie die gewünschte Reihenfolge abgeschlossen, klicken Sie in der Werkzeugleiste erneut auf das Symbol für **Eingabefolge**.
4D kehrt zur normalen Operation des Formulareditors zurück. Das zuvor ausgewählte Objekt ist wieder an der ursprünglichen Position. Es ist nicht mehr nach hinten gesetzt.

Eingabegruppe verwenden


Beim Ändern der Eingabefolge können Sie im Formular eine Objektgruppe auswählen, so dass die Standardeingabefolge sich auf die Objekte in der Gruppe auswirkt. So können Sie problemlos die Eingabefolge in Formularen festlegen, in denen Datenfelder gruppen- oder spaltenweise angeordnet sind.

Um eine Gruppe zur Dateneingabe festzulegen:

1. Klicken Sie in der Werkzeugleiste auf das Symbol .
2. Ziehen Sie einen gestrichelten Rahmen um die Objekte, die Sie für die Eingabe gruppieren wollen.
Wenn Sie die Maustaste loslassen, folgen die Objekte, die innerhalb des Rahmens liegen oder von ihm berührt werden, der Standardeingabefolge. Die Eingabefolge für die verbleibenden Objekte wird entsprechend angepasst.

Standardeingabefolge wiederherstellen

Sie können die Standardeingabefolge jederzeit wiederherstellen, d.h. die Reihenfolge, in der die Objekte angelegt wurden.

1. Klicken Sie in der Werkzeugleiste auf das Symbol .
2. Ziehen Sie einen gestrichelten Rahmen um alle Objekte im Formular.
Wenn Sie die Maustaste loslassen, folgen die Objekte, die innerhalb des Rahmens liegen oder von ihm berührt werden, der Standardeingabefolge.


Objektbibliothek verwenden

Jedes Benutzerformular kann zur Bearbeitung eine Objektbibliothek verwenden.

Eine Objektbibliothek ist eine von 4D angelegte externe Datei. Hier können Sie alle Arten von Objekten zur Verwendung in Formularen speichern: Schaltflächen, Texte, Bilder, hierarchische Listen, etc. Bibliotheken werden zusammengelegt, ihr Inhalt lässt sich per Drag&Drop oder Kopieren-Einsetzen verwenden. Sie funktionieren wie eine dauerhafte Zwischenablage.

Weitere Informationen dazu finden Sie im Abschnitt **Eigene Objektbibliothek anlegen und einsetzen**.

Objekte werden mit all ihren Eigenschaften und Standardaktionen gespeichert. Im Benutzerformulareditor lassen sich keine Objekte mit zugewiesenen Objektmethoden verwenden. Die Objektbibliothek zeigt hier nur Objekte ohne Objektmethoden.

Der Entwickler setzt die dem Formular zugewiesene Bibliothek mit dem letzten Parameter des 4D Befehls **EDIT FORM**. Ist dieser Parameter gültig, lässt sich die Bibliothek anzeigen und mit dem Benutzerformular verwenden. Der Benutzer öffnet die Bibliothek über das entsprechende Icon in der Werkzeugleiste des Editors: 

Auf dem Bildschirm erscheint dann das Bibliotheksfenster:











Hinweis: Eine Objektbibliothek kann bis zu 32.000 Einträge enthalten.

Der Benutzer kann Objekte daraus per Drag&Drop oder Kopieren/Einfügen in das Formular setzen. Alle Eigenschaften der übertragenen Objekte, z.B. Koordinaten, Ausmaße, Darstellung, sind automatisch für den Benutzer zugänglich.

Jede Objektbibliothek hat ein Kontextmenü mit den grundlegenden Befehlen zum Bearbeiten. Im Benutzerformulareditor ist nur der Befehl **Kopieren** verfügbar. Der Benutzer kann außerdem in der Bibliothek Objekte weder entfernen, umbenennen noch hinzufügen.

Methoden bearbeiten

-  Überblick über Methoden
-  Methoden verwalten
-  Eigenschaften für Projektmethoden
-  Methodeneditor
-  Methode schreiben
-  Makros erstellen und einsetzen
-  Suchen und Ersetzen in Methoden
-  Methoden ausführen

Überblick über Methoden

Sie können einem 4D Objekt eine Methode zuweisen, um die Aktion des Objektes festzulegen. Eine Methode ist eine Reihe von Anweisungen, die dem Objekt mitteilen, etwas auszuführen. Sie können z.B. Methoden einsetzen, um:

- Bei der Eingabe von Daten Arbeitsregeln festzulegen
- Werte für Datenfelder und Variablen zu berechnen
- Oberflächenelemente zu verwalten, wie z.B. Combo Boxen, hierarchische Listen und Registerkarten
- Drag-and-Drop Aktionen zu verwalten
- Eigenen Menübefehlen Aktionen zuzuordnen
- Multiple Prozesse zu erstellen und zu verwalten
- Transaktionen vorzunehmen,
- Eigene Berichte zu verwalten,
- Zugriffe für Benutzer auf eine Anwendung im Mehrplatzbetrieb zu regeln

Sie können fünf Typen von Methoden erstellen:

- **Objektmethode**n sind mit einzelnen Objekten in einem Formular verbunden
- **Formularmethode**n sind an einzelne Formulare gebunden
- **Trigger** werden ausgelöst, wenn bestimmte Ereignisse auf der Ebene der Datenbank-Engine auftreten
- **Datenbankmethode**n laufen automatisch ab, sobald Ereignisse auftreten, die mit Arbeitssitzungen zusammenhängen
- **Projektmethode**n lassen sich auf unterschiedliche Art und Weise aufrufen

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Methoden**.

Dieses Kapitel erläutert den Einsatz des 4D Methodeneditors, um Methoden zu erstellen und zu ändern. Weitere Informationen über die Programmierbefehle und Syntax finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Grundbegriffe** und in der Beschreibung der einzelnen Befehle.

Methode erstellen und öffnen

Sie erstellen eine Methode mit dem Methodeneditor. Er enthält Werkzeuge, mit denen Sie eine Methode erstellen, testen und bearbeiten können.

Dieser Abschnitt beschreibt, wie Sie die verschiedenen Methoden erstellen und öffnen (Objektmethoden, Projektmethoden, Trigger und Formularmethoden).

Beachten Sie, dass in allen Anwendungen bereits leere Datenbankmethoden angelegt sind, die Sie direkt im Explorer öffnen können.

Hinweis zu 4D Server: Eine Methode wird gesperrt, wenn zwei oder mehr Entwickler versuchen, dieselbe Methode zur gleichen Zeit zu ändern. Öffnet ein Benutzer eine solche Methode in der Designumgebung, ist sie gesperrt. Andere Benutzer können diese Methode erst verändern, wenn der erste Benutzer sie durch Schließen des Fensters wieder freigibt. Sie können die Methode jedoch im Modus "Nur Lesen" öffnen, um die Methode komplett oder Teile daraus zu kopieren.

Objektmethode erstellen oder öffnen

Sie erstellen Objektmethoden für ein Objekt in einem Formular. Sie beginnen im Formulareditor mit einem Formular auf dem Bildschirm.

Um eine Objektmethode im Formulareditor zu erstellen bzw. zu öffnen:

1. Halten Sie unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** gedrückt und klicken auf das Objekt
ODER
Wählen Sie das Objekt und dann im Menü **Objekt** den Eintrag **Objektmethode**
ODER
Klicken Sie mit der rechten Maustaste in das Objekt und wählen im Kontextmenü den Eintrag **Objektmethode**
ODER
Klicken Sie in der Eigenschaftenliste für die Objektmethode auf die Schaltfläche **Bearbeiten**.

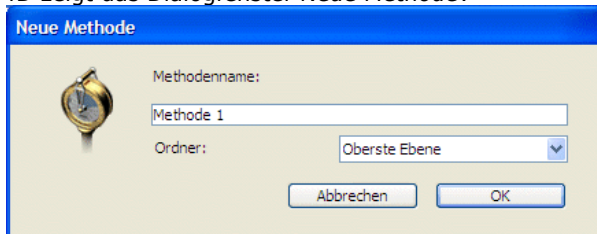
Auf dem Bildschirm erscheint der **Methodeneditor**. Haben Sie die Methode gerade angelegt, ist er leer.

Projektmethode erstellen oder öffnen

Sie erstellen bzw. öffnen eine Projektmethode auf verschiedene Weise:


Um eine Projektmethode über das Menü **Datei/Ablage** zu erstellen:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Neu>Methode**
ODER
Wählen Sie in der Werkzeugleiste die Schaltfläche **Neu>Methode**
4D zeigt das Dialogfenster *Neue Methode*:



2. Geben Sie einen Methodennamen ein.
Methodennamen können bis zu 31 Zeichen enthalten. Das können Buchstaben, Zahlen, Leerzeichen und Unterstriche sein (siehe **Namenskonventionen** im Handbuch *4D Programmiersprache*).
3. (Optional) Wählen Sie einen Ordner zum Speichern der Methode.
Wählen Sie einen Ordner, wird die Methode hier abgelegt. Über Ordner können Sie die Objekte Ihrer Anwendung organisieren. Sie werden im Explorer auf der **Seite Home** verwaltet. Die Methode wird standardmäßig auf der obersten Ebene abgelegt, d.h. außerhalb der Ordner.
4. Klicken Sie auf die Schaltfläche **OK**.
4D öffnet ein leeres Fenster im Methodeneditor. Sie können nun die neue Methode schreiben.

Um eine Projektmethode über den Explorer zu erstellen:

1. Zeigen Sie im Explorer die **Current form table** an.
2. Wählen Sie den Eintrag **Projektmethoden**.
3. Klicken Sie am unteren Rand der Liste auf das Symbol .
- Das Dialogfenster *Neue Methode* erscheint, wo Sie den Namen eintragen und einen Ordner zuweisen können.
4. Geben Sie den Methodennamen ein und klicken Sie auf die Schaltfläche **OK**.

Um eine Projektmethode im Menü **Datei/Ablage** zu öffnen:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Neu>Methode**.
Die **Seite Home** des Explorers erscheint (siehe unten).

Um eine Projektmethode im **Explorer** zu öffnen:

1. Zeigen Sie im Explorer die **Seite Home** an.

2. Erweitern Sie den Eintrag **Projektmethoden** und doppelklicken Sie auf den gewünschten Methodennamen.
ODER
Wählen Sie im Kontextmenü den Eintrag **Methode ändern**. Es erscheint, wenn Sie mit der rechten Maustaste auf den Methodennamen klicken.
Die Methode öffnet sich in einem Fenster des Editors.

Um eine Projektmethode im **Menüeditor** zu öffnen:

1. Klicken Sie im **Menüeditor** für einen Befehl der aktuellen Menüleiste in den Bereich „Methodenname“.
Er zeigt - sofern vorhanden - den Namen der Methode, die diesem Befehl zugewiesen ist.
2. Klicken Sie rechts neben dem Eingabebereich für Methodenname auf das Icon Bearbeiten
Die Methode wird in einem neuen Methodeneditor-Fenster angezeigt.

Um eine Projektmethode im **Methodeneditor** zu öffnen:

1. Wählen Sie im Methodeneditor den Namen der gewünschten Projektmethode.
2. Wählen Sie im Kontextmenü des Editors den Eintrag **Methode ändern** oder verwenden Sie unter Windows die Tastenkombination **Strg+k**, auf Mac OS **Befehlstaste+k**
ODER:
Doppelklicken Sie unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahltaste** auf den Namen der Projektmethode (Er darf kein Leerzeichen enthalten!).
Die Projektmethode erscheint in einem neuen Fenster des Methodeneditors. Kann 4D die Projektmethode nicht identifizieren, öffnet sich im Explorer die **Seite Home**.

Trigger erstellen oder öffnen

Sie können einen Trigger mit einem Tastenkürzel im **Struktureditor** oder direkt im **Explorer** erstellen.

Um einen Trigger im Fenster des **Struktureditor** zu erstellen:

1. Doppelklicken Sie unter Windows mit gedrückter **Alt-Taste**, auf Mac OS mit gedrückter **Wahltaste** auf den Titel der Tabelle im Fenster des Struktureditors
ODER
Klicken Sie im Inspektorfenster für **Tabelleneigenschaften** auf die Schaltfläche **Bearbeiten**.
Auf dem Bildschirm erscheint der **Methodeneditor**. Er ist leer, wenn Sie den Trigger gerade erstellt haben.


Um einen Trigger im **Explorer** zu erstellen bzw. zu öffnen:

1. Zeigen Sie die **Current form table** des Explorer an und erweitern den Eintrag "Trigger".
2. Markieren Sie die gewünschte Tabelle und klicken am unteren Rand auf das Symbol .
ODER
Doppelklicken Sie auf die gewünschte Tabelle.
Der Trigger erscheint im Fenster des **Methodeneditor**.

Formularmethode erstellen oder öffnen

Sie können eine Formularmethode im Explorer auf der **Current form table** oder im **Formulareditor** öffnen.

Um eine Formularmethode im Explorer zu erstellen oder zu öffnen:

1. Zeigen Sie im Explorer die **Current form table**.
2. Erweitern Sie den Eintrag Projektformularmethode oder Tabellenformularmethode, je nachdem für welchen Formulartyp Sie eine Methode öffnen wollen.
3. (nur Tabellenformulare): Erweitern Sie die Tabelle, zu der das Formular gehört.
4. Wählen Sie das Formular, dessen Methode Sie öffnen möchten.
5. Klicken Sie am unteren Rand auf das Symbol .
ODER
Doppelklicken Sie auf den Formularnamen.
Die Formularmethode erscheint im Fenster des **Methodeneditor**.

Hinweis: Sie können eine Formularmethode auch im Explorer über die **Seite Formulare** erstellen bzw. öffnen. Klicken Sie dazu mit der rechten Maustaste auf den Formularnamen und wählen im Kontextmenü den Eintrag **Formularmethode bearbeiten**.

Um eine Formularmethode im Formulareditor zu erstellen:

1. Wählen Sie im Menü **Formular** den Befehl **Formularmethode**
ODER
Klicken Sie mit der rechten Maustaste in einen leeren Bereich des Formulareditors und wählen im Kontextmenü den Eintrag **Formularmethode**.
ODER
Klicken Sie in der Eigenschaftenliste neben der Zeile Formularmethode auf die Schaltfläche **Bearbeiten**.

Die Formularmethode erscheint im Fenster des **Methodeneditor**.


Methoden löschen

Sie können jederzeit einen Trigger, eine Projekt- oder Formularmethode über den **Explorer** löschen. Eine Objektmethode löschen Sie über den Formulareditor.

Dagegen können Sie Datenbankmethoden nicht löschen. Um solch eine Methode abzuschalten, löschen Sie alle Anweisungen der Methode oder setzen vor jede Zeile das Symbol `"/"`. Dieses Zeichen wird für Kommentare verwendet. Weitere Informationen dazu finden Sie im Abschnitt .

Projekt- oder Formularmethode oder Trigger über den Explorer löschen

Um einen Trigger, eine Projekt- oder Formularmethode über den Explorer zu löschen:

1. Zeigen Sie im Explorer die **Current form table** an.
Überprüfen Sie, ob Sie wirklich diese Seite ausgewählt haben, denn bestimmte Löschoptionen können Sie nicht rückgängig machen.
2. Erweitern Sie den jeweiligen Methodentyp, wo Sie die Methode löschen wollen (Projektformularmethode, Tabellenformularmethode, Trigger oder Projektmethode).
3. Wählen Sie den Namen der Tabelle bzw. des Formulars, der die zu löschende Methode zugewiesen ist.
ODER
Markieren Sie den Namen der Projektmethode, die Sie löschen wollen.
4. Klicken Sie am unteren Rand des Fensters auf das Symbol .

Hinweis: Um eine Projektmethode zu löschen, können Sie auch mit der rechten Maustaste auf den Namen der Projektmethode klicken und im Kontextmenü den Eintrag **Methode löschen** wählen.

- Löschen Sie eine Formularmethode oder einen Trigger, zeigt 4D eine Warnung, die Sie bestätigen müssen. Das Löschen ist definitiv.
- Löschen Sie eine Projektmethode, führt 4D die Operation direkt aus. Das Löschen ist jedoch erst dann definitiv, wenn der Papierkorb geleert wurde. Weitere Informationen dazu finden Sie auf der [Seite Papierkorb](#).

Objektmethode löschen

Sie können Objektmethode im Formulareditor löschen. In manchen Fällen läuft die Anwendung schneller, wenn Sie unnötige Objektmethode löschen.

Um eine Objektmethode zu löschen:

1. Zeigen Sie das Formular mit den zu löschenden Methoden an.
2. Wählen Sie die Objekte, denen Objektmethode zugeordnet sind, die Sie löschen wollen.
3. Wählen Sie im Menü **Objekt** den Befehl **Objektmethode löschen**.
4D entfernt die Objektmethode des/der gewählten Objekte.

Hinweis: Haben Sie eine Objektmethode aus Versehen gelöscht, wählen Sie im Menü **Bearbeiten** den Befehl **Widerrufen**.

Methoden importieren und exportieren

Sie können in 4D Datei-, Datenbank-, Projekt- und Objektmethode sowie Trigger importieren und exportieren. Diese Befehle sind im **Methodeneditor** verfügbar.

- Wählen Sie den Befehl **Exportiere Methode**, erscheint ein Standard Sichern-Dialog, wo Sie Name, Position und Format der Exportdatei wählen (siehe unten). Wie beim Drucken wird der gesamte Code exportiert, unabhängig davon, ob Teile ausgeblendet sind.
- Wählen Sie den Befehl **Importiere Methode**, erscheint ein Standard Öffnen-Dialog, wo Sie die Datei für den Import angeben.
Der Import ersetzt den gewählten Text in der Methode. Wollen Sie eine vorhandene Methode ersetzen, wählen Sie vor dem Import die gesamte Methode aus.

Die Import/Export Funktion ist plattformunabhängig: Sie können eine auf Mac OS exportierte Methode in Windows importieren und umgekehrt. 4D verwaltet – soweit erforderlich – die Konvertierung der Zeichen.

Dateiformate für Methoden

4D exportiert/importiert Methoden in zwei Formaten:

- **4D Methode** (Endung ".c4d"): Dieses Format exportiert Methoden in codierter Form. Die Objektnamen sind mit Tokens versehen.
Dieses Format dient insbesondere dazu, um Methoden zwischen 4D Programmen und Plug-Ins in verschiedenen Sprachen auszutauschen. Die Anzeige in einem Texteditor ist nicht möglich.
- **Text** (Endung ".txt"): Dieses Format exportiert Methoden im Format Nur-Text.
In diesem Fall sind die Methoden über einen Standard-Texteditor oder ein Source Control Tool lesbar.
Hinweis: Ab 4D v15 nutzen alle 4D Versionen standardmäßig dieselbe Sprache und dieselben Eingabeeinstellungen für Werte vom Typ Zahl und Datum. Das vereinfacht den Austausch von Code zwischen Entwicklern per Copy/Paste oder beim Exportieren von Methoden. Um die lokalen Einstellungen (wie in vorherigen 4D Versionen) zu verwenden, können Sie unter **4D Einstellungen -> Methoden** die Option **Verwende regionale Systemeinstellungen** aktivieren (siehe [Seite Methoden](#)).

Eigenschaften für Projektmethoden

Nachdem Sie eine Projektmethode erstellt haben, können Sie diese umbenennen und ihre Eigenschaften ändern. Das sind hauptsächlich Zugriffsrechte, Sicherheitsbedingungen (Zugriff durch Benutzer, integrierte Server oder Services) und ihr Ausführungsmodus.

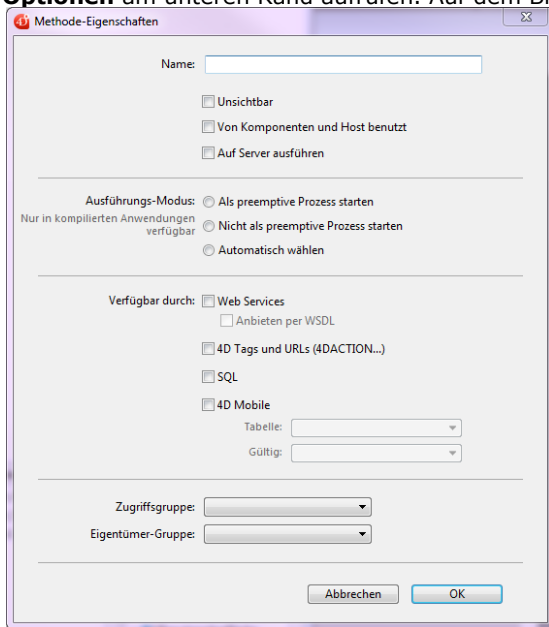
Die anderen Methodenarten haben keine spezifischen Eigenschaften. Ihre Eigenschaften sind mit denen der dazugehörigen Objekte verbunden.

Um die Eigenschaften einer Projektmethode zu ändern:

1. Wählen Sie im **Methodeneditor** im Menü **Methode** den Befehl **Methoden-Eigenschaften**

ODER

Klicken Sie im 4D Explorer auf der **Seite Methoden** mit der rechten Maustaste auf eine Projektmethode und wählen dann im Kontextmenü den Eintrag **Methodenauswahl-Eigenschaften**. Sie können diesen Eintrag auch im Menü **Optionen** am unteren Rand aufrufen. Auf dem Bildschirm erscheint folgendes Dialogfenster:



Hinweis: Über die Funktion Methodenauswahl-Eigenschaften können Sie eine Eigenschaft auch für alle oder Teile von Projektmethoden der Datenbank in einer Operation ändern (siehe **Methodeneigenschaften als Gruppe bearbeiten**).

Methodenname

Sie können den Namen einer Projektmethode im Bereich "Name" des Fensters Methode-Eigenschaften oder im **Explorer** ändern.

Der neue Name muss die 4D Namensregeln beachten (siehe Handbuch *4D Programmiersprache* unter **Namenskonventionen**). Wenn bereits eine Methode mit gleichem Namen besteht, zeigt 4D eine Meldung, dass dieser Methodenname bereits verwendet wurde. 4D sortiert bei Bedarf die Liste der Methoden neu.

Warnung: Ändern Sie den Namen einer Methode, die in der Datenbank bereits verwendet wird, kann das sämtliche Methoden oder Formeln ungültig machen, die den alten Methodennamen verwenden, was zu einer Störung in der Funktionsweise der Anwendung führen kann. Sie können die Methode manuell umbenennen, wir raten aber dringend, die Funktion zum Umbenennen für Projektmethoden zu verwenden. Weitere Informationen dazu finden Sie im Abschnitt **Umbenennen**. Auf diese Weise können Sie den Namen überall da, wo die Methode in der Designumgebung aufgerufen wird, automatisch aktualisieren.

Hinweis zu 4D Server: Der Name der Methode wird auf dem Server geändert, sobald Sie die Bearbeitung des Namens abgeschlossen haben. Wenn mehr als ein Benutzer gleichzeitig den Methodennamen ändern, wird der vom letzten Benutzer bearbeitete Name der neue Name.

Sie können Benutzer für Methoden festlegen, damit nur diese Benutzer den Namen der Methode ändern können.

Hinweis: Datenbank-Methoden können Sie nicht umbenennen. Dasselbe gilt für Trigger, Formularmethoden und Objektmethoden, denn sie sind an Objekte gebunden und erhalten ihre Namen vom Objekt.

Eigenschaften

Über Eigenschaften können Sie steuern, wie Projektmethoden verwendet bzw. in verschiedenen Kontexten aufgerufen werden. Über den Explorer können Sie Eigenschaften auch für die gesamte Auswahl von Projektmethoden setzen (siehe nächster Abschnitt).

Unsichtbar

Über den Menübefehl **Start > Methode** können Benutzer Projektmethoden auslösen. Um das zu verhindern, markieren Sie im Dialogfenster Methode-Eigenschaften die Eigenschaft **Unsichtbar**. Eine ausgeblendete Methode erscheint nicht mehr im Dialogfenster Methode Ausführen (siehe Abschnitt **Über das Fenster Methode ausführen**).

Wenn Sie eine Projektmethode ausblenden, steht sie für Programmierer der Datenbank weiterhin zur Verfügung. Sie verbleibt in der Liste auf der **Seite Formulare** des Explorers und in der Routinenliste des Methodeneditors.

Gemeinsam von Komponenten und Host benutzt

Diese Option wird im Rahmen von Komponenten verwendet. Ist sie markiert, gibt sie an, dass die Methode für Komponenten verfügbar ist, wenn die Anwendung als Host Datenbank verwendet wird und umgekehrt für Host Datenbanken, wenn die Anwendung als Komponente verwendet wird.

Weitere Informationen zu Komponenten finden Sie im Kapitel **4D Komponenten entwickeln und installieren**.

Auf Server ausführen

Diese Eigenschaft wird nur für eine 4D Anwendung im Client/Server Modus berücksichtigt. Ist die Option markiert, wird die Projektmethode immer auf dem Server ausgeführt, unabhängig wie sie aufgerufen wird.

Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Attribut Auf Server ausführen**.

Ausführungsmodus

In diesem Teil können Sie die Methode als im preemptive Modus ausführbar deklarieren. Standardmäßig führt 4D alle Projektmethoden Ihrer Anwendungen im kooperativen Modus aus. Um die Vorteile des preemptive Modus zu nutzen, müssen Sie zuerst alle Methoden, die nach Möglichkeit im preemptive Modus starten sollen, explizit deklarieren. Der Compiler prüft dann, ob solche Methoden tatsächlich thread-safe sind.

Hinweis: Die Ausführung im preemptive Modus ist nur im kompilierten Modus und nur in 64-bit Versionen von 4D möglich. Weitere Informationen finden Sie im Abschnitt **Preemptive 4D Prozesse**.

Es gibt folgende Optionen:


- **Als preemptive Prozess starten:** Mit dieser Option deklarieren Sie, dass die Methode in einem preemptive Prozess laufen kann, d.h. sie soll wenn möglich, im preemptive Modus laufen. Der 4D Compiler überprüft, ob die Methode aktuell preemptiv nutzbar ist, andernfalls gibt er Fehler zurück -- das ist der Fall, wenn Befehle oder Methoden direkt oder indirekt aufgerufen werden, die nicht im preemptive Modus laufen können (die gesamte Aufrufkette wird durchlaufen, aber Fehler werden auf der ersten Unterebene protokolliert). Sie können dann die Methode bearbeiten, damit sie thread-safe wird, oder eine andere Option wählen. Ist die preemptive Nutzbarkeit der Methode bestätigt, wird sie intern als "thread-safe" markiert und im preemptive Modus ausgeführt, sobald alle erforderlichen Bedingungen zutreffen. Diese Eigenschaft definiert die Wahlmöglichkeit des preemptive Modus, garantiert aber nicht, dass die Methode tatsächlich in diesem Modus läuft, da dafür ein spezifischer Kontext erforderlich ist (siehe).
- **Nicht als preemptive Prozess starten:** Mit dieser Option deklarieren Sie, dass die Methode nie im preemptive Modus laufen soll, d.h. sie muss immer im kooperativen Modus laufen, wie in bisherigen 4D Versionen. Der 4D Compiler überprüft nicht, ob die Methode im preemptive Modus laufen kann; sie wird automatisch intern als "thread-unsafe" eingestuft (selbst wenn sie theoretisch preemptiv nutzbar wäre). Beim Aufrufen in Echtzeit kontaminiert diese Methode alle anderen Methoden im gleichen Thread und erzwingt so die Ausführung im kooperativen Modus (selbst wenn andere Methoden "thread-safe" sind).
- **Automatisch wählen:** Mit dieser Option deklarieren Sie, dass Sie die preemptive Nutzbarkeit für die Methode nicht verwalten wollen. Der 4D Compiler bewertet die preemptive Nutzbarkeit der Methode und stuft sie intern als "thread-safe" oder "thread-unsafe" ein. Es wird kein Fehler zur preemptive Ausführung gemeldet. Wird die Methode als "thread-safe" gewertet und in einem preemptive Kontext aufgerufen, verhindert das in Echtzeit nicht die preemptive Thread-Ausführung. Wird sie umgekehrt als "thread-unsafe" gewertet, verhindert das bei Aufrufen in Echtzeit jede preemptive Thread-Ausführung. Beachten Sie, dass mit dieser Option eine Methode, unabhängig von ihrer internen Bewertung der Thread-Sicherheit, immer im kooperativen Modus ausgeführt wird, wenn sie von 4D direkt als erste übergeordnete Methode aufgerufen wird (z.B. über die Funktion **New process**). Ist sie intern als thread-safe eingestuft, wird sie nur berücksichtigt, wenn sie von anderen Methoden innerhalb einer Aufrufkette aufgerufen wird.

Verfügbar durch

Hier können Sie angeben, welche externen Dienste explizit die Methode aufrufen dürfen.

Web Services


Mit dieser Option können Sie die aktuelle Methode über eine SOAP Anfrage als Web Service veröffentlichen. Weitere Informationen dazu finden Sie im Kapitel **Web Services publizieren und einsetzen**. Ist diese Option markiert, ist auch die Option **Anbieten per WSDL** aktiv.

Im Explorer sind Projektmethoden, die als Web Service angeboten werden, mit dem Icon  gekennzeichnet.

Hinweis: Sie können eine Methode nicht als Web Service veröffentlichen, wenn ihr Name Zeichen enthält, die nicht mit der XML Nomenklatur kompatibel sind (z.B. Leerzeichen). In diesem Fall weist 4D die Eigenschaft nicht zu.


Anbieten per WSDL

Dieses Attribut ist nur verfügbar, wenn die Option "Web Services" markiert ist. Damit können Sie die aktuelle Methode in die WSDL-Datei von 4D einfügen. Weitere Informationen dazu finden Sie im Abschnitt **WSDL erzeugen**.

Im Explorer sind Projektmethoden, die als Web Service und im WSDL angeboten werden, mit dem Icon  gekennzeichnet.

4D HTML Tags und URLs (4DACTION...)

Diese Option verstärkt die Sicherheit des 4D Web Server: Ist sie inaktiv, kann die Projektmethode nicht über eine HTTP Anfrage ausgeführt werden, wenn sie das spezielle URL 4DACTION, (sowie das frühere URL 4DMETHOD) zum Aufrufen von 4D Methoden enthält, bzw. die speziellen Tags 4DSCRIPT, 4DTEXT und 4DHTML. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter **URLs und Form Actions** und **4D HTML Tags**.

Im Explorer sind Projektmethoden mit diesem Attribut mit dem Icon  gekennzeichnet.

Aus Sicherheitsgründen ist diese Option standardmäßig inaktiv. Sie müssen Methoden, die über spezielle Web URLs oder Tags ausführbar sind, einzeln angeben.

SQL

Mit dieser Option lässt sich die Projektmethode mit der SQL Engine von 4D ausführen. Sie ist standardmäßig nicht gewählt, d.h. 4D Projektmethoden sind geschützt und können nicht von der SQL Engine von 4D aufgerufen werden - außer sie wurden explizit autorisiert.

Diese Eigenschaft gilt für alle internen und externen SQL Anfragen - ausgeführt über den ODBC Treiber, zwischen den 4D Tags **Begin SQL/End SQL** oder den 4D Befehl **QUERY BY SQL**.

Hinweise:

- Auch wenn eine Methode das Attribut „SQL“ hat, werden zum Ausführen der Methode die Zugriffsrechte, die in den Datenbank-Eigenschaften und Methodeigenschaften gesetzt sind, berücksichtigt.
- Die ODBC-Funktion **SQLProcedure** gibt nur Projektmethoden mit dem Attribut „SQL“ zurück.

Weitere Informationen dazu finden Sie im Handbuch *4D SQL Reference* unter **4D SQL engine implementation**.

4D Mobile

Diese Option definiert, dass die Methode als Teil eines 4D Mobile Links mit einem Wakanda Server verfügbar und gültig ist.

Weitere Informationen dazu finden Sie im Abschnitt **Die 4D Datenbank konfigurieren** des Handbuchs *4D Mobile*.

Zugriff und Eigentumsrechte

Sie können den Zugang zu Methoden regeln, indem Sie Zugriffs- und Eigentumsrechte für Benutzergruppen festlegen. Weitere Informationen dazu finden Sie im Kapitel **Benutzer und Gruppen**.

- Die Dropdown-Liste für Zugriff regelt, welche Gruppe die Methode ausführen kann. Versucht ein Benutzer, der nicht zu dieser Gruppe gehört, die Methode auszuführen, zeigt 4D eine Meldung, dass das Kennwort des Benutzers die Ausführung der Methode nicht zulässt.
- Die Dropdown-Liste der Benutzer regelt, welche Gruppe die Methode im Designmodus bearbeiten kann. Versucht ein Benutzer, der nicht zu dieser Gruppe gehört, die Methode im Designmodus zu bearbeiten, zeigt 4D eine Meldung, dass der Benutzer nicht berechtigt ist, diese Methode zu bearbeiten.

Benutzer, die zu beiden Gruppen gehören, können die Methode ohne Einschränkung verwenden und ändern.

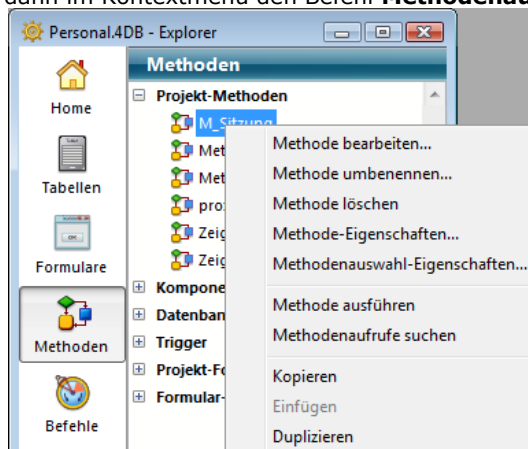
Methodeneigenschaften als Gruppe bearbeiten

Über das Dialogfenster "Eigenschaften für Methoden" können Sie Eigenschaften, wie Unsichtbar, Verfügbar durch Web Service, etc., für alle oder Teile der Projektmethoden in der Datenbank in einer einzigen Operation zuweisen. Das ist besonders hilfreich, um Eigenschaften in einer großen Anzahl Projektmethoden zu ändern. Die Funktionalität eignet sich auch, um in der Entwicklungsphase rasch gängige Eigenschaften auf Gruppen mit ähnlichen Methoden anzuwenden.

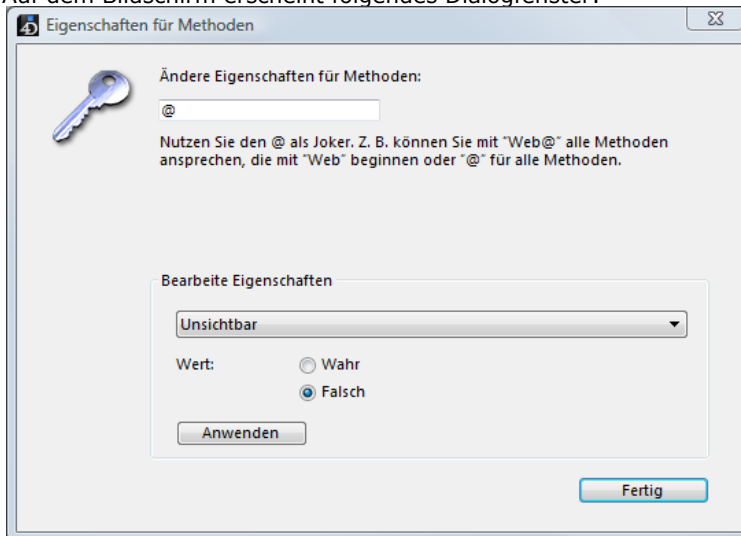
Hinweis: Um diese Aktionen auszuführen, müssen Sie als Designer angemeldet sein.

Um Methoden-Eigenschaften zu setzen:

1. Klicken Sie im Explorer auf der **Seite Methoden** mit der rechten Maustaste auf eine Projektmethode und wählen dann im Kontextmenü den Befehl **Methodenauswahl-Eigenschaften**.



Auf dem Bildschirm erscheint folgendes Dialogfenster:



2. Geben Sie im Bereich "Ändere Eigenschaften für Methoden" eine Zeichenfolge ein, um die Methoden zu bestimmen, die Sie als Gruppe ändern wollen. Diese Zeichenfolge dient als Suchkriterium für Methodennamen. Zur Unterstützung können Sie den Joker @ verwenden:
 - Setzen Sie für *Methodennamen, die beginnen mit...* den Joker @ an das Ende der Zeichenfolge, zum Beispiel web@
 - Setzen Sie für *Methodennamen, die enthalten...*, den Joker @ innerhalb der Zeichenfolge, z.B. web@write
 - Setzen Sie für Methodennamen, die enden mit..., den Joker @ an den Anfang der Zeichenkette, z.B. @write
 - Setzen Sie für alle Methoden allein den Joker @ in den Bereich.

Hinweise:

 - Die Suche achtet nicht auf Klein- und Großschreibung.
 - Sie können auch mehrere Joker @ in der Zeichenfolge eingeben, z.B. dtro_@web@pro.@"
3. Wählen Sie im Bereich "Bearbeite Eigenschaften" eine Eigenschaft aus der DropDown-Liste und markieren dann das Optionsfeld **Wahr** bzw. **Falsch**.

Hinweis: Hat die Eigenschaft "Anbieten als WSDL" den Wert Wahr, wird sie nur auf die Projektmethoden angewandt, die bereits die Eigenschaft "Verfügbar durch Web Service" enthalten.
4. Klicken Sie auf die Schaltfläche **Anwenden**.

Die Änderung wird sofort für alle Projektmethoden ausgeführt, die mit der definierten Zeichenfolge beginnen.

Methodeneditor

Der Methodeneditor besteht standardmäßig aus mehreren Bereichen: Werkzeugleiste, Eingabebereich, Listenbereich, Statusleiste und Spalte für Unterbrechungspunkte.

Den Eingabebereich können Sie verändern, z.B. Listen hinzufügen oder Zeilennummern ausblenden.

Werkzeugleiste

Jedes Fenster des Methodeneditors enthält eine Werkzeugleiste, über die Sie direkt auf die Grundfunktionen zum Ausführen und Bearbeiten von Methoden zugreifen können.

Methode ausführen

Diese Schaltfläche löst die Ausführung der Methode aus. Über das PopUp-Menü können Sie zwischen verschiedenen Arten wählen. Nur Projekt- und Datenbankmethoden können dieses Icon nutzen. Weitere Informationen dazu finden Sie im Abschnitt **Methoden ausführen**.

Suche

Diese Schaltfläche zeigt den Standard Suchen-Dialog an, über den Sie eine Suche in der aktuellen Methode ausführen können. Über die zugeordnete Combo Box können Sie direkt eine Standardsuche starten: Geben Sie eine Zeichenkette ein und drücken Sie die **Enter**-Taste oder die **Zeilenschaltung**. Die Combo-Box zeigt auch die zuletzt ausgeführten Suchläufe; wählen Sie einen davon aus, wenn Sie die Suche erneut ausführen wollen. Weitere Informationen dazu finden Sie im Abschnitt **Suchen und Ersetzen in Methoden**.

Makros

Diese Schaltfläche zeigt ein PopUp-Menü mit den verfügbaren Makro-Befehlen an. Weitere Informationen dazu finden Sie im Abschnitt **Makros erstellen und einsetzen**.

Alle ausblenden/Alle einblenden

Damit klappen Sie alle Programmierstrukturen der Methode auf oder zu. Weitere Informationen dazu finden Sie unter "Ein/Ausblenden" im Abschnitt **Methode schreiben**.

Listen ein/ausblenden

Diese Schaltfläche blendet die Listen im Fenster ein oder aus. Weitere Informationen dazu finden Sie im unteren Abschnitt "Listenbereich".

Methodeninformationen

Diese Schaltfläche zeigt das Dialogfenster der Methodeigenschaften an (Nur Projektmethoden). Weitere Informationen dazu finden Sie im Abschnitt **Eigenschaften für Projektmethoden**.

Letzte Werte in Zwischenablage

Diese Schaltfläche zeigt ein PopUp-Menü mit den letzten 20 Elementen an, die im Fenster kopiert wurden. Wählen Sie ein Element aus, wird es an die Stelle, wo der Cursor liegt, kopiert.

Zwischenablage

Dieser Bereich zeigt die Icons für die 9 Zwischenablagen, die im Methodeneditor verfügbar sind. Ein weiß-oranges Icon mit Nummer zeigt an, dass die Zwischenablage Daten enthält, ein oranges, dass sie leer ist. Um diese Zwischenablagen zu verwenden, können Sie direkt darauf klicken oder Tastenkürzel verwenden. Weitere Informationen dazu finden Sie im Abschnitt **Methode schreiben**.

Sperre (nur 4D Server)

Diese Schaltfläche gibt an, dass die Methode durch einen anderen Benutzer im Client-/Server-Betrieb gesperrt ist. Klicken Sie auf dieses Icon, wird die Methode erneut geladen und zwar mit den vom anderen Benutzer ausgeführten Änderungen.

Eingabebereich

Der Eingabebereich enthält den Text der Methode. In diesen Bereich geben Sie den Text der Methode ein und ändern ihn. Der Editor rückt den Methodentext automatisch ein und setzt für eine klare Programmstruktur die verschiedenen Syntaxelemente in Farbe. Zur Erläuterung geben Sie Kommentare in den Methodentext ein.

Sie können die Anzeige dieses Bereichs an Ihre Wünsche anpassen. Die Anpassung wird automatisch auf alle Fenster des Editors übertragen.

- **Schrift** und **Schriftgröße**: Sie können Schriftart und -größe für den Eingabebereich in den 4D Einstellungen auf der **Seite Methoden** festlegen. Sie können die Schriftgröße auch über den Menübefehl **Methode > Anzeigen** ändern: Die Einträge **Größere Schrift** und **Kleinere Schrift** ermöglichen, die Schriftgröße in Ein-Punkt Schritten zu variieren.
- **Stil und Farbe von Syntaxelementen**: Sie können jedem Elementtyp der 4D bzw. SQL Sprache eine spezifische Farbe bzw. einen spezifischen Stil zuweisen. Klicken Sie dazu mit der rechten Maustaste auf ein Programmiersprache-Element (Variable, Schlüsselwort, etc.) und wählen eine Option im Untermenü **Stil**. Die Änderung gilt für alle Elemente vom gleichen Typ im aktuellen Fenster sowie in anderen Fenstern des Editors. Sie können diese Optionen auch in den 4D Einstellungen auf der **Seite Methoden** definieren.
- **Leerzeichen**: Sie können die Abstände zwischen Wörtern mit Punkten (.) anstelle von Leerzeichen anzeigen. Dazu wählen Sie den Menübefehl **Methode > Anzeigen** den Eintrag **Leerzeichen**. Er ist dann mit einem Häkchen markiert. Diese Funktion gilt für alle Code-Elemente (Befehlsnamen, Variablen, Kommentare, etc.).
`APPEND TO ARRAY($_CurLang; "en")` `APPEND·TO·ARRAY($_CurLang; "en")`
- **Breite der Einrückung von Code**: Diese Option setzen Sie in den 4D Einstellungen auf der **Seite Methoden**.
- **Farben für die Oberfläche**: Sie können die verschiedenen Farben für die Oberfläche des Eingabebereichs im Methodeneditor ändern, wie Hervorhebung, Hintergrund, usw. Die Optionen finden Sie in den 4D Einstellungen auf der **Seite Methoden**.

Weitere Informationen zum Eingeben von Code finden Sie im Abschnitt **Methode schreiben**.

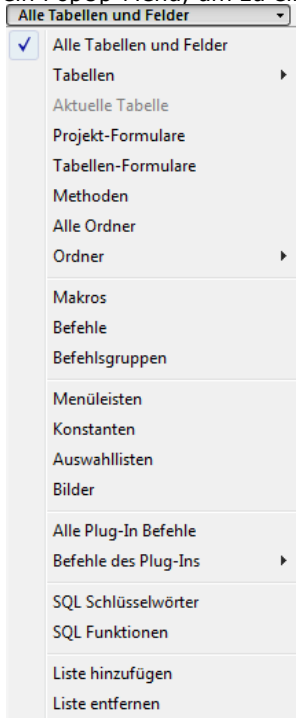
Listenbereich


In diesem Bereich können Sie eine oder mehrere Listen mit den Elementen zum Schreiben von Methoden anzeigen, wie z.B. Befehle, Konstanten, Formulare. Sie können Anzahl und Inhalt der Listen im Fenster selbst bestimmen.

Anzeige der Listen

Der Methodeneditor zeigt standardmäßig vier Listen. Sie können die Breite jeder Liste verändern. Dazu ziehen Sie die vertikale Trennlinie zwischen den Listen an die gewünschte Stelle. Sie können auch die horizontale Trennlinie zwischen Eingabebereich und Listen nach oben oder unten bewegen, um diese Teile kleiner oder größer zu ziehen.

- Doppelklicken Sie auf ein Element in einer Liste, wird es im Eingabebereich an der Stelle des Cursors eingefügt.
- Um den **Inhalt einer Liste zu ändern**, klicken Sie in den Titel der betreffenden Liste. Auf dem Bildschirm erscheint ein PopUp-Menü, um zu einem anderen Element zu wechseln.



- Um eine **Liste hinzuzufügen oder zu entfernen**, klicken Sie auf den Titel der Liste und wählen im PopUp-Menü den Eintrag *Liste hinzufügen* bzw. *Liste entfernen* (siehe unten). Beachten Sie, dass im Editor mindestens eine Liste angezeigt werden muss. Klicken Sie auf die letzte Liste, wird der Befehl **Liste entfernen** deaktiviert. Um alle Listen auf einmal auszublenden, klicken Sie entweder auf das Symbol  oder blenden diese standardmäßig in den 4D Einstellungen aus (siehe unten).
- Sie können die Listen auch in allen Fenstern ausblenden. Deaktivieren Sie dazu im Menü **Methode** den Befehl **Anzeigen>Listen** (das Häkchen zeigt an, dass die Listen angezeigt werden) oder deaktivieren Sie in den 4D Einstellungen auf der **Seite Methoden** unter Optionen den Eintrag **Listen einblenden**. Die Änderungen in den Einstellungen werden erst wirksam, wenn Sie die Methode schließen und erneut wieder öffnen.

Sie können die im Methodeneditor definierten Parameter auch als Vorlage sichern. Mehr Informationen dazu finden Sie im Abschnitt **Als Vorlage übernehmen**.

Listentypen

Im Listenbereich des Methodeneditors können Sie folgende Listen anzeigen:

- **Alle Tabellen und Felder**: Namen der Tabellen und Felder der Datenbank als hierarchische Liste. Doppelklicken Sie auf einen Feldnamen, um ihn in der Methode einzufügen, fügt 4D ihn unter Berücksichtigung der Syntax ein und hängt – sofern vorhanden – den Namen der Tabelle bzw. Untertabelle an.
- **Tabellen** (Untermenü): Feldnamen der über das Untermenü ausgewählten Tabelle
- **Aktuelle Tabelle**: Feldnamen der aktuellen Tabelle (verfügbar in Triggern, Formularmethoden und Objektmethoden)
- **Projekt-Formulare**: Namen der Projektformulare der Datenbank. Doppelklicken Sie auf einen Formularnamen, fügt 4D ihn unter Berücksichtigung der Syntax ein: Der Formularnamen steht zwischen Anführungszeichen.
- **Tabellen-Formulare**: Namen der Tabellenformulare der Datenbank als hierarchische Liste. Doppelklicken Sie auf einen Formularnamen, fügt 4D ihn unter Berücksichtigung der Syntax ein: Der Formularnamen steht zwischen Anführungszeichen, der Tabellenname wird mit Strichpunkt vorangestellt. Beispiel: [Tabelle];"Formular".
- **Methoden**: Namen der Projektmethoden der Datenbank.
- **Alle Ordner**: Namen der Ordner und Unterordner in der Datenbank als hierarchische Liste. Über Ordner können Sie die Objekte Ihrer Anwendung nach eigenen Kriterien klassifizieren. Sie werden im Explorer über die **Seite Home** verwaltet.
- **Ordner** (Untermenü): Inhalt des gewählten Ordners.
- **Makros**: Namen der für die Datenbank definierten Makros (siehe **Makros erstellen und einsetzen**).
- **Befehle**: Die Befehle der 4D Programmiersprache in alphabetischer Reihenfolge.
- **Befehlsgruppen**: Die Befehle der 4D Programmiersprache sortiert nach Thema, als hierarchische Liste.

- **Menüleisten:** Namen und Nummern der Menüleisten, die mit dem 4D Menüleisteditor erstellt wurden (siehe Kapitel [Menüs und Menüleisten](#)).
- **Konstanten:** 4D Konstanten und alle Plug-Ins, sortiert nach Thema als hierarchische Liste.
- **Auswahllisten:** Namen der Listen
- **Bilder:** Namen und Nummern der Bilder, gespeichert in der 4D [Bildbibliothek](#).
- **Alle Plug-In Befehle:** Befehle für alle in der Datenbank installierten Plug-Ins als hierarchische Liste, sortiert nach Thema
- **Befehle des Plug-In (Untermenü):** Befehle für das ausgewählte Plug-In in einem Untermenü. Standardmäßig sind die Befehle folgender Plug-Ins verfügbar:
 - **4D Internet Commands:** Dieses Plug-In fügt 4D zusätzliche Internet Funktionen hinzu, insbesondere zum Verwalten von E-Mails. Siehe [4D Internet Commands](#)
 - **4D Chart:** Das ist der in 4D integrierte Editor zum Erstellen von Diagrammen. Siehe [4D Chart Programmiersprache](#)
 - **OLE_Tools (nur Windows):** Das ist das Plug-in zum Verwalten von OLE Bereichen. Es wird mit der 4D Version unter Windows geliefert. Siehe [OLE Tools](#).
- **SQL Schlüsselwörter:** Satz von Schlüsselwörtern, die der 4D SQL Syntax Parser erkennt. Diese Liste enthält Befehle (z.B. SELECT), Klauseln (z.B. WHERE) sowie Funktionen (ABS).
- **SQL Funktionen:** 4D SQL Funktionen

Hinweis: Mit Ausnahme von Makros sind alle Listen in alphabetischer Reihenfolge.

Spalte für Unterbrechungspunkte

In dieser Spalte links vor dem Eingabebereich können Sie die Zeilennummern anzeigen und Unterbrechungspunkte setzen. Unterbrechungspunkte sind sehr nützliche Hilfen beim Testen einer Methode. Sie stoppen die Ausführung des Code an den markierten Stellen und zeigen den Debugger an.

Um einen Unterbrechungspunkt einzufügen, klicken Sie in der linken Spalte an die Stelle, wo Sie einen Unterbrechungspunkt einfügen wollen oder wählen Sie im Kontextmenü des Editors den Eintrag **Breakpoint umschalten**. Der Unterbrechungspunkt erscheint als roter Punkt.

```

15  |  If (Count parameters>=1)
16  |  |  FORM SET INPUT ($1->"INPUT FORM")
17  |  |  End if
18  |  End if

```

Im [Runtime Explorer](#) können Sie die Position aller Unterbrechungspunkte der Datenbank ansehen.

Um einen Unterbrechungspunkt zu löschen, klicken Sie erneut auf den roten Punkt oder wählen Sie im Kontextmenü des Editors den Eintrag **Breakpoint umschalten**.

Um einen Unterbrechungspunkt zeitweise zu deaktivieren oder seine Eigenschaften zu ändern, klicken Sie unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahltaste** auf den Unterbrechungspunkt oder wählen Sie im Kontextmenü des Editors den Eintrag **Breakpoint bearbeiten**. Auf dem Bildschirm erscheint das Dialogfenster mit seinen Eigenschaften. Eine ausführliche Beschreibung der Unterbrechungspunkte finden Sie im Handbuch *4D Programmiersprache* im Abschnitt [Unterbrechungspunkte](#).

Zeilennummern ein-/ausblenden

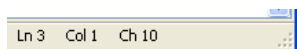
Sie können in allen Fenstern des Methodeneditors die Zeilennummern anzeigen bzw. ausblenden.

- Dazu aktivieren bzw. deaktivieren Sie in den 4D Einstellungen auf der [Seite Methoden](#) die Option **Zeilennummern anzeigen**.
- Sie können die Anzeige auch für jedes Fenster einzeln einstellen. Dazu aktivieren bzw. deaktivieren Sie im Menü **Methode** den Befehl **Zeilennummern**.

Die Anzeige der Zeilennummern erleichtert das Navigieren im Fenster. Über den Befehl **Gehe zu Zeile** im Menü **Methode** können Sie direkt zu einer bestimmten Zeile springen (siehe "Gehe zu Zeile" unter [Suchen und Ersetzen in Methoden](#)).

Statusleiste

Die Statusleiste im Editorfenster rechts unten zeigt kontinuierlich die Position des Cursors an:



- **Ln:** Zeilennummer
- **Col:** Spaltennummer, d.h. die Ebene zwischen der Hierarchie der Programmierstrukturen. Die erste Ebene ist 0. Die Spaltennummer ist zum Debuggen hilfreich, da der Interpreter diese Angabe bei einem Fehler im Code weitergeben kann.
- **Ch:** Position des Zeichens in der Zeile

Als Vorlage übernehmen

Sie können die für den Methodeneditor festgelegten Einstellungen als Vorlage (template) sichern. Sobald Sie die Vorlage sichern, gelten diese Einstellungen für jedes neue Fenster, das der Methodeneditor öffnet.

Die Vorlage speichert folgende Einstellungen:

- Relative Größe des Eingabe- und des Listenbereichs
- Nummern der Listen
- Position und Inhalt der einzelnen Listen
- Relative Breite jeder Liste

Um das Fenster des Methodeneditors zu sichern, wählen Sie in dessen Kontextmenü oder im Menü **Methode** den Befehl **Als Vorlage übernehmen**. Die Vorlage wird unmittelbar gesichert, d.h. es erscheint kein Dialogfenster zum Bestätigen. Sie wird in den Einstellungen der 4D Anwendung gespeichert. Ist bereits eine Vorlage vorhanden, wird sie ersetzt.

Methode schreiben

Der 4D Methodeneditor arbeitet wie ein Texteditor. Sie schreiben eine Methode, indem Sie Text schreiben, Komponenten auswählen und bei gedrückter Maustaste Elemente aus dem Explorer oder anderen Fenstern ziehen. Sie können auch verschiedene „type-ahead“ Funktionen nutzen, um Methoden schneller zu erstellen.

Sie können durch die Methoden scrollen. Eine Methode kann bis zu 32.000 Zeilen Code oder 2 GB Text enthalten.

Der 4D Methodeneditor bietet eine grundlegende Überprüfung auf Syntax-Fehler. Eine zusätzliche Fehlerprüfung erfolgt, wenn die Methode ausgeführt wird.

Hinweis zur Sprache für Befehle und Konstanten

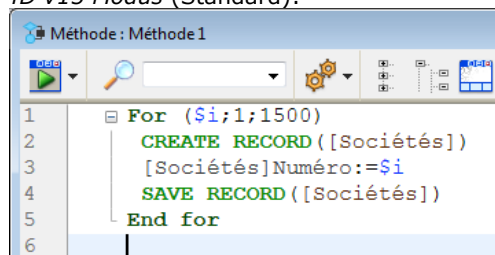
In 4D Versionen vor v15 verwendeten die französische Versionen die Programmiersprache in Französisch, alle anderen Versionen (Englisch, Deutsch, Spanisch, etc.) verwendeten die Programmiersprache in Englisch.

So hieß z.B. der Befehl Nr. 53 in Französisch **STOCKER ENREGISTREMENT** und in allen anderen Sprachen **SAVE RECORD**. Auch bei der Eingabe von Datum oder Werten vom Typ Zahl wurden lokale Einstellungen verwendet.

Das hat sich ab 4D v15 geändert: Der Methodeneditor von 4D verwendet den internationalen Modus, unabhängig von der Sprache der 4D Anwendung oder den lokalen Einstellungen. So lautet der Befehl No. 53 jetzt auch in der französischen 4D Version **SAVE RECORD**. Listen, Fenster mit type-ahead, online Hilfe, etc., zeigen die Namen von Befehlen und Konstanten in Englisch an. Weitere Informationen dazu finden Sie im Abschnitt [Einführung in die 4D Sprache](#).

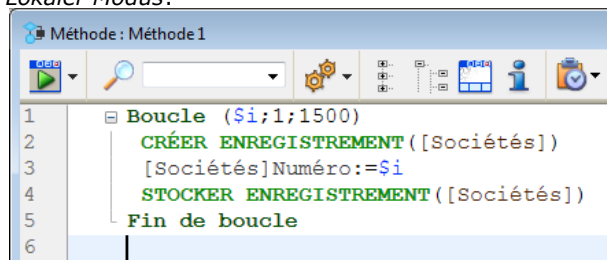
Der Entwickler kann bei Bedarf diese Standardeinstellung im Dialogfenster **4D Einstellungen** auf der Seite **Methoden** deaktivieren. Weitere Informationen dazu finden Sie auf der [Seite Methoden](#). Die nachfolgend beschriebenen Vorgehensweisen zum Schreiben von Code bleiben natürlich dieselben, egal welche Einstellung verwendet wird.

4D v15 Modus (Standard):



```
Méthode : Méthode 1
1  For ($i;1;1500)
2      CREATE RECORD ([Sociétés])
3      [Sociétés]Numéro:=$i
4      SAVE RECORD ([Sociétés])
5  End for
6
```

Lokaler Modus:



```
Méthode : Méthode 1
1  Boucle ($i;1;1500)
2      CRÉER ENREGISTREMENT ([Sociétés])
3      [Sociétés]Numéro:=$i
4      STOCKER ENREGISTREMENT ([Sociétés])
5  Fin de boucle
6
```

Text eingeben und bearbeiten

4D verwendet zum Schreiben und Bearbeiten im Methodeneditor die Standardtechniken der Textverarbeitung. Sie schreiben und die Zeichen erscheinen am Einfügekpunkt. Sie beenden jede Zeile, indem Sie die **Eingabetaste** oder die **Zeilenschaltung** drücken.

Hinweis: Die **Eingabetaste** auf dem Nummernblock hat ein anderes Verhalten als die **Zeilenschaltung** auf der Haupttastatur. Verwenden Sie die **Eingabetaste** auf dem Nummernblock, prüft 4D die Syntax der Codezeile, ohne den Einfügekpunkt in die nächste Zeile zu verschieben.

Der Methodeneditor verwendet für die Syntaxelemente bestimmte Vorgaben für Stil und Farbe. Sie können diese Vorgaben ändern (siehe [Methodeneditor](#)). Wenn Sie tippen und Ihre Eingabe bestätigen, bewertet 4D den Text der Zeile und wendet das passende Anzeigeformat an. Wenn Sie Programmiererelemente wie *If*, *End if* verwenden, rückt 4D auch jede Zeile in Bezug auf die vorige Zeile ein.

Über die Pfeiltasten können Sie rasch von einer Zeile zur anderen wechseln. Das ist schneller, als darauf zu klicken, da der Editor die Zeile wegen der Überprüfung auf mögliche Fehler verzögert.

Unter Windows enthält der Code-Editor einen Eingabe Methodeneditor (IME), der das Schreiben von Code in japanischen oder chinesischen Systemen erleichtert.

Der Methodeneditor enthält unzählige Tastaturkürzel zum Navigieren. Die Liste finden Sie im unteren Abschnitt ["Tastaturkürzel zum Navigieren"](#).

Elemente per Drag-and-Drop einfügen

In 4D können Sie beim Schreiben von Methoden Drag-and-Drop verwenden. Sie können Elemente per Drag-and-Drop aus dem Explorer, innerhalb derselben Methode oder zwischen zwei verschiedenen Methoden einfügen.

- Im **Explorer** können Sie Drag-and-Drop einsetzen für:
 - Namen von Tabellen, Datenfeldern, Formularen und Projektmethoden auf der Seite Home
 - Tabellen- und Datenfeldnamen von der Seite Tabellen
 - Tabellen- und Formularnamen von der Seite Formulare
 - Projektmethoden und Formularnamen von der Seite Methoden
 - Konstanten von der Seite Konstanten
 - 4D Befehle und Funktionen von der Seite Befehle

Setzen Sie ein Element per Drag-and-Drop ein, verwendet 4D immer die korrekte Syntax. Ziehen Sie z.B. den Datenfeldnamen "Vorname" aus der Tabelle [Angestellte], erscheint er im Methodeneditor als "[Angestellte]Vorname". Ziehen Sie den Formularnamen "Eingabe" aus der Tabelle Angestellte, erscheint er im Methodeneditor als "[Angestellte];"Eingabe".

Fügen Sie einen Befehl per Drag-and-Drop von der **Seite Befehle** des Explorer ein, erscheint er mit seiner vollständigen Syntax, d.h. mit allen Parametern im Methodeneditor. Sie müssen diese Syntax natürlich an Ihre Verwendungsweise anpassen. Diese Funktionsweise erinnert Sie lediglich daran, welche Parameter dieser Befehl erwartet, so dass Sie nichts vergessen.

- Drag-and-Drop in einer Methode oder zwischen zwei Methoden:
Im Methodeneditor wird der Drag-and-Drop Mechanismus aktiviert, sobald ein Textbereich gewählt wurde. Drag-and-Drop verschiebt standardmäßig den Text. Um ihn zu kopieren, führen Sie die Operation unter Windows bei gedrückter **Strg-Taste**, auf Mac OS bei gedrückter **Befehlstaste** aus.

Mehrzeilige Eingabe

Sie können eine einzelne Anweisung in mehreren Zeilen schreiben. Dazu müssen Sie lediglich am Zeilenende einer Anweisung einen umgekehrtem Schrägstrich "\" setzen. 4D bewertet dann alle Zeilen auf einmal. Hier ein Beispiel: Die beiden nachfolgenden Anweisungen sind identisch:

```
[DOCS] Plural:=Uppercase([DOCS] Plural[[1]])+Lowercase(Substring([DOCS] Plural;2))
```

```
[DOCS] Plural:=Uppercase([DOCS] Plural[[1]])+\
Lowercase(Substring([DOCS] Plural;2))
```

Schreibweise ändern

Sie können die Schreibweise ausgewählter Zeichen automatisch ändern. Wählen Sie dazu im Menü **Methode** bzw. im Kontextmenü des Editors den Eintrag **Schreibweise**:

- **Groß- / Kleinbuchstaben**: Damit können Sie die ausgewählten Zeichen in Groß- oder Kleinbuchstaben setzen.
- **camelCase / CamelCase**: Damit können Sie ausgewählte Zeichen in Initialen setzen, d.h. jeweils der erste Buchstabe einer Wortgruppe wird großgeschrieben. Diese Schreibweise wird oft für Variablen verwendet, wie z.B. *hireDate* und *PurchaseDate*.
Wenden Sie einen dieser Befehle auf eine Textauswahl an, werden Leerzeichen und Unterstrich entfernt. Die Wörter werden aneinandergeschrieben und der erste Buchstabe jedes Worts wird großgeschrieben.

Ausdruck tauschen

Mit der Funktion **Ausdruck tauschen** können Sie die Argumente eines Ausdrucks, der einen Wert zuweist, umkehren. Zum Beispiel:

```
variable1:=variable2
```

wird zu

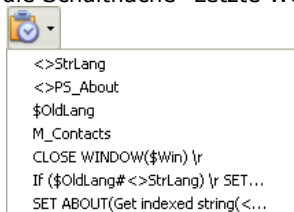
```
variable2:=variable1
```

Diese Funktionalität ist äußerst hilfreich zum Umkehren von Anweisungen, die Eigenschaften erhalten oder setzen bzw. zum Korrigieren von Eingabebefehlern. Zuerst wählen Sie die betreffende(n) Zeile(n), dann im Menü **Methode** oder im Kontextmenü des Bereichs den Befehl **Ausdruck tauschen**. Nur die Zeilen innerhalb der Auswahl, die einen Wert zuweisen, werden geändert.

Mehrfach Kopieren/Einsetzen und Zwischenablagen nummerieren

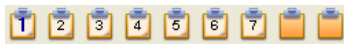
4D bietet zusätzlich zu den Standardbefehlen Kopieren/Einfügen zwei weitere Funktionen zum Arbeiten mit verschiedenen Zwischenablagen:

- Das Programm behält die letzten 10 Aktionen „Kopieren“ bzw. „Ausschneiden“ im Speicher, die in der aktuellen Sitzung im Methodeneditor ausgeführt wurden. Ein so gesicherter Inhalt lässt sich jederzeit wiederverwenden. Dazu rufen Sie im Kontextmenü des Methodeneditors den Eintrag **Kopieren Historie** oder wählen in der Werkzeugleiste die Schaltfläche "Letzte Werte Zwischenablage":



Es werden die ersten Zeichen der kopierten oder ausgeschnittenen Elemente angezeigt. Wählen Sie ein Element aus, wird es an der aktuellen Position des Cursors eingesetzt.

- 9 weitere durchnummerierte Zwischenablagen sind verfügbar und direkt über die Schaltflächen der Werkzeugleiste des Methodeneditors oder über Tastenkürzel einsetzbar:



	Ausgewählten Text in eine Zwischenablage kopieren	Inhalt der Zwischenablage an Cursor-Position setzen
Windows	Shift oder Alt+Klick auf Icon der Zwischenablage Shift oder Alt+Nummer der Zwischenablage	Strg+Klick auf Icon der Zwischenablage Strg+Nummer der Zwischenablage
Mac OS	Shift oder Wahl taste+Klick auf Icon der Zwischenablage Befehl+Shift-taste+Nummer der Zwischenablage	Befehl+Klick auf Icon der Zwischenablage Befehl+Nummer der Zwischenablage

Beachten Sie, dass Sie entweder die Tasten des Nummernblocks oder die erforderlichen Tastenkürzel verwenden müssen, um auf die Nummerntasten der alphanumerischen Tastatur zuzugreifen.

Zeilen bewegen

Sie können die Zeile, wo der Cursor steht, direkt ohne Auswählen bewegen. Wählen Sie dazu im Menü **Methode** den Befehl **Zeilen nach oben verschieben** bzw. **Zeilen nach unten verschieben**. Sie können auch die Kombination **Alt-Taste/Wahl taste + Pfeil oben** oder **Pfeil unten** verwenden.

Änderungsbalken

Farbige Balken erscheinen sofort, wenn Code-Zeilen nach dem Öffnen der Methode verändert wurden:

```
14 ALL RECORDS ([DOCUMENTATIONS])
15 ARRAY TEXT (<>KeyList; 0)
16 ARRAY TEXT (<>PluralKeyList; 0)
```

Die Farbe richtet sich danach, ob die Änderungen gesichert sind oder nicht:

- Gelb: Die Zeile wurde geändert, die Methode wurde noch nicht gesichert
- Grün: Die Zeile wurde geändert und die Methode wurde gesichert

Type-ahead Funktion verwenden

Der Methodeneditor enthält die Funktion "type-ahead". Damit zeigt 4D anhand der ersten eingegebenen Zeichen automatisch eine Auswahl an.

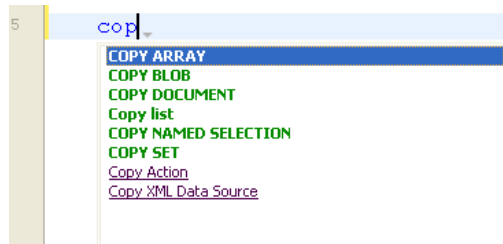
Im nachfolgenden Beispiel löst die Eingabe der Zeichenfolge "cop" die Anzeige eines blauen Dreiecks aus und zeigt somit an, dass es mehrere Auswahlmöglichkeiten gibt:

```
5 cop
```

Hinweis: Entspricht die eingegebene Zeichenfolge einer einzigen Möglichkeit, erscheint der vorgeschlagene Wert in Grauschrift. Der komplette Eintrag wird eingefügt, wenn Sie die **Tabulatortaste** drücken:

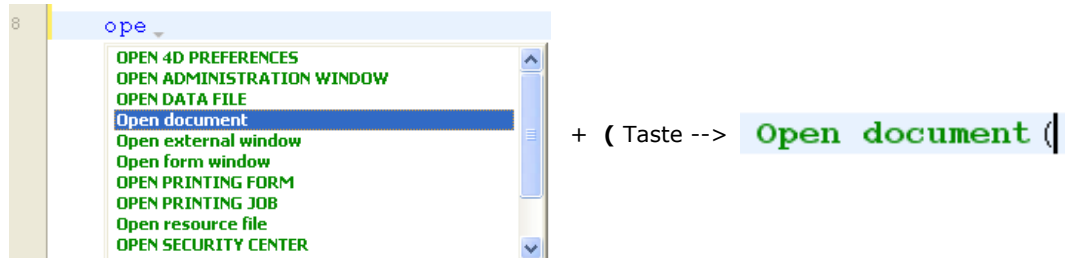
```
aleRT ---> ALERT
```

Gibt es mehrere Vorschläge, zeigt 4D sie in einer PopUp-Liste an, wenn Sie die **Tabulatortaste** drücken:



Die Liste erscheint in alphabetischer Reihenfolge. Doppelklicken Sie auf einen Eintrag, um ihn auszuwählen oder scrollen Sie über die Pfeiltasten an die gewünschte Stelle und drücken die **Enter-Taste**, **Zeilenschaltung** oder **Tabulatortaste**, um den gewählten Wert einzufügen.

Sie können standardmäßig einen vorgeschlagenen Wert, nachdem er ausgewählt wurde, auch über einen der folgenden Begrenzer (; : = < [{ einfügen. Der Wert wird dann zusammen mit dem Begrenzer eingefügt und ist zur Dateneingabe bereit:



Hinweis: Sie können die Verwendung von Begrenzern zum Eingeben vorgeschlagener Werte in den 4D Einstellungen auf der **Seite Methoden** deaktivieren.

Sie können auch die **Esc-Taste** drücken, um die PopUp-Liste zu schließen oder weitere Zeichen einzutippen. Die vorgeschlagenen Werte in der Liste aktualisieren sich, wenn neue Zeichen dazu kommen.

Treffen die eingegebenen Zeichen für verschiedene Objekttypen zu, erscheinen sie in der Liste im entsprechenden Stil. Es lassen sich folgende Objekttypen anzeigen:

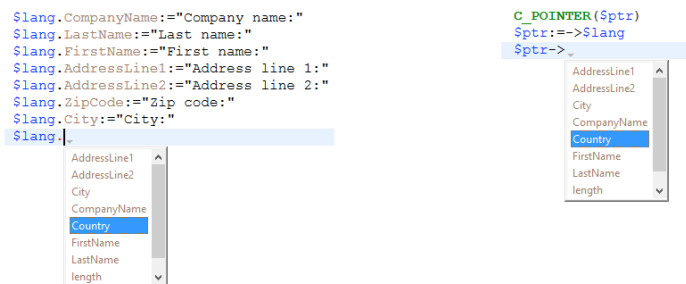
- 4D Befehle
- SQL Befehle
- Benutzermethoden
- Tabellennamen
- Feldnamen
- Konstanten
- Lokale, Prozess- oder Interprozessvariablen, die in der Methode deklariert sind
- Plug-In Befehle
- 4D Schlüsselwörter
- SQL Schlüsselwörter
- Makros (angezeigt zwischen < >)

Hinweis: Aus praktischen Gründen können Sie die automatische Anzeige der Liste mit Vorschlägen für **Konstanten, lokale und Interprozessvariablen** und/oder **Tabellen** deaktivieren. Diese Optionen finden Sie in den 4D Einstellungen auf der [Seite Methoden](#).

Objektattribute

Ist die **Objektnotation** aktiviert, zeigt 4D - unter Berücksichtigung von Groß- und Kleinschreibung - automatisch Vorschläge aller gültigen Namen für Objektattribute in 4D Methoden, wenn Sie:

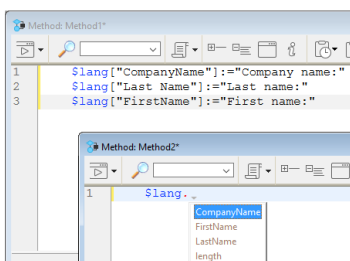
- nach einem Objekt einen Punkt "." setzen oder
- nach einen dereferenzierten Objektzeiger "->" die Tabulatortaste drücken.



Hinweis: Das Attribut *length* erscheint immer (zum Verwenden mit Collections)

Beachten Sie, dass Namen von Objektattributen die ECMA Schreibregeln berücksichtigen müssen. Weitere Informationen dazu finden Sie unter [Identifizier für Objekteigenschaft](#).

Angelegte Attributnamen werden in einer internen globalen Liste gespeichert und sind immer verfügbar, wenn eine Methode geöffnet, geschlossen bzw. der Fokus geändert wird.



Die Liste mit Vorschlägen wird dynamisch aktualisiert, wenn eine Methode bearbeitet wird. Beim Wechseln zwischen Fenstern werden neue/geänderte Attribute immer zur globalen Liste hinzugefügt. Die Liste wird auch aktualisiert, wenn Sie die Methode als Vorschau im Explorer öffnen.

Beim Neustart der Anwendung wird die Liste zurückgesetzt.

Hinweis: Auf der [Seite Methoden](#) der 4D Einstellungen können Sie das automatische Anzeigen von Attributnamen aktivieren/deaktivieren.

Text auswählen

Äußeren Block auswählen

Mit der Funktion **Äußeren Block** auswählen wählen Sie den "eingeschlossenen Block" im Code, der den Einfügepunkt enthält. Der Block kann definiert werden durch:

- Anführungszeichen
- Runde Klammern
- Eine logische Struktur (If/Else/End if, While/End while, Repeat/Until Case of/End case)
- Eckige Klammern

Wurde bereits ein Textblock ausgewählt, wählt die Funktion den auf der nächsthöheren Ebene gelegenen Block usw., bis die gesamte Methode ausgewählt ist.

Wollen Sie die Operation wieder rückgängig machen, wählen Sie unter Windows die Kombination **Strg- + Umschalttaste+B**, auf Mac OS **Befehls- + Umschalttaste+B** und heben die letzte Auswahl auf.

Hinweis: Liegt der Einfügepunkt in einer Struktur vom Typ **If** oder **Else**, enthält der eingeschlossene Block jeweils die **If-** oder **Else-**Anweisung.

Doppelklick


Mit Doppelklick lassen sich "Wörter" auswählen. Enthält der Name eines Elements Leerzeichen, z.B. Befehl, Konstante, Methode, können Sie ihn mit der Kombination **Alt/Wahltaste + Doppelklick** vollständig auswählen.



Tastaturkürzel zum Navigieren

Der Methodeneditor von 4D bietet nützliche Tastaturkürzel, um im Code zu navigieren.

Tastaturkürzel	Aktion
[Shift]+[Pfeil nach rechts]	Auswahl erstellen und Zeichen um Zeichen nach rechts vergrößern, oder Zeichen um Zeichen nach links verkleinern
[Shift]+[Pfeil nach links]	Auswahl Zeichen um Zeichen von rechts aus verkleinern oder Auswahl erstellen und Zeichen um Zeichen nach links vergrößern
[Shift]+[Pfeil nach unten]	Auswahl erstellen und Zeile um Zeile von oben nach unten vergrößern
[Shift]+[Pfeil nach oben]	Auswahl erstellen und Zeile um Zeile von unten nach oben vergrößern
[Strg]+[Shift]+[Pfeil nach rechts]	Auswahl erstellen und Wort um Wort von rechts aus vergrößern
[Strg]+[Shift]+[Pfeil nach links]	Auswahl Wort um Wort von rechts aus verkleinern oder Auswahl erstellen und Wort um Wort von links aus vergrößern
[Strg]+[Pfeil nach rechts]	Einfügekpunkt Wort um Wort von links nach rechts bewegen
[Strg]+[Pfeil nach links]	Einfügekpunkt Wort um Wort von rechts nach links bewegen
[Pos1]	Einfügekpunkt an Zeilenbeginn setzen (gilt nur unter Windows)
[Ende]	Einfügekpunkt an Zeilenende setzen (gilt nur unter Windows)
[Strg]+[Pos1]	Einfügekpunkt an Methodenbeginn setzen
[Strg]+[Ende]	Einfügekpunkt an Methodeneende setzen
[Shift]+[Pos1]	Alle Zeichen der Zeile wählen, die links vom Cursor stehen
[Shift]+[Ende]	Alle Zeichen der Zeile wählen, die rechts vom Cursor stehen
[Bild Pfeil nach oben]	Inhalt der Methode Seite für Seite von unten nach oben scrollen (verändert nicht den Einfügekpunkt)
[Bild Pfeil nach unten]	Inhalt der Methode Seite für Seite von oben nach unten scrollen (verändert nicht den Einfügekpunkt)

Syntaxfehler überprüfen und berichtigen

4D überprüft automatisch die Methodensyntax, um zu sehen, ob sie korrekt ist. Geben Sie Text ein oder wählen einen Ausdruck, der syntaktisch nicht korrekt ist, zeigt 4D das Symbol , um den fehlerhaften Ausdruck zu markieren. Bewegen Sie die Maus über das Symbol, erscheint ein Hilfetipp mit der Beschreibung des Fehlers:

```
7  ALERT("Pi ist gleich:"+String(Arctan(1)*4)
8  Eine rechte Klammer fehlt.
```

Sie können die Syntax der aktuellen Zeile sehr schnell überprüfen, ohne Übergang zur nächsten Zeile, indem Sie die **Eingabetaste** des Nummernblocks drücken. 4D untersucht die Zeile, formatiert diese, markiert mögliche Fehler und setzt den Einfügekpunkt an das Ende der Zeile. Wird eine Zeile der Methode mit inkorrekturer Syntax hervorgehoben, können Sie prüfen und die Eingabe korrigieren. Ist die Zeile jetzt korrekt, entfernt 4D das Fehlersymbol. Sichern oder schließen Sie das Fenster, wird die gesamte Methode bestätigt. Sie können die Methode auch mit der **Eingabetaste** bestätigen.

Beim Bestätigen überprüft 4D jede Zeile auf Syntaxfehler und die Struktur der Anweisungen (If... End if, etc.). 4D prüft auch umschließende Zeichen im Code, wie Klammern oder Anführungszeichen. Haben Sie das öffnende Zeichen getippt, zeigt 4D das dazugehörige schließende Zeichen durch graue Umrahmung des startenden/endenden Zeichens an.

```
ALERT("Pi ist gleich:"+String(Arctan(1)*4))
```

Hinweis: Klicken Sie im Code auf ein startendes Zeichen, zeigt 4D das dazugehörige schließende Zeichen standardmäßig mit grauen Vierecken. Sie können diese Anzeige in den 4D Einstellungen auf der **Seite Methoden** mit der Option "Passende Klammern" verändern oder deaktivieren.

Der Methodeneditor überprüft nur offensichtliche Syntaxfehler, wie z.B. falsche Schreibweise. Er überprüft nicht auf Fehler, die nur bei der Ausführung auftreten. Fehler in der Ausführung erfasst 4D, sobald die Methode abläuft. 4D enthält einen Debugger zum Beheben solcher Fehler. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Debugging**. Auch der Compiler ist eine unumgängliche Hilfe zur Fehlersuche. Weitere Informationen dazu finden Sie im Kapitel **Kompilieren**.

Hilfetipps verwenden

Der Methodeneditor bietet über Hilfetipps verschiedene Informationen im Kontext. Sie erscheinen, wenn Sie die Maus über ein Objekt bewegen.

- **Fehler:** Ziehen Sie die Maus in der linken Spalte vor dem Eingabebereich über das Fehlersymbol, zeigt ein Hilfetipp den Grund des Fehlers an (siehe Abschnitt "Syntaxfehler prüfen und korrigieren").
- **4D Befehl Beschreibung:** Ziehen Sie die Maus über einen 4D Befehl, zeigt ein Hilfetipp die Syntax und eine kurze Beschreibung zum Befehl.

```
SET WEB SERVICE PARAMETER ("Kundenname"; $1)
CALL WEB SERVICE PARAMETER (Name ; Wert {; SoapTyp})
definiert einen Parameter für eine SOAP Anfrage auf Client-Ebene.
```

- **Variablentypen:** Ziehen Sie die Maus über eine Variable, zeigt ein Hilfetipp ihren Typ an (wenn er explizit in der Methode definiert wurde).

```
C_TEXT($1)
C_TEXT($0) $1:C_TEXT
```

- **Projektmethoden:** Ziehen Sie die Maus über eine Projektmethode, zeigt ein Hilfetipp folgendes an:
 - entweder die im Explorer angegebenen Kommentare, sofern vorhanden (siehe **Kommentare verwenden**).
 - oder die ersten Zeilen der Methode, wenn sie Kommentare enthält (Zeilen, die mit // beginnen). Es ist gängige Praxis, die Methode und ihre Parameter in Form von Kommentarzeilen zu Beginn der Methode zu dokumentieren. Sie können diese Information direkt im Hilfetipp erhalten. Sie müssen dafür nur zuvor evtl. vorhandene Kommentare im Explorer entfernen.

Kommentare zu Beginn einer Methode:

```
Methode: proxy_WS_Kunde
1 //
2 // proxy_WS_Kunde
3 // 192.168.10.234/4dwsdl
4 //
5 // Vom 4D SOAP Assistenten automatisch erzeugter Quellcode.
6 // -----
```

Hilfetipp in einer anderen Methode:

```
MakeHotSpot (arCTTheme{ $v1Theme }; Name_KSI
```

```
Make Hot Spot global procedure
Make Hot Spot (String ; String ; String )
Make Hot Spot ( Hot Spot String ; Topic Name { ; Anchor } )-> Text
```

Komplette Beschreibung eines Befehls anzeigen

Sie können jederzeit die HTML Dokumentation eines 4D Programmiersprache Befehls anzeigen: Dazu wählen Sie den kompletten Befehlsnamen aus und setzen einfach den Cursor in den Namen und drücken die Taste **F1**. Die vollständige Beschreibung des Befehls erscheint in einem neuen Fenster Ihres Standard Browser.

4D sucht nach der online Dokumentation des Befehls (auf der Website *4D Doc Center*) oder lokal. Das hängt von den Einstellungen der Anwendung ab (siehe **Speicherort der Dokumentation**).

Kommentar An/Aus

Kommentare sind inaktive Zeilen von Code. Diese Zeilen werden nicht vom Programm interpretiert und nicht ausgeführt, wenn die Methode aufgerufen wird. Um einen Kommentar zu erstellen, setzen Sie einfach die Zeichen // an den Zeilenanfang. Kommentare können bis zu 32.000 Zeichen lang sein.

Sie rufen den Befehl **Kommentar An/Aus** im Menü **Methode** oder im Kontextmenü des Methodeneditors auf. Damit markieren Sie eine Gruppe Code-Zeilen als Kommentar oder entfernen umgekehrt das Zeichen für Kommentar aus der Auswahl.

Zuerst wählen Sie den betreffenden Ausschnitt im Code, dann den Befehl **Kommentar An/Aus**:

```
12 If (Count parameters)>=2
13 EXECUTE FORMULA(Command name (55) + " (['
14 Else
15 If (Count parameters)>=1
16 FORM SET INPUT ($1-> ; "INPUT FORM")
17 End if
18 End if

12 If (Count parameters)>=2
13 EXECUTE FORMULA(Command name (55) + " (['
14 //Else
15 //If (Count parameters)>=1
16 //FORM SET INPUT ($1-> ; "INPUT FORM")
17 //End if
18 End if
```

Enthält die Auswahl nur aktiven Code, wird der Befehl **Kommentar an** angewandt.

Enthält die Auswahl aktiven Code und Zeilen mit Kommentar, wird letzteren ein weiteres Paar Kommentarzeichen (//) hinzugefügt; auf diese Weise bleibt die kommentierte Zeile erhalten, wenn später der Befehl **Kommentar Aus** verwendet wird.

Enthält die Auswahl nur kommentierte Zeilen, wird der Befehl **Kommentar Aus** angewandt.

Hinweis: Der Befehl **Kommentar An/Aus** arbeitet nur mit kompletten Zeilen – er kann nicht für einen Teil der Zeile verwendet werden.

Escape Sequenzen verwenden

Der Methodeneditor lässt Escape Sequenzen (Steuerzeichen) zu. Das ist eine Zeichenfolge, die spezielle Zeichen ersetzt.

Die Sequenz enthält einen linksgerichteten Schrägstrich (\) und ein Zeichen. \t ersetzt z.B. die Eingabe *Character*(Tab) für den Tabulator. Escape Sequenzen erleichtern die Eingabe von Sonderelementen.

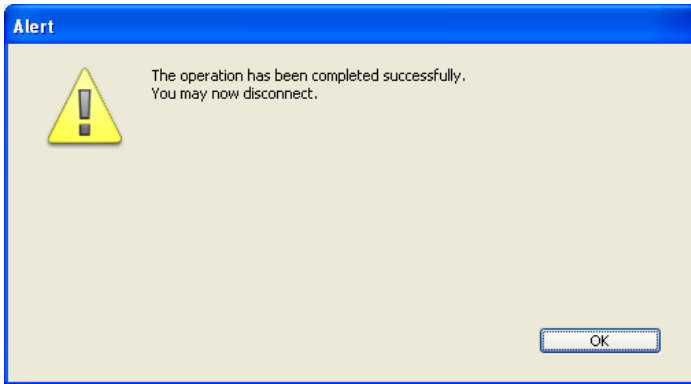
4D arbeitet mit folgenden Sequenzen:

Escape Sequenz	Ersetzte Zeichen
\n	LF (Zeilenvorschub)
\t	HT (Tabulator)
\r	CR (Zeilenschaltung)
\\	\ (linksgerichteter Schrägstrich)
\"	" (Anführungszeichen)

Hinweis: In Escape Sequenzen spielen Klein- oder Großbuchstaben keine Rolle.

Im folgenden Beispiel wird in eine Anweisung die **Zeilenschaltung** (Escape Sequenz\r) eingefügt, um das darunterliegende Dialogfenster anzuzeigen:

```
ALERT("The operation has been completed successfully.\rYou may now disconnect.")
```



Warnung: Der linksgerichtete Schrägstrich (\) dient unter Windows in Zugriffspfaden als Separator. 4D interpretiert im Methodeneditor eingetragene Zugriffspfade unter Windows in der Regel korrekt, weil es den einfachen Schrägstrich (\) durch doppelte Schrägstriche (\\) ersetzt. Aus C:\Ordner wird also C:\\Ordner. Schreiben Sie dagegen "C:\MeineDokumente\Neu", zeigt 4D "C:\\MeineDokumente\Neu" an. In diesem Fall wird der zweite linksgerichtete Schrägstrich \ fälschlicherweise als die Escape Sequenz \N interpretiert. Aus diesem Grund müssen Sie bei Zeichen, die 4D als Escape Sequenz erkennt, stets zwei linksgerichtete Schrägstriche (\\) eingeben.

Code bearbeiten leichter machen

Verschiedene Funktionalitäten im Methodeneditor erleichtern das Lesen von Code und vereinfachen das Durchlaufen der Anweisungen.

Ein/Ausblenden

4D Code innerhalb von Schleifen und Bedingungen lässt sich ein- oder ausblenden. Das macht die Methode übersichtlich und leichter lesbar:

- Aufgeklappter Code:

```
12  If (Count parameters>=2)
13  | EXECUTE FORMULA(Command name (55) + " (" +$2+ " ; "INPUT FORM" ) )
14  | Else
15  | If (Count parameters>=1)
16  |   FORM SET INPUT ($1-> ; "INPUT FORM" )
17  |   End if
18  | End if
19
```


- Zugeklappter Code:

```
12  If (Count parameters>=2)
13  | EXECUTE FORMULA(Command name (55) + " (" +$2+ " ; "INPUT FORM" ) )
14  | Else
15  | If (Count parameters>=1) ...
16  | End if
19
```

Setzen Sie die Maus auf den Icon zum Aufklappen, erscheint ein Hilfetipp, der die ersten Zeilen des ausgeblendeten Code anzeigt.

Sie können den ausgeblendeten Teil des Code auswählen, kopieren, an anderer Stelle einsetzen oder löschen. Das gilt für alle darin enthaltenen Zeilen. Ausgeblendeter Code wird beim Einsetzen an anderer Stelle automatisch eingebildet.

Es gibt verschiedene Wege, um Code ein- und auszublenden:

- Klicken Sie auf das Icon [+] zum Aufklappen, [-] zum Zuklappen oder wählen Sie die Schaltfläche zum Öffnen [...]
- Wählen Sie im Menü **Methode > Auf/Zuklappen**:
 - **Auswahl zuklappen / Auswahl aufklappen**: klappt alle Code-Strukturen innerhalb der Auswahl auf bzw. zu.
 - **Aktuelle Stufe zuklappen / Aktuelle Stufe aufklappen**: klappt die Code-Struktur der Ebene, wo der Cursor steht, auf bzw. zu. Diese Befehle sind auch im Kontextmenü des Editors verfügbar.
 - **Alle zuklappen / Alle aufklappen**: klappt alle Schleifen und Bedingungen einer Methode auf bzw. zu. Diese Befehle sind auch in der Werkzeugleiste des Editors verfügbar: 

Zum Blockanfang oder -ende gehen

Zwei Befehle erleichtern das Navigieren in Code-Strukturen (If...Else...End if, etc.):

- **Beginn des Blocks** setzt den Cursor direkt vor das öffnende Schlüsselwort der aktuellen Struktur
- **Ende des Blocks** setzt den Cursor direkt nach das schließende Schlüsselwort der aktuellen Struktur

Diese Befehle sind im Menü **Methode** oder im Kontextmenü des Editors verfügbar. Sie können auch folgende Tastaturkürzel verwenden:

- Windows: **Strg + Pfeil nach oben** oder **Strg + Pfeil nach unten**,

- Mac OS: **Befehlstaste + Pfeil nach oben** oder **Befehlstaste + Pfeil nach unten**

Bookmarks verwenden

In 4D können Sie bestimmten Zeilen in Ihren Methoden Sprungmarken zuweisen. Sie können dann rasch den Code durchlaufen, indem Sie über spezielle Befehle von einer Sprungmarke zur nächsten gehen:



```
15  
16 ARRAY TEXT ($_CurLang; 0)  
17
```

Sprungmarken wandern mit der Zeile mit, wenn zusätzliche Zeilen in die Methode eingefügt werden. Sie werden mit den Methoden gesichert.

Sprungmarken werden über das Menü **Methode > Bookmarks** verwaltet:

- **Umschalten** weist der aktuellen Zeile, d.h. der Zeile mit dem Cursor, eine Sprungmarke zu, wenn noch keine vorhanden ist. Gibt es dagegen bereits eine Marke, wird sie entfernt. Diese Aktion ist auch im Kontextmenü des Editors über den Eintrag **Bookmark umschalten** oder über die Tastenkombination **Strg+F3** unter Windows, **Befehlstaste+F3** auf Mac OS verfügbar..
- **Alle löschen** entfernt alle Sprungmarken aus dem Fenster im Vordergrund.
- **Gehe zum nächsten/Gehe zum vorherigen**: Damit navigieren Sie in den Sprungmarken, die im Fenster gesetzt sind. Wählen Sie einen der Befehle aus, springt der Cursor auf das erste Zeichen der Zeile, der die Sprungmarke zugeordnet ist. Sie können auch die Tastenkombination **F3** (nächste) bzw. **Shift+F3** (vorige) verwenden.

Hinweis: Sie können Sprungmarken als Marker für Zeilen verwenden, die einen Eintrag enthalten, der über eine Suche gefunden wurde. In diesem Fall fügt 4D automatisch die Sprungmarken hinzu. Weitere Informationen dazu finden Sie im Abschnitt **Suchen und Ersetzen in Methoden**.

Makros erstellen und einsetzen

Sie können in Ihren Methoden Makro-Befehle verwenden. Über solche Befehle sparen Sie viel Zeit beim Schreiben einer Methode.

Was ist ein Makro?

Ein Makro-Befehl ist der Teil eines 4D Codes, der ständig verfügbar und überall in Ihrer Methode einsetzbar ist – unabhängig von der Art der geöffneten Datenbank. Makros können alle Arten von 4D Text, Befehlen und Konstanten enthalten, ebenso wie spezielle Marker, die beim Einsetzen durch Werte ersetzt werden, die sich aus dem Makro Methodenkontext ergeben.

Ein Makro kann z.B. den Marker `<method_name/>` enthalten; beim Einsetzen wird der Marker dann durch den Namen der aktuellen Projektmethode ersetzt.

Makros werden als Text in einer oder mehreren (Text)dateien im XML Format gespeichert. Sie können in einer Liste des Methodeneditors liegen; sie lassen sich aber auch über das Kontextmenü oder die Funktion type-ahead aufrufen.

4D Makros werden im XML Format geschrieben. Sie können entweder die Standard-Makros von 4D verwenden oder diese an Ihre eigenen Bedürfnisse anpassen.

Position der Makros

4D lädt Makros aus einem Ordner mit Namen "**Macros v2.**" Makros müssen die Form von einer oder mehreren XML Dateien haben, die in diesem Ordner liegen.

Der Ordner „Macros v2“ kann an folgende Stellen gesetzt werden:

- Im aktiven 4D Ordner des Rechners. Diese Makros können alle Datenbanken gemeinsam nutzen.
Hinweis: Die Position des aktiven 4D Ordners variiert je nach dem Betriebssystem. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter der Funktion **Get 4D folder**.
- Neben der Strukturdatei der Datenbank: Diese Makros werden nur für diese Struktur geladen.
- Für Komponenten im Ordner Components der Datenbank: Makros werden dann nur geladen, wenn die Komponente installiert ist.

Sie können diese drei Speicherorte auch gleichzeitig benutzen: Sie können an den drei Speicherorten einen Ordner "Macros v2" installieren. Makros werden dann in folgender Reihenfolge geladen: 4D Ordner, Strukturdatei, Komponente 1... Komponente X.

Standard-Makros

4D bietet eine Reihe Standard-Makros, sie entsprechen den Schlüsselwörtern in den früheren Versionen von 4D. Sie liegen in der Datei "Macros.xml" im Ordner „Macros v2“, der beim ersten Start von 4D im aktiven 4D Ordner des Rechners angelegt wird. Sie können diese Datei oder den Inhalt des Ordners später nach Belieben ändern. Bei Problemen können Sie den Ordner auch löschen. 4D erstellt ihn beim nächsten Starten wieder neu.

Eigene Makros hinzufügen

Sie können über einen standardmäßigen Texteditor oder per Programmierung eigene Makros in die Datei "Macros.xml" einfügen. Sie können auch XML Dateien von eigenen Makros in diesem Ordner ablegen.

Im lokalen Modus lässt sich die Makro Datei beim Einsatz von 4D öffnen. Die Liste der verfügbaren Makros wird bei jedem Ereignis aktualisiert, das 4D aktiviert. Sie können z.B. den Texteditor in den Vordergrund holen, die Makro Datei ändern, dann in die 4D Methode zurückkehren: Das neue Makro ist im Methodeneditor verfügbar.

Leere oder fehlerhafte Makros werden nicht angezeigt.

Syntax eigener Makros prüfen

Die Dateien mit Makro-Befehlen müssen konform zum XML Standard sein. Das bedeutet insbesondere, dass am Anfang einer Makro-Datei Anweisungen mit der XML Deklaration `<?xml version="1.0" ...?>` und der Dokument Deklaration `<!DOCTYPE macros SYSTEM "http://www.4d.com/dtd/2007/macros.dtd">` Die verschiedenen Arten von XML Codierung werden unterstützt. Es empfiehlt sich jedoch, eine Codierung zu verwenden, die kompatibel ist zu Mac/PC (UTF-8). 4D bietet eine DTD Datei, mit der sich die Makro-Dateien bestätigen lassen. Diese Datei finden Sie an folgender Stelle:

- Windows: `\Resources\DTD\macros.dtd`
- Mac OS: `:Contents:Resources:DTD:macros.dtd`

Enthält eine Makro-Datei nicht die Anweisungen zur Deklaration oder lässt sie sich nicht bestätigen, wird sie nicht geladen.

Syntax von 4D Makros

4D Makros werden mit individuell angepassten XML Tags, auch "Elemente" genannt, erstellt.

Bestimmte Marker definieren Anfang und Ende. Das sind doppelte Marker vom Typ <tag> </tag>), andere werden durch eingefügte Werte ersetzt (<tag/>).

In Anlehnung an XML Spezifikationen können Marker von Elementen auch Attribute enthalten. Sie sind optional, wenn nichts anderes angegeben ist. Es wird ein Standardwert angegeben. Die Syntax für Elemente mit Attributen lautet:

- Double tags: <tag attribute="value"> </macro>
- Single tags: <tag attribute="value"/>

Lässt ein Element mehrere Attribute zu, können Sie diese, getrennt durch Leerzeichen, in einer Befehlszeile gruppieren: <tag attribute1="value" attribute2="value" attribute3="value"... >

Es folgt die Liste der Marker mit ihrer Verwendung:

Element Tags	Beschreibung
<macros> </macros>	Start und Ende der Makro Datei (zwingender Tag).
<macro> </macro>	Start und Ende für Definition eines Makros und seiner Attribute. <i>Attribute:</i> <ul style="list-style-type: none">- name: Name** des Makro, der in Menüs und Listen des Methodeneditors erscheint (zwingende Attribute).- type_ahead_text: Einzugebende Zeichenkette**, um das Makro über die type-ahead Funktion* aufzurufen.- in_menu: Boolean Wert, der angibt, ob das Makro über das Kontextmenü* aufrufbar ist. Werte = "true" (Standard) oder "false."- type_ahead: Wert, der angibt, ob das Makro über die type-ahead Funktion* aufrufbar ist. Werte = "true" (Standard) oder "false."- method_event: Damit lässt sich das automatische Aufrufen des Makro je nach Bearbeitungsphase der Methode auslösen (Erstellen, Schließen, etc). Werte = "on_load": Das Makro wird beim Öffnen jeder Methode ausgelöst, "on_save": Das Makro wird beim Sichern jeder Methode ausgelöst (Schließen einer geänderten Methode oder Sichern im Menü Datei/Ablage>Speichern), "on_create": Das Makro wird beim Erstellen jeder Methode ausgelöst, "on_close": Das Makro wird beim Schließen jeder Methode ausgelöst. "on_save" und "on_close" können nebeneinander benutzt werden — mit anderen Worten, beide Ereignisse werden erzeugt, wenn eine geänderte Methode geschlossen wird. Dagegen werden "on_create" und "on_load" nie aufeinanderfolgend erzeugt. Mit diesem Attribut können Sie z.B. Methoden beim Erstellen vorformatieren (Kommentare im Kopfbereich) oder Information wie Datum und Uhrzeit beim Schließen speichern.- version: Damit können Sie den neuen Modus zur Unterstützung von Textauswahlen für das Makro einstellen (siehe unteren Abschnitt "Tag <method>"). Um ihn zu aktivieren, übergeben Sie im Makro-Element das Attribut Version="2". Lassen Sie dieses Attribut weg oder übergeben Sie version="1", wird der vorige Modus beibehalten.- in_toolbar: Boolean, der angibt, ob das Makro im Menü der Makro-Schaltfläche der Toolbar vorhanden sein muss. Werte: "true" (Standard) oder "false"
<selection/>	Tag, das beim Einfügen des Makro in Text verwandelt wird. Die Auswahl kann leer sein.
<text> </text>	Start und Ende von Code, der in die Methode eingefügt werden muss. Vor und nach dem Code wird eine Zeilenschaltung eingefügt.
<method> </method>	Gibt Anfang und Ende des Namens der Projektmethode und ihre Parameter an (optional). Die Methode wird ausgeführt, wenn das Makro aufgerufen wird. Sie können es als Parameter in der Form ("param1;param2;...") übergeben. Dieser Parameter wird in der Methode über die Variablen \$1, \$2, empfangen. Weitere Informationen dazu finden Sie im unteren Abschnitt "Tag <method>".
<caret/>	Stelle des Einfügepunkts im Code nach Einfügen des Makro.
<user_4D/>	Tag wird durch den Namen des aktuellen 4D Benutzers ersetzt.
<user_os/>	Tag wird durch den Benutzernamen für das aktuelle Betriebssystem ersetzt.
<method_name/>	Tag wird durch den Namen der aktuellen Projektmethode ersetzt.
<method_path/>	Tag wird durch den kompletten Pfadnamen der aktuellen Projektmethode ersetzt.
<date/>	Tag wird durch das aktuelle Datum ersetzt. <i>Attribute:</i> <ul style="list-style-type: none">- format: 4D Format zur Datumsanzeige. Wurde kein Format definiert, erscheint das Standardformat. Nummer des 4D Formats (0 bis 8).
<time/>	TTag wird durch die aktuelle Zeit ersetzt. <i>Attribute:</i> <ul style="list-style-type: none">- format: 4D Format zur Zeitanzeige. Wurde kein Format definiert, erscheint das Standardformat. Werte = Nummer des 4D Formats (0 bis 6).
<clipboard/>	Tag wird durch den Inhalt der Zwischenablage ersetzt. <i>Attribute:</i> <ul style="list-style-type: none">- index: Einzufügende Zwischenablage. Werte = Nummer der Zwischenablage (0 bis 9).

* Makros lassen sich über das Kontextmenü des Methodeneditors oder die type-ahead Funktion aufrufen (siehe folgende Absätze).

** In Anlehnung an die Regeln der XML Sprache sollten Sie nicht den erweiterten Zeichensatz verwenden. Das sind Zeichen wie Akzente, Anführungszeichen, etc..

Beispiel:

Inhalt des Makro	Kommentare
<?xml version="1.0" ...?	XML Deklaration
<!doctype macros SYSTEM>	Dokument Deklaration
<macros>	Start der XML Datei des Makro
<macro name="RecordLoop">	Start von Definition und Name des Makro
<text>	Start des Makro Codes
For(\$i;1;Records in selection(<Selection/>))	Der Marker <Selection/> wird beim Einsetzen des Makro durch den gewählten Code in der 4D Methode ersetzt, z.B. ein Tabellenname.
SAVE	
RECORD(<Selection/>)	
NEXT	
RECORD(<Selection/>)	
End for	
</text>	Ende des Makro Code
</macro>	Ende der Makro Definition
</macros>	Ende der Makro XML Datei

Tag <method>

Das Tag <method> ermöglicht, Makro-Befehle zu erstellen und einzusetzen, die 4D Projektmethoden ausführen. Über Makro-Befehle, gekoppelt mit Komponenten, können Entwickler komplexe Funktionsabläufe erstellen. Das folgende Makro löst zum Beispiel die Ausführung der Methode *MyMethod* aus, wobei der Name der aktuellen Methode als Parameter dient:

```
<method>MyMethod("<method_name/>")</method>
```

Der Code einer aufgerufenen Methode wird in einem neuen Prozess ausgeführt. Dieser wird beendet, sobald die Methode ausgeführt ist.

Hinweis: Der Strukturprozess bleibt eingefroren, bis die Ausführung der aufgerufenen Methode abgeschlossen ist. Sie müssen sicherstellen, dass die Ausführung rasch ist und kein Risiko besteht, das Programm zu blockieren. Tritt dieser Fall ein, verwenden Sie die Tastenkombination **Strg+F8** unter Windows, **Befehlstaste+F8** auf Mac OS), um den Prozess zu „killen“.

Makros aufrufen

Makros lassen sich über das Kontextmenü oder die Werkzeugleiste des Methodeneditors, über die Funktion type-ahead oder eine spezifische Liste im unteren Fensterteil des Methodeneditors aufrufen. Sie können für jedes Makro einstellen, ob es über das Kontextmenü und/oder die Funktion type-ahead aufrufbar ist.

Kontextmenü und Werkzeugleiste

Alle Makros werden standardmäßig mit dem hierarchischen Befehl **Makro einfügen** über das Kontextmenü des Methodeneditors oder das Symbol für Makros in der Werkzeugleiste aufgerufen.

Das Attribut *in_menu* des Tag <macro> bestimmt, ob das Makro im Menü erscheint oder nicht.

Im Kontextmenü erscheinen Makros in der Reihenfolge in der Datei "Macros.xml". Sie können die Reihenfolge durch Ändern dieser Datei umstellen.

Type-ahead

Alle Makros sind standardmäßig über die Funktion type-ahead verfügbar. Weitere Informationen dazu finden Sie im Abschnitt **Methode schreiben**. Über das Attribut *type Ahead* des Tag <macro> können Sie ein Makro von dieser Funktionsweise ausschließen.

Hinweis: Ein Makro mit dem Tag <selection/> erscheint nicht im PopUp-Menü von type-ahead.

Liste im Methodeneditor

Sie können Ihre Makros in einer Liste des Methodeneditors anzeigen. Weitere Informationen dazu finden Sie im Abschnitt **Methode schreiben**. Sie rufen das Makro durch Doppelklick auf einen Eintrag in der Liste auf. Es ist nicht möglich, ein bestimmtes Makro aus der Liste auszuschließen.

Hinweise zur Kompatibilität

Die Unterstützung von Makros kann sich von einer 4D Version zur nächsten ändern. Damit die verschiedenen Versionen kompatibel und Ihre eigenen Anpassungen erhalten bleiben, entfernt 4D keine früheren Versionen. Wollen Sie die neuesten verfügbaren Funktionalitäten nutzen, müssen Sie Ihre Version entsprechend anpassen.

Variablen zur Textauswahl für Methoden

In 4D Versionen vor v11 hat 4D automatisch eine Reihe von Prozessvariablen verwendet, um Text in Methoden mit dem Tag <method>: Eingabevariablen (*_textSel*, *_blobSel*, *_selLen*, *_textMethod*, *_blobMethod*, *_methodLen*), um Text zu finden, Ausgabevariablen (*_textReplace*, *_blobReplace*, *_action*) um Text einzufügen. Diese Funktionsweise wird zur Wahrung der Kompatibilität in 4D v11 weiter unterstützt, sie ist jedoch aus folgenden Gründen jetzt überholt:

- Die Verwendung von BLOB Variablen bei Text mit mehr als 32.000 Zeichen ist nicht mehr notwendig,

- Die Verwaltung von Variablen ist nicht kompatibel mit der neuen Komponenten Architektur, denn hier ist der Ausführungsbereich von Variablen aufgeteilt. Eine Komponente in Version 11 kann nicht über vordefinierte Variablen auf Text von Methoden der Host Datenbank zugreifen und umgekehrt.

Von daher empfiehlt es sich, über die 4D Befehle **GET MACRO PARAMETER** und **SET MACRO PARAMETER** den neuen Modus zur Unterstützung von Textauswahlen zu verwenden. Damit können Sie die aufgeteilten Ausführungsbereiche der Host Datenbank bzw. Komponente übergehen und so Komponenten für die Verwaltung der Makros erstellen. Um diesen neuen Modus für ein Makro zu aktivieren, müssen Sie das Attribut Version im Makro-Element mit dem Wert 2 deklarieren. Dann verwaltet 4D nicht mehr die vordefinierten Variablen `_textSel`, `_textReplace`, etc. sondern verwendet die 4D Befehle **GET MACRO PARAMETER** und **SET MACRO PARAMETER**. Die Anweisung dazu lautet:

```
<macro name="MyMacro" version="2">
--- Text of the macro ---
</macro>
```

Übergeben Sie dieses Attribut nicht, wird der vorige Modus beibehalten.

Im XML Standard nicht kompatible Elemente

Ab 4D v11 gelten für Makro-Dateien strikte Syntaxregeln, um den XML Standard zu berücksichtigen. Das kann zu Inkompatibilität mit dem Code in Makros aus früheren Versionen führen und das Laden von XML Dateien verhindern. Die Hauptursachen dafür sind:

- Kommentare vom Typ `// my comment` die in bisherigen Makro-Formaten innerhalb von `<macro>` Elementen verwendet. Kommentarzeilen müssen standardmäßig die Form `<!-- my comment -->` haben.
- Die Symbole `<>` insbesondere für Interprozess Objektnamen verwendet, müssen kodiert werden. Die Variable `<>params` muss jetzt `<>params` lauten.
- In früheren 4D Versionen konnte das Tag `<macros>` das den Start deklariert hat, wegleiben. Jetzt ist es zwingend; denn sonst wird die Datei nicht geladen.

Datei Macros.xml aktualisieren

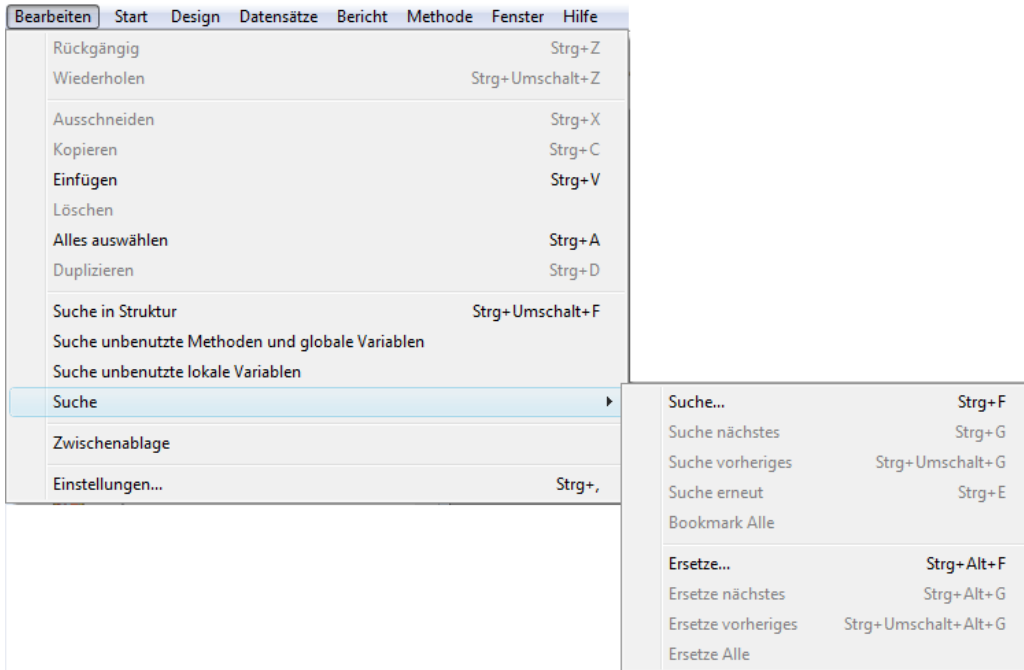
In 4D Version 12 gibt es neue Makro-Befehle, die den Einsatz der SQL Befehle erleichtern. Da die Datei "Macros.xml" individuell anpassbar ist, wird beim Installieren einer neuen Version von 4D die vorhandene Datei nicht automatisch ersetzt. Um die neuen SQL Makro-Befehle von 4D v12 zu nutzen, müssen Sie folgendes ausführen:

- entweder die Datei "Macros.xml" im Ordner "Macros v2" löschen (wenn sie nie verändert wurde) und 4D starten. Die neue Datei wird automatisch angelegt.
- oder die neuen Makros manuell in die Datei "Macros.xml" im Ordner "Macros v2" einfügen (wenn Sie den Inhalt verändert haben). Die neue Vorlagendatei der Macros liegt im Programm 4D unter `4D\Resources\en.lproj` oder `4D\Resources\de.lproj`.

Suchen und Ersetzen in Methoden

Der Methodeneditor bietet spezifische Funktionen zum Suchen und Ersetzen, die für das aktuelle Fenster gelten. Über den Suchbereich in der Werkzeugleiste können Sie einfache Suchen ausführen und den Suchdialog aufrufen (siehe **Methodeneditor**).

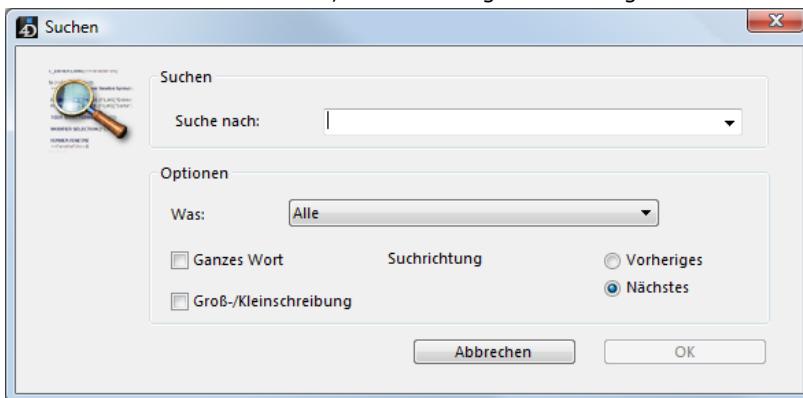
Die Befehle **Suche/Ersetze** für Methoden liegen im Menü **Bearbeiten** von 4D:



Hinweis: Die Suchbefehle gelten nicht nur für den Methodeneditor. Sie lassen sich auch verwenden, um nach einem Wert in allen Methoden zu suchen. Weitere Informationen dazu finden Sie im Abschnitt **Suchen und Ersetzen in der Struktur**.

Suchen

Wählen Sie den Befehl **Suche**, erscheint folgendes Dialogfenster:



Die hier definierte Suche wird in der Methode im Vordergrund ausgeführt.

- Im Bereich "Suche nach" geben Sie eine Zeichenfolge für die Suche ein. Sie dient als Combo Box, welche die letzten 15 Einträge für Suchen bzw. Ersetzen während der 4D Sitzung enthält. Markieren Sie Text, bevor Sie den Befehl **Suche** auswählen, erscheint er in diesem Bereich. Sie können dann entweder diesen Text verwenden oder ihn durch einen anderen ersetzen.
- Die Option **Ganzes Wort** beschränkt die Suche auf das exakte Vorkommen eines Wortes. Mit dieser Option findet beispielsweise die Suche nach "client" weder "clients" noch "myclient". Diese Option ist standardmäßig inaktiv. So findet die Suche nach "var" auch "Meinevar" und "Variation". Beachten Sie, dass die Option **Ganzes Wort** im Gegensatz zur Option **Whole Object Name** für übergreifende Suchläufe in der Datenbank, Objektnamen außer Acht lässt. Suchen Sie mit der ersten Option in einer Methode nach dem Wort "Meine", wird die Variable "MeineVariable" gefunden; die Option **Whole Object Name** gibt kein Ergebnis zurück, da sie nur nach kompletten Objektnamen sucht, das wäre hier "MeineVariable".
- Die Option **Groß-/Kleinschreibung** berücksichtigt die Groß- und Kleinschreibung für Einträge im Bereich "Suche nach". Eine Suche nach "MeineVar" findet also nicht "meineVar".
- Die Optionsfelder **Vorheriges/Nächstes** bestimmen die Suchrichtung, vom Cursor ausgehend entweder zum Anfang oder zum Ende der aktuellen Methode.

Klicken Sie auf die Schaltfläche **OK**, startet die Suche. Im Fenster des Methodeneditors wird das erste Element ausgewählt, auf welches die Suchkriterien zutreffen. Über die Befehle **Suche Nächstes** und **Suche Vorheriges** im Menü **Bearbeiten** finden Sie weitere passende Elemente.

Suche erneut

Der Befehl **Suche erneut** im Menü **Bearbeiten** findet Zeichenfolgen, die mit der gewählten übereinstimmen. Dieser Befehl ist nur aktiv, wenn im Methodeneditor mindestens ein Zeichen ausgewählt wurde. Diese Suche entspricht dem Typ "Suche nächstes" in der aktuellen Methode.

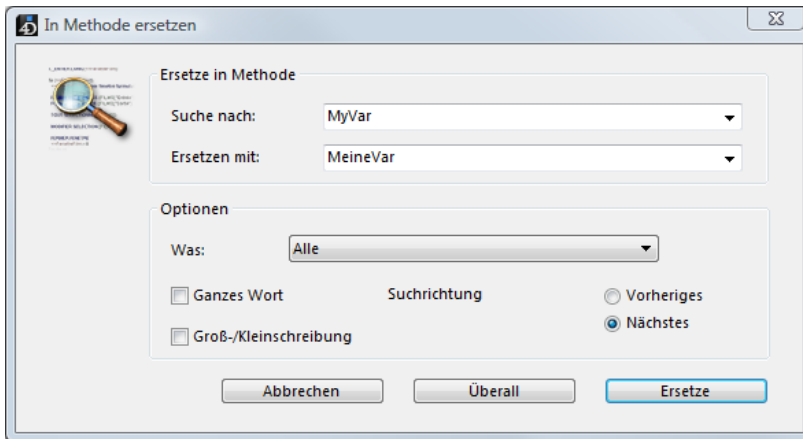
Bookmark Alle

Der Befehl **Bookmark All** ist aktiviert, wenn im Dialogfenster Suchen bzw. Ersetzen bereits eine Suche festgelegt wurde. Wählen Sie diesen Befehl, setzt 4D bei jeder Zeile, die einen Eintrag enthält, der zum "aktuellen" Suchkriterium passt, eine Sprungmarke. So erhalten Sie rasch einen Überblick über alle Suchergebnisse.

Weitere Informationen dazu finden Sie im Abschnitt "Bookmarks verwenden" unter **Methode schreiben**.

Ersetzen

Wählen Sie den Befehl **Ersetzen**, erscheint folgendes Dialogfenster:



- Im Bereich "Suche nach" geben Sie eine Zeichenkette für die Suche ein. Er dient als Combo Box, welche die letzten 15 gesuchten Einträge während der 4D Sitzung enthält. Markieren Sie Text, bevor Sie den Befehl **Ersetze** auswählen, erscheint er in diesem Bereich.
- Im Bereich "Ersetzen mit" definieren Sie die Zeichenfolge, welche die oben gesuchte Zeichenfolge ersetzen soll. Er dient auch als Combo Box, welche die letzten 15 ersetzten Einträge während der 4D Sitzung enthält.
- Die Option **Ganzes Wort** beschränkt Suchen/Ersetzen auf das exakte Vorkommen eines Wortes. So findet z.B. die Suche nach "client" weder "clients" noch "meinclient".
- Die Option **Groß-/Kleinschreibung** berücksichtigt die Groß- und Kleinschreibung für Einträge im Bereich "Suche nach". So findet eine Suche nach "MeineVar" nicht "meineVar".
- Die Optionsfelder **Vorheriges** und **Nächstes** bestimmen die Suchrichtung, ausgehend von der Cursor-Position, entweder zum Anfang oder zum Ende der aktuellen Methode.

Klicken Sie auf die Schaltfläche **Ersetze**, startet die Suche und ersetzt das erste gefundene Element. Mit den Befehlen **Ersetze nächstes** und **Ersetze vorheriges** aus dem Menü **Bearbeiten** können Sie weitersuchen und ersetzen.

Klicken Sie auf **Ersetze Alle**, werden alle Vorkommen in der geöffneten Methode ersetzt, auf welche die Suchkriterien zutreffen.

Gehe zu Definition

Der Befehl **Gehe zu Definition** öffnet die Definition eines referenzierten Objekts (Methode, Feld, Tabelle oder Formular) im Methodeneditor direkt in einem neuen Fenster. Dazu setzen Sie einfach den Cursor in den Objektnamen oder markieren den Namen und wählen im Menü **Methode** bzw. im Kontextmenü des Editors den Befehl **Gehe zu Definition**.

Hinweis: Ist der Name des Objekts ausgewählt, können Sie die Aktion auch über die Tastenkombination **Strg+K** unter Windows bzw. **Befehl+K** auf Mac OS aufrufen.

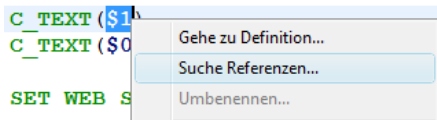
Dieser Befehl funktioniert mit folgenden Objekten:

- **Projektmethode** zeigt den Inhalt der Methode in einem neuen Fenster des Methodeneditors
- **Feld** zeigt die Feldeigenschaften im Inspektor des Strukturfensters
- **Tabelle** zeigt die Tabelleneigenschaften im Inspektor des Strukturfensters
- **Formular** zeigt das Formular im Formulareditor
- **Variable** (lokal, Prozess, Interprozess oder \$n Parameter) zeigt die Zeile, welche die Variable deklariert, in der aktuellen Methode oder über die Compiler Methoden.
- **4D Befehl** zeigt die Beschreibung in der HTML Dokumentation

Suche Referenzen

Der Befehl **Suche Referenzen** im Menü **Methode** bzw. im Kontextmenü des Methodeneditors findet alle Objekte (Methoden und Formulare) in der Datenbank, wo auf den aktuellen Eintrag der Methode verwiesen wird.

Der aktuelle Eintrag ist entweder der ausgewählte oder der, wo der Cursor steht. Es kann ein Feld- oder Variablenname, ein Befehl, String, usw. sein. Die folgende Aktion findet z.B. alle Vorkommen der Variablen \$1 in der Datenbank:



Dieser Befehl zeigt die Ergebnisse in einem neuen Fenster an. Weitere Informationen dazu finden Sie im Abschnitt **Suchergebnis**.

Suche Aufrufe

Der Befehl **Suche Aufrufe** im Menü **Methode** ist nur für Projektmethoden aktiviert. Er sucht nach allen Objekten (andere Methoden oder Menüs), die auf die Projektmethode verweisen.

Hinweis: Der Befehl **Suche Aufrufe** ist auch im 4D Explorer auf der **Seite Methoden** verfügbar.

Dieser Befehl zeigt die Ergebnisse in einem neuen Fenster an. Weitere Informationen dazu finden Sie im Abschnitt **Suchergebnis**.

Gehe zu Zeile

Dieser spezifische Suchbefehl liegt im Menü **Methode**. Er öffnet ein Dialogfenster, um die gesuchte Zeilennummer einzutragen. Klicken Sie auf die Schaltfläche **OK**, findet der Editor diese Zeile in der Methode und hebt sie hervor. Diese Suche ist hilfreich in Verbindung mit dem Compiler, der Runtime-Fehler durch die Nummer der Zeile anzeigt, in welcher sie auftreten.

Sie können wählen, ob die Zeilennummern im Fenster des Methodeneditors angezeigt werden sollen. Weitere Informationen dazu finden Sie im Abschnitt **Methodeneditor**.

Methoden ausführen

Projektmethode in Ihrer Anwendung werden in der Regel automatisch beim Einsatz der Anwendung über Menübefehle, Schaltflächen, andere Methoden, etc. aufgerufen. Analog zu Datenbankmethoden werden sie ausgeführt, wenn in der Anwendung bestimmte Ereignisse eintreten.

Für Testzwecke und zur Fehlerbehebung ermöglicht 4D aber auch, Projektmethode und bestimmte Datenbankmethoden im Designmodus manuell auszuführen. In solchen Fällen lässt sich die Methode in einem neuen Prozess ausführen bzw. direkt im Schrittmodus (Debugger), um die Ausführung Schritt für Schritt zu prüfen.

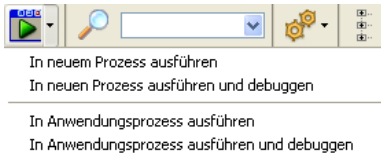
Im Client/Server-Betrieb können Sie bestimmen, ob 4D Server eine Projektmethode auf dem Server oder anderen Client-Rechnern ausgeführt werden soll.

Sie können Methoden auf zwei Arten ausführen:

- Über den Methodeneditor
- Über das Fenster Methode ausführen (nur Projektmethode)

Über den Methodeneditor

Jedes Fenster des Methodeneditors hat oben links ein Icon mit grünem Pfeil, um die aktuelle Methode auszuführen. Über das zu geordnete PopUp-Menü können Sie zwischen verschiedenen Arten wählen:



Dieses Icon ist nur für Projektmethode und folgende Datenbankmethoden aktiv:

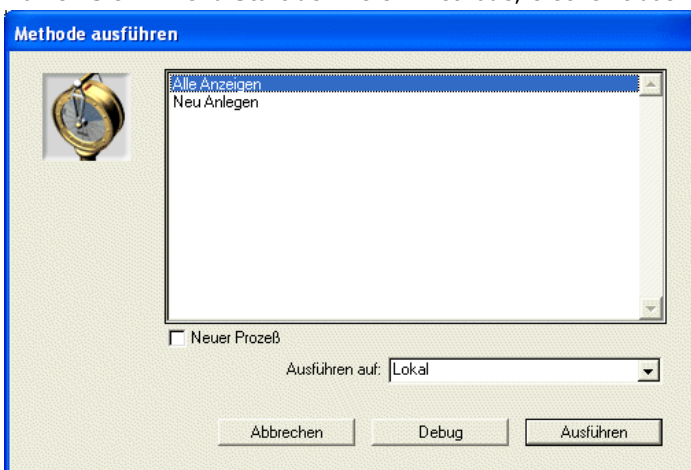
- On Startup
- On Exit
- On Server Startup
- On Server Shutdown

Es gibt folgende Ausführungsarten:

- **In neuem Prozess ausführen:** Erstellt einen Prozess und führt die Methode im Standardmodus in diesem Prozess aus.
- **In neuem Prozess ausführen und debuggen:** Erstellt einen neuen Prozess und zeigt die Methode in diesem Prozess im Debugger an zur Ausführung Schritt für Schritt.
- **In Anwendungsprozess ausführen:** Führt die Methode im Standardmodus im Kontext des Anwendungsprozesses aus, d.h. im Fenster zur Anzeige der Datensätze.
- **In Anwendungsprozess ausführen und debuggen:** Zeigt die Methode im Debugger an, um sie Schritt für Schritt im Kontext des Anwendungsprozesses auszuführen, d.h. im Fenster zur Anzeige der Datensätze.

Über das Fenster Methode ausführen

Wählen Sie im Menü Start den Befehl Methode, erscheint das Fenster *Methode ausführen*:



Es zeigt alle Projektmethode der Datenbank, inkl. gemeinsam genutzter Projektmethode von Komponenten. Dagegen erscheinen keine Projektmethode, die ausgeblendet wurden.

Um eine Projektmethode auszuführen, markieren Sie den Namen in der Liste und klicken auf die Schaltfläche **Ausführen**. Um eine Methode Schritt für Schritt auszuführen, klicken Sie auf die Schaltfläche Debug. Weitere Informationen zum Debugger finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Debugging**.

Hinweis zu 4D Server: Die Option **Debug** ist nicht verfügbar, wenn die Methode auf dem Server ausgeführt wird.









Markieren Sie das Kontrollkästchen **Neuer Prozeß**, wird die gewählte Methode in einem anderen Prozess ausgeführt. Führt die Methode eine zeitaufwändige Aufgabe aus, etwa das Drucken einer großen Zahl von Datensätzen, können Sie währenddessen mit ihrer Datenbank weiterarbeiten, wie z.B. Datensätze zu einer Tabelle hinzufügen oder Daten in Diagrammen darstellen.

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Einführung in Prozesse**.

Hinweis zu 4D Server:

- Soll die Methode auf dem Server Rechner und nicht auf dem Client-Rechner ausgeführt werden, wählen Sie im DropDown-Menü **Ausführen** den Eintrag **Auf 4D Server ausführen**. Dann wird auf dem Server Rechner ein neuer Prozess, genannt *Serverprozedur*, erstellt, um die Methode auszuführen. Diese Option dient dazu, den Netzwerkverkehr zu verringern und die Funktionsweise von 4D Server zu optimieren, insbesondere für Methoden, die Daten von der Festplatte aufrufen. Die meisten Methoden können auf einem Server ausgeführt werden, es sei denn, die Methode erzeugt oder modifiziert auf dem Client-Rechner Oberflächenelemente.
- Sie können die Methode auch auf einer anderen Arbeitsstation ausführen. Sie erscheinen in diesem DropDown-Menü jedoch nur, wenn sie zuvor "registriert" wurden. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter dem Befehl **REGISTER CLIENT**. Standardmäßig ist die Option **lokal** ausgewählt. In der Einzelplatzversion von 4D steht nur diese Option zur Verfügung.

Benutzer und Gruppen

-  Überblick über das Zugriffssystem
-  Designer und Administrator
-  Kennwortsystem aktivieren
-  Standardbenutzer einrichten
-  Benutzer und Gruppen verwalten
-  Benutzern Zugriff auf die Designumgebung geben
-  Datenbankobjekten eine Gruppe zuweisen
-  Kennwortsystem pflegen

Überblick über das Zugriffssystem

Wenn mehr als eine Person eine Datenbank nutzt, möchten Sie unter Umständen den Zugriff auf die Datenbank steuern oder verschiedene Funktionen und Oberflächen verschiedenen Benutzern zur Verfügung stellen. Wenn Sie Anwendungen entwickeln, um sie in einer Mehrplatzumgebung oder für das World Wide Web zu verwenden, kann der Schutz kritischer Daten essentiell sein. Sie können diesen Schutz durch das Zuweisen von Kennwörtern an Benutzer und Erzeugen von Zugriffsgruppen herstellen, die verschiedene Zugriffsrechte auf die Information in der Datenbank oder die Datenbankfunktionen haben.

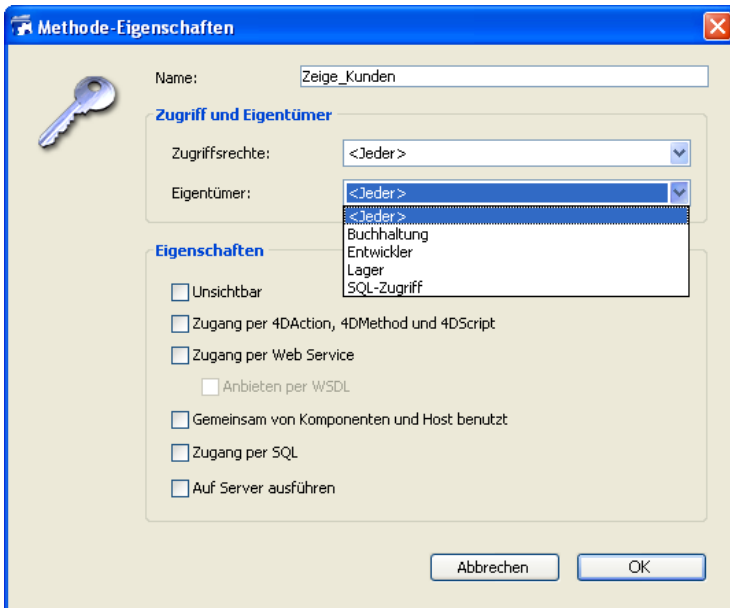
Einen allgemeinen Überblick über die Sicherheitsvorkehrungen in 4D finden Sie im [4D Security guide](#).

Zugriffsrechte

Zugriffsrechte zuweisen

Das Kennwortsystem von 4D beruht auf Benutzern und Gruppen. Sie legen Benutzer fest und weisen Kennwörter zu, teilen Benutzer in Gruppen ein und weisen jeder Gruppe Zugriffsrechte zu entsprechenden Teilen der Datenbank zu. Gruppen können Zugriffsrechte für Operationen an Datensätzen in der Tabelle und für Tabellendefinition zugewiesen werden.

Das folgende Beispiel zeigt „Eigentümer“ Zugriffsrechte für die Projektmethode **Zeige_Kunden** die einer Gruppe zugewiesen wurde. Sie können Gruppen generell Rechte auf der Ebene Zugriff (Benutzung) und/oder „Eigentümer“ (Ändern) zuweisen.



Weitere Informationen dazu finden Sie unter **Kennwortsystem pflegen**.

Ein Benutzer wählt zum Öffnen der Datenbank seinen Benutzernamen aus oder tippt ihn ein und gibt sein Kennwort ein. Dann kann er je nach Gruppenzugehörigkeit und deren Rechten die Teile der Datenbank benutzen, die durch das Zugriffssystem festgelegt sind.

Zugriff auf geschützte Anwendungen

Sie können den Zugriff auf die Datenbank in den Datenbank-Eigenschaften auf der Seite "Sicherheit" konfigurieren (siehe unter **Seite Sicherheit**).

- Standardmäßig erscheint folgendes Dialogfenster:

In diesem Dialogfenster wählt der Benutzer in der Benutzerliste seinen Namen aus und gibt im Eingabebereich sein Kennwort ein.

- Deaktivieren Sie in den Datenbank-Eigenschaften die Option **Benutzerliste im Dialogfenster Kennwort anzeigen**, erscheint folgendes Dialogfenster:

Hier muss der Benutzer seinen Namen und sein Kennwort eingeben, was die Sicherheit der Anwendung verstärkt.

- Haben Sie in den Datenbank-Eigenschaften einen **Standardbenutzer** für Ihre Datenbank definiert und ein Kennwort zugeordnet, erscheint folgendes Dialogfenster:

Benutzer müssen nur das Kennwort eingeben.

- Wurde dem Standardbenutzer kein Kennwort zugeordnet, erscheint der Kennwortdialog nicht. Die Datenbank öffnet sich direkt. In diesem Fall hat jeder Benutzer die Zugriffsrechte und -einschränkungen, die für den Standardbenutzer gelten.

Ist in den Datenbank-Eigenschaften **die Option Die Anwender können ihre Kennwörter ändern** markiert, erscheint im Eingabedialog die Schaltfläche **Ändern**. Damit kann der aktuelle Benutzer sein eigenes Kennwort ändern.

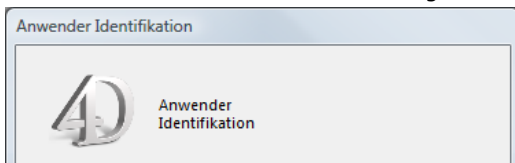
Hinweis zu 4D Server: Sie können die Parameter und die Anmelde-Identifizier zur Datenbank in einer Zugriffsdatei vom Typ .4DLink sichern. Weitere Informationen dazu finden Sie im *4D Server Handbuch* im Abschnitt **Anmeldung an eine 4D Server Datenbank**.

Der Benutzer verwendet die Anwendung wie gewohnt. Versucht er, ein Objekt (Formular, Menübefehl, Methode) zu verwenden, auf die seine Gruppe keinen Zugriff hat, zeigt 4D eine Fehlermeldung, z.B. „Ihr Kennwort erlaubt nicht, dieses Formular zu benutzen.“

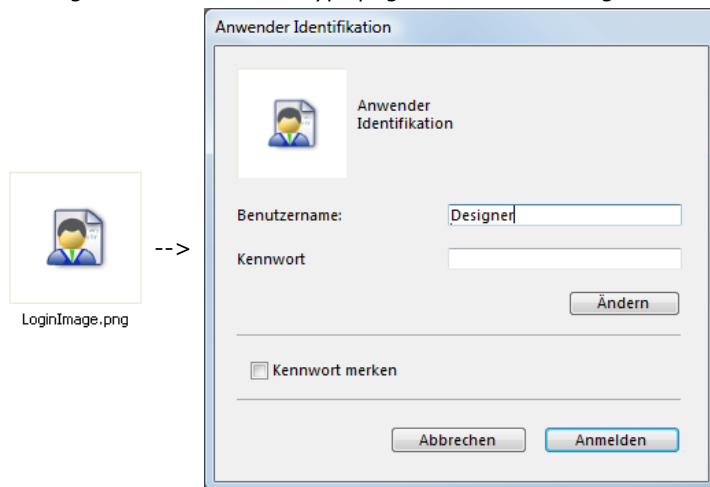
Hinweis: Wenn eine Methode **ON ERR CALL** installiert ist, wird die Fehlermeldung bei Methoden nicht angezeigt.

Icon im Anmeldedialog ersetzen

Sie können das Icon im Kennwortdialog durch ein eigenes ersetzen. Standardmäßig erscheint das 4D Logo.



Um dieses Icon durch ein eigenes zu ersetzen, müssen Sie nur ein Bild mit Namen LoginImage.png in den Ordner Resources der Datenbank legen. Er liegt neben der Strukturdatei der Datenbank (siehe **Beschreibung der 4D Dateien**). Das eigene Bild muss vom Typ "png" und 80x80 Pixel groß sein.



Kennwort merken

Markiert der Benutzer die Option **Kennwort merken**, sichert 4D das Kennwort beim Bestätigen des Dialogfensters und verwendet es automatisch wieder bei nachfolgenden Anmeldungen. Das Kennwort ist verschlüsselt und wird lokal in der Datei .4DLink der entsprechenden Anwendung gespeichert.

Aus diesem Grund können Sie diese Funktionalität nicht für alle 4D Applikationen verwenden. Nachfolgende Tabelle zeigt, wann sie verfügbar ist:

	4D lokaler Modus	4D remote Modus
Datenbank ohne Engine	Ja	Ja
Datenbank mit einkompilierter 4D Engine	Nein	Ja

Externen Zugriff über das 4D Kennwortsystem steuern

Haben Sie ein Kennwortsystem eingerichtet, können Sie es für verschiedene Arten von externen Zugriffen auf die 4D Datenbank verwenden. Sie können je nach Kontext eine spezifische Zugriffshierarchie definieren.

Sie können das Benutzer- und Gruppensystem von 4D verwenden für:

- Den **in 4D integrierten HTTP Server**. Weitere Informationen über die Berücksichtigung von 4D Benutzern und Gruppen beim Anmelden an den 4D Web Server finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Sicherheit der Verbindung**.
- Den **in 4D integrierten SQL Server**. Weitere Informationen über Zugriffsgruppen für den in 4D integrierten SQL Server finden Sie im Handbuch *4D SQL Reference* im Abschnitt **Configuration of 4D SQL Server**.

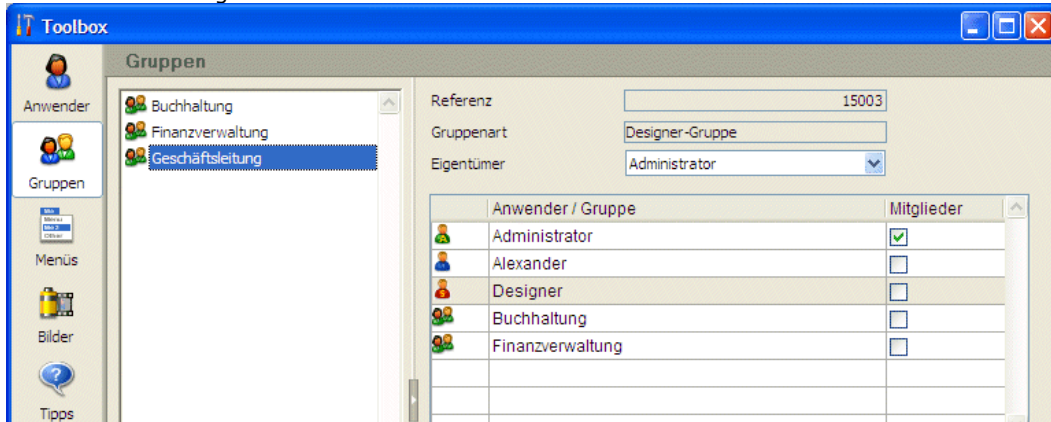
Ein Schema für Zugriffshierarchie

Die beste Möglichkeit, den Schutz Ihrer Datenbank zu gewährleisten und Benutzer mit verschiedenen Zugriffsrechten auszustatten, ist die Verwendung eines Schemas für Zugriffshierarchie. Benutzer lassen sich entsprechenden Gruppen zuweisen und Gruppen können ineinander verschachtelt werden, um eine Hierarchie der Zugriffsrechte zu erzeugen. Nachfolgend sehen Sie ein Beispiel.

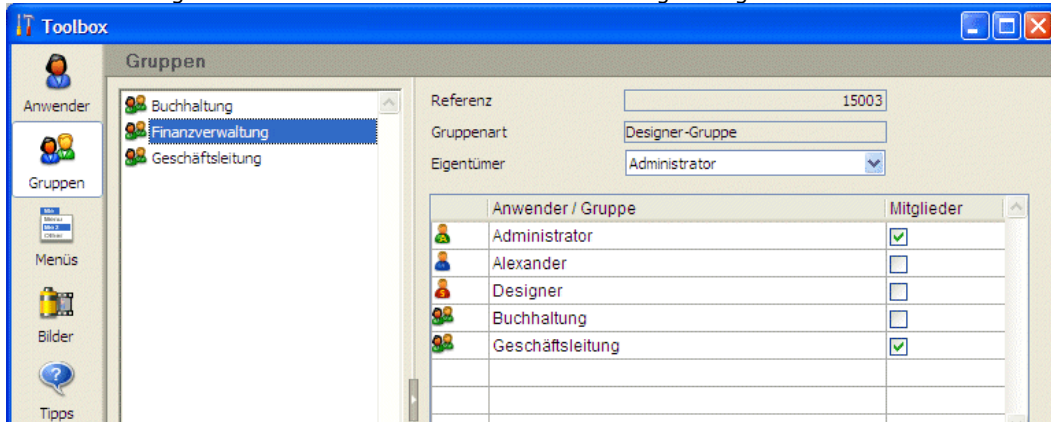
In diesem Beispiel wird ein Benutzer je nach seiner Zuständigkeit einer von drei Gruppen zugeordnet. Benutzer der Gruppe Buchhaltung sind für Dateneingaben zuständig. Benutzer der Gruppe Finanzverwaltung sind für die Datenpflege zuständig, wozu auch die Aktualisierung der Datensätze und das Löschen veralteter Datensätze gehört. Benutzer der Gruppe Geschäftsleitung sind zuständig für das Analysieren von Daten, sie können auch Suchläufe durchführen und Berichte erstellen.

Die Gruppen werden dann so verschachtelt, dass die Rechte beim Zuweisen der Gruppen korrekt verteilt werden.

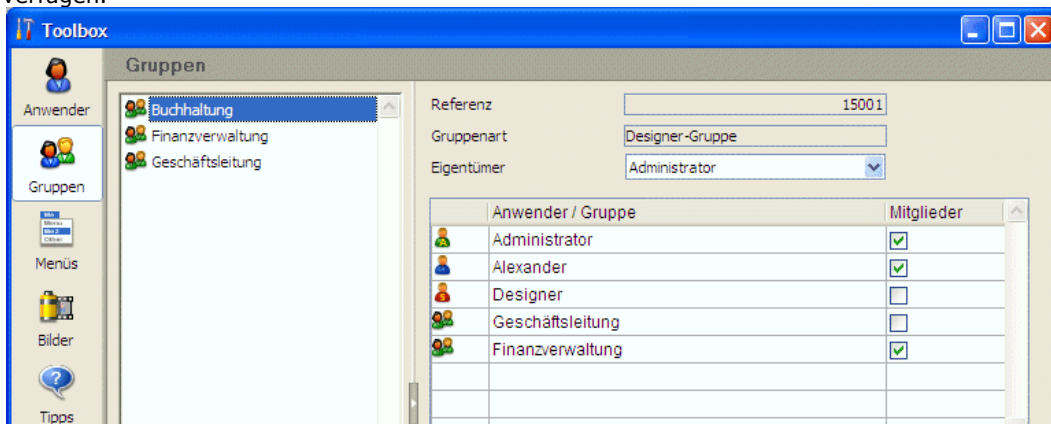
- Die Geschäftsleitung enthält nur Benutzer der höchsten Ebene.



- Die Finanzverwaltung enthält Benutzer für die Datenpflege und die Geschäftsleitung, so dass die Gruppe der Geschäftsleitung auch über die Rechte der Finanzverwaltung verfügt.



- Die Gruppe Buchhaltung enthält Benutzer für die Dateneingabe, sowie Benutzer der Gruppe Finanzverwaltung, so dass Benutzer, die zur Finanzverwaltung und zur Geschäftsleitung gehören, auch über die Rechte der Buchhaltung verfügen.



Sie können je nach Zuständigkeit festlegen, welcher Gruppe welche Zugriffsrechte zugewiesen werden. Wenn Sie zum Beispiel der Gruppe Buchhaltung einem Eingabeformular zuweisen, dann bedeutet dies, dass jeder dieses Eingabeformular verwenden kann. Wenn Sie die Gruppe Finanzverwaltung dem Formular zuweisen, wird es beschränkt auf Mitglieder der Finanzverwaltung und Geschäftsleitung.

Wenn Sie die Gruppe Geschäftsleitung zuweisen, können nur Mitglieder dieser Gruppe das Formular verwenden. Solch ein hierarchisches System macht es leichter, sich zu merken, welcher Gruppe ein neuer Benutzer zugewiesen werden muss. Sie müssen jeden Benutzer nur einer Gruppe zuweisen und verwenden die Gruppenhierarchie zum Festlegen des Zugriffs.

Ihr Zugriffsschema sollte den Zugriff auf möglichst niedere Ebene beschränken, normalerweise auf der Formularebene.

Designer und Administrator



4D bietet Benutzern bestimmte Standardzugriffsrechte und bestimmte Befugnisse in der jeweiligen Umgebung. Wenn ein Kennwortsystem eingerichtet ist, werden diese Standardrechte wirksam.

Der mächtigste Benutzer wird Designer genannt. Der Designer kontrolliert das Layout der Datenbank. Er kann Benutzer und Gruppen erzeugen, Gruppen Zugriffsrechte zuweisen und die Designumgebung verwenden. Dem Designer stehen alle Aspekte der Datenbank offen.

Auf den Designer folgt der Administrator, der normalerweise das Verwalten des Kennwortsystems übernimmt. Wenn der Kennworteditor zum ersten Mal geöffnet wird, erscheinen Designer und Administrator in der Benutzerliste. Zu diesem Zeitpunkt ist der Administrator nur ein normaler Benutzer mit keinen besonderen Zugriffsrechten, insbesondere wenn der Zugriff auf den Designmodus eingeschränkt ist. Um das Kennwortsystem verwenden zu können, muss er spezifische Zugriffsrechte erhalten. Weitere Informationen dazu finden Sie im Abschnitt **Kennwortsystem pflegen**. Der Administrator ist der einzige Benutzer, der Gruppen speichern und öffnen kann. Weitere Informationen dazu finden Sie im Abschnitt **Gruppen laden und sichern**.

Der Zugriff des Administrators auf andere Teile der Datenbank wird durch die Gruppenmitgliedschaft beschränkt – der Administrator muss einer oder mehreren Gruppen angehören, um Zugriffsrechte in der Datenbank zu haben. Der Administrator wird in jede neue Gruppe eingefügt. Sie können aber auch den Namen des Administrators aus jeder Gruppe entfernen.

Im Dialogfenster zum Verwalten der Benutzer sind die Icons von Designer und Administrator jeweils in rot und grün:

- Designer: 
- Administrator: 

Sie können Designer und Administrator umbenennen, jedoch nicht ihre Icons verändern.

Die Icons der vom Designer angelegten Benutzer und Gruppen haben eine andere Farbe als die des Administrators:

- Icons der Gruppen, die vom Designer erstellt wurden, sind rot; Icons der Gruppen des Administrators sind grün.
- Icons der Benutzer, die vom Designer erstellt wurden, sind blau; Icons der Benutzer des Administrators sind grün.

Der Gruppeneigentümer kann den Gruppennamen jederzeit ändern.

Designer und Administrator können jeweils 16.000 Gruppen und 16.000 Benutzer erzeugen.

Kennwortsystem aktivieren

Sie aktivieren das Kennwortsystem von 4D, indem Sie dem Designer ein Kennwort zuweisen.

Solange Sie dem Designer kein Kennwort zuweisen, kann jeder Anwender auf alle Teile der Datenbank zugreifen, selbst wenn Sie Benutzer und Gruppen angelegt haben. Der Grund dafür ist, dass zum Öffnen der Anwendung ist kein Kennwort abgefragt wird.

Weisen Sie dem Designer ein Kennwort zu, werden alle Zugriffsrechte wirksam, die Sie Tabellen, Formularen, Menüs und Methoden zugeordnet haben. Um die Anwendung zu öffnen, müssen Benutzer jetzt ein Kennwort eingeben.

Warnung: Vergessen Sie nicht das Kennwort für den Designer! Denn sonst können Sie die Anwendung nicht mehr in der Designumgebung öffnen.

Wollen Sie das Kennwortsystem deaktivieren, müssen Sie lediglich das Kennwort des Designers entfernen.

Standardbenutzer einrichten

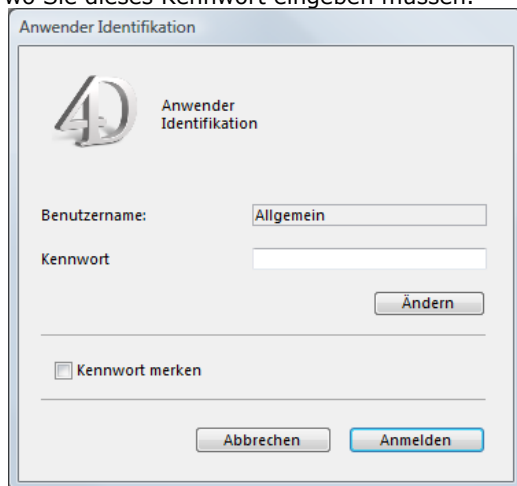
Sie können für Ihre Datenbank einen Standardbenutzer definieren. Ist diese Option aktiv, müssen Benutzer, die die Datenbank öffnen oder sich an diese anmelden, nicht länger einen Namen eingeben. Haben Sie dem Standardbenutzer dagegen kein Kennwort zugeordnet, erscheint der Kennwortdialog nicht und die Datenbank öffnet sich direkt.

Jeder angemeldete Benutzer hat die Zugriffsrechte und -einschränkungen, die für den Standardbenutzer gelten. Diese Option vereinfacht den Zugriff auf die Datenbank und hält gleichzeitig ein komplettes System zum Steuern der Benutzeraktionen aufrecht.

Um einen Standardbenutzer festzulegen, wählen Sie in den Datenbank-Eigenschaften auf der **Seite Sicherheit** in der DropDown-Liste "Standardbenutzer" den passenden Benutzer aus. Natürlich müssen Sie diesen Benutzer zuerst im Editor Benutzer und Gruppen anlegen und entsprechende Zugriffsrechte und -einschränkungen festlegen.

Der Zugriff auf die Datenbank ist nicht länger individuell eingestellt. Wenn Sie die Datenbank öffnen, passiert folgendes:

- Haben Sie dem Standardbenutzer kein Kennwort zugeordnet, erscheint der Kennwortdialog nicht.
- Haben Sie dem Standardbenutzer ein Kennwort zugeordnet, erscheint beim Öffnen der Datenbank ein Dialogfenster, wo Sie dieses Kennwort eingeben müssen.



Ist der Modus „Standardbenutzer“ aktiviert und ein Kennwort erforderlich, empfehlen wir, in den Datenbank-Eigenschaften auf der Seite Zugriff die Option **Die Anwender können ihre Kennwörter ändern** zu deaktivieren.

Kennwortdialog erneut anzeigen

Sie können 4D "zwingen", den Standard-Kennwortdialog anzuzeigen, um sich z.B. als Designer oder Administrator an die Datenbank anzumelden.

Um den Kennwortdialog bei aktivem Modus "Standardbenutzer" wieder anzuzeigen:

1. Öffnen Sie die Datenbank mit gedrückter **Umschalttaste**.
Sie erhalten ein Dialogfenster, wo Sie Name und Kennwort eingeben können.

Benutzer und Gruppen verwalten

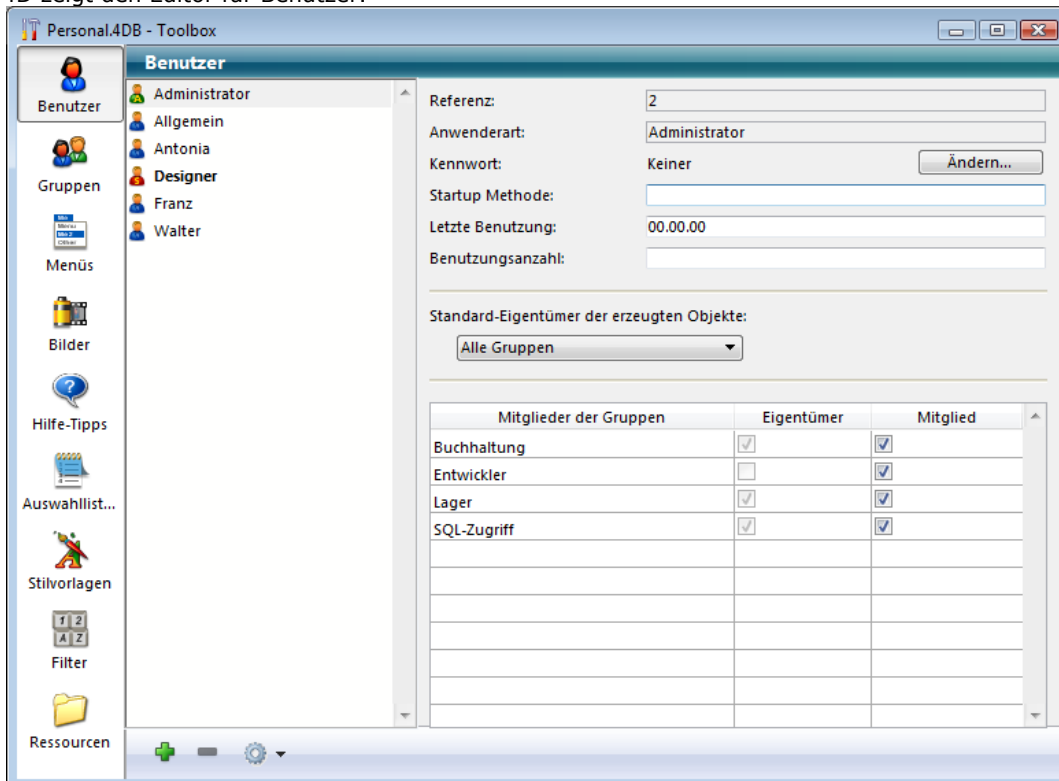
Die Editoren für Gruppen und Benutzer liegen in der Toolbox von 4D. Hier können Sie Benutzer und Gruppen einrichten, den Benutzern Kennwörter zuweisen, Benutzer in Gruppen setzen, etc..

Benutzer hinzufügen und ändern

Auf der Seite Benutzer erstellen Sie Benutzerkonten, legen deren Eigenschaften fest und weisen sie den verschiedenen Gruppen zu, zusätzlich zur Definition, wie sie die Datenbank verwenden können.


Um einen Benutzer hinzuzufügen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox>Anwender** ODER klicken Sie in der Werkzeugleiste auf das Icon **Toolbox**. 4D zeigt den Editor für Benutzer.

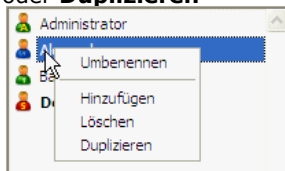


Die Liste der Benutzer zeigt alle Benutzer, die für den aktuellen Benutzer sichtbar sind, das sind:

- Für den Designer alle Benutzer
- Für den Administrator die Benutzer, welche er angelegt hat (grüne Icons).

2. Wählen Sie unter der Benutzerliste das Pluszeichen  ODER

Klicken Sie mit der rechten Maustaste in die Benutzerliste und wählen im Kontextmenü den Eintrag **Hinzufügen** oder **Duplizieren**



Hinweis: Über den Eintrag **Duplizieren** können Sie rasch mehrere Benutzer mit denselben Merkmalen erstellen.

4D fügt in der Liste einen neuen Benutzer hinzu, er hat standardmäßig den Namen *NeuerAnwenderN*. Der rechte Teil zeigt Informationen über die Art des Benutzers:

- Das Feld "Referenz" gibt die Referenznummer des gewählten Benutzers an. Diese Nummer wird von den Befehlen der Programmiersprache verwendet.
- Das Feld "Anwenderart" gibt an, woher das Benutzerprofil stammt. Es gibt folgende Benutzerarten:
 - Designer: Designer Benutzer
 - Administrator: Administrator Benutzer
 - Developer: Vom Designer angelegte Benutzer
 - Benutzer: Vom Administrator angelegte Benutzer

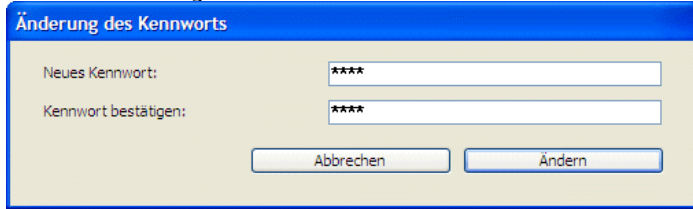
3. Geben Sie einen neuen Benutzernamen ein. Diesen Namen verwendet der Benutzer, um die Datenbank zu öffnen. Um einen Benutzer umzubenennen, klicken Sie

zweimal auf seinen Namen oder öffnen Sie das Kontextmenü, unter Windows mit der Kombination **Alt-Taste+Klick**, auf Mac OS mit **Wahltaste+Klick**, und wählen Sie den Eintrag **Umbenennen**.

4. Klicken Sie auf die Schaltfläche **Ändern**, um das Kennwort des Benutzers einzugeben.

Auf dem Bildschirm erscheint der Eingabedialog:

5. Geben Sie im Eingabebereich ein Kennwort für den Benutzer ein, dann erneut im 2. Bereich *Bestätigen*:



Sie können für ein Kennwort bis zu 15 alphanumerische Zeichen verwenden. Der Kennworteditor unterscheidet Groß- und Kleinschreibung – der Benutzer muss das Kennwort genau so eingeben, wie es hier eingegeben wird. Wenn Sie zum Beispiel ein Benutzerkennwort "BunteKuh" festlegen, muss der Benutzer es mit großem B und großem K eingeben. Andernfalls akzeptiert 4D das Kennwort nicht.


Nach dem Festlegen eines Benutzerkennworts ist es im Dialogfenster nicht mehr sichtbar. Statt der Kennwortzeichen erscheinen Sternchen.

6. Bestätigen Sie das Dialogfenster.
Sind die beiden Einträge unterschiedlich, ertönt ein Beep. Der Dialog bleibt geöffnet.
7. Diese Gruppe besitzt alle Objekte (Formulare, Methoden usw.), die der Benutzer erzeugt. Sie können zum Beispiel festlegen, dass die Gruppe Buchhaltung die Objekte besitzt, die von den einzelnen Benutzern in dieser Gruppe erzeugt werden. Wenn ein Benutzer aus einer anderen Gruppe versucht, ein Formular zu ändern, das von einem Mitglied der Gruppe Buchhaltung erzeugt wurde, erscheint eine Meldung, dass er kein Recht zum Ändern des Formulars besitzt.
8. (Optional) Geben Sie den Namen einer zugewiesenen Methode ein, die ausgeführt wird, wenn der Benutzer die Datenbank öffnet.
Über diese Methode lassen sich z.B. Voreinstellungen des Benutzers laden.
9. Wählen Sie in der Tabelle „Mitglieder der Gruppen“ eine oder mehrere Gruppen aus, zu denen der Benutzer gehört.
 - Um den gewählten Benutzer der Gruppe hinzuzufügen, markieren Sie die Option in der Spalte **Mitglieder**. Wollen Sie ihn wieder entfernen, deaktivieren Sie diese Option.
 - Die Spalte Eigentümer gibt an, ob der gewählte Benutzer der Gruppeneigentümer ist. Diese Spalte lässt sich nicht verändern.

HInweis: Sie können die Zugehörigkeit zu verschiedenen Gruppen auch auf der Seite Gruppe definieren.

Um die Eigenschaften eines vorhandenen Benutzers zu ändern, wählen Sie ihn in der Benutzerliste aus und nehmen die Änderungen vor. Weitere Informationen zu den Eigenschaften finden Sie in den Schritten 3 bis 9.

Benutzer löschen

Um einen Benutzer zu löschen, markieren Sie ihn und wählen unter der Liste das Minuszeichen  oder verwenden Sie den Eintrag **Löschen** im Kontextmenü.

Gelöschte Benutzernamen erscheinen nicht mehr im Benutzereditor. Beim Erstellen neuer Benutzerkonten können Sie die Nummern von gelöschten Benutzern wiederverwenden.

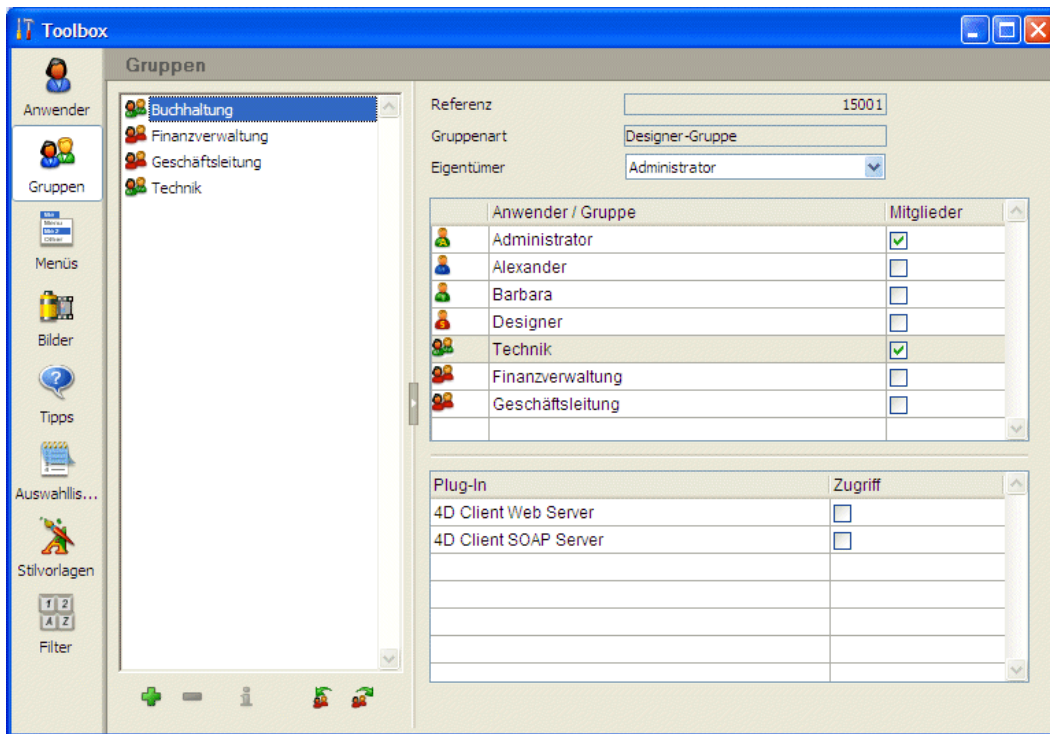
Zugriffsgruppen erstellen und konfigurieren

Auf der Seite Gruppen definieren Sie die Benutzer bzw. andere Gruppen, die eine Gruppe enthalten und definieren den Zugriff auf Plug-Ins. Wenn Sie eine Gruppe erzeugen, können Sie unter den Benutzern einen Gruppeneigentümer bestimmen.

Beachten Sie, dass sich eine einmal angelegte Gruppe nicht mehr löschen lässt. Um eine Gruppe zu deaktivieren, können Sie lediglich alle darin enthaltenen Benutzer entfernen.

Um eine Gruppe zu erstellen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox>Gruppen**
ODER
klicken Sie in der Werkzeugleiste auf das Icon **Toolbox** und wählen dann im Dialogfenster das Icon **Gruppen**.

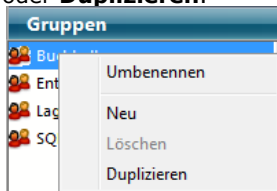


Die Gruppenliste zeigt alle Gruppen der Datenbank, egal welcher Benutzer sie angelegt hat.

2. Klicken Sie unter der Gruppenliste auf das Pluszeichen **+**.

ODER

Klicken Sie mit der rechten Maustaste in die Anwenderliste und wählen im Kontextmenü den Eintrag **Hinzufügen** oder **Duplizieren**.



Hinweis: Über den Befehl Duplizieren können Sie rasch mehrere Gruppen mit denselben Merkmalen erstellen.

4D fügt in der Liste eine neue Gruppe mit Namen *Neue Gruppe1* hinzu.

Der rechte Teil zeigt Informationen über die Gruppe:

- Das Feld "Referenz" gibt die Referenznummer der gewählten Gruppe an. Diese Nummer wird von den Befehlen der Programmiersprache verwendet.
- Das Feld "Gruppentyp" gibt die Herkunft der Gruppe an. Es gibt folgende Gruppen:
 - Designer-Gruppe: vom Designer erstellte Gruppe
 - Administrator-Gruppe: vom Administrator erstellte Gruppe

3. Geben Sie den Namen der neuen Gruppe ein.

Der Gruppenname kann bis zu 15 Zeichen umfassen.

Über das Kontextmenü können Sie eine Gruppe jederzeit umbenennen: Klicken Sie zweimal auf ihren Namen oder öffnen Sie das Kontextmenü, unter Windows mit der Kombination **Alt-Taste+Klick**, auf Mac OS mit

Wahltaste+Klick, und wählen Sie den Eintrag **Umbenennen**. Sie können nur die Gruppen umbenennen, die Sie erstellt haben. Der Designer kann keine Gruppe umbenennen, die der Administrator erstellt hat und umgekehrt.

4. (Optional) Wählen Sie einen Eigentümer im DropDown-Menü „Eigentümer“ aus.

Der Gruppeneigentümer kann in der Gruppe Benutzer hinzufügen und entfernen (siehe nächster Abschnitt).

Beachten Sie, dass der Administrator standardmäßig Eigentümer aller Gruppen ist, die der Designer angelegt hat.

5. Definieren Sie in der Spalte **Mitglieder** die dazugehörigen Mitglieder.
6. (Optional) Weisen Sie den Zugriff auf Plug-Ins zu.

Weitere Informationen zu den beiden letzten Punkten finden Sie in den folgenden Abschnitten.

Gruppeneigentümer festlegen

In der Spalte **Eigentümer** können Sie für jede Gruppe einen Eigentümer bestimmen. Normalerweise ist der Eigentümer der Administrator. Sie können aber auch ein beliebiges Gruppenmitglied als Eigentümer bestimmen.

Der Gruppeneigentümer kann die Möglichkeit erhalten, Benutzer in seiner Gruppe hinzuzufügen oder zu löschen. Die Benutzer, die zur Gruppe hinzugefügt werden, müssen bereits vorhanden sein. Gruppeneigentümer können weder Benutzer noch Benutzereigenschaften wie Kennwörter erzeugen bzw. ändern, noch andere Gruppen hinzufügen oder entfernen.

Analog zum Administrator muss auch der Gruppeneigentümer die Möglichkeit erhalten, den Kennworteditor aufzurufen, der normalerweise nur dem Designer der Datenbank offensteht. Weitere Informationen dazu finden Sie im Abschnitt **Kennwortsystem pflegen**.

Benutzer oder Gruppen in Gruppen setzen

Sie können jeden Benutzer oder jede Gruppe in eine Gruppe setzen und die Gruppe selbst in mehrere andere Gruppe setzen. Sie müssen einen Benutzer nicht zwingend in eine Gruppe legen.

Der Designer kann den Inhalt jeder Gruppe in der Datenbank verändern. Administrator und Gruppeneigentümer können dagegen nur den Inhalt der Gruppen verändern, deren Eigentümer sie sind.

Unabhängig von Ihrem Benutzerstatus können Sie in Ihren eigenen Gruppen jeden Benutzer bzw. jede Gruppe sichtbar machen, hinzufügen oder entfernen.

Um einer Gruppe einen Benutzer bzw. eine Gruppe hinzuzufügen, markieren Sie in der Spalte **Mitglieder** die Option für den jeweiligen Benutzer bzw. die Gruppe:

Anwender/Gruppe	Mitglieder
Administrator	<input checked="" type="checkbox"/>
Allgemein	<input type="checkbox"/>
Antonia	<input checked="" type="checkbox"/>
Designer	<input type="checkbox"/>
Franz	<input checked="" type="checkbox"/>
Entwickler	<input type="checkbox"/>
SQL-Zugriff	<input type="checkbox"/>
Lager	<input type="checkbox"/>

Kreuzen Sie einen Benutzernamen an, wird dieser Benutzer der Gruppe hinzugefügt. Kreuzen Sie einen Gruppennamen an, werden alle Benutzer der Gruppe der neuen Gruppe hinzugefügt. Der Benutzer oder die zugewiesene Gruppe haben dann dieselben Zugriffsrechte, die für die neue Gruppe definiert wurden.

Um eine Benutzerhierarchie zu erzeugen, verschachteln Sie Gruppen, d.h. Sie setzen eine Gruppe in eine andere. Die Benutzer einer verschachtelten Gruppe erhalten die Rechte beider Gruppen. Setzen Sie zum Beispiel die Gruppe Geschäftsleitung in die Gruppe Buchhaltung, erhalten Benutzer der Gruppe Geschäftsleitung automatisch die Rechte der Buchhaltung zusätzlich zu den Rechten der Geschäftsleitung. Hingegen haben Benutzer innerhalb der Gruppe Buchhaltung keinen Zugriff auf die Rechte der Gruppe Geschäftsleitung. Weitere Informationen dazu finden Sie im Abschnitt **Überblick über das Zugriffssystem**.

Um einen Benutzer oder eine Gruppe aus einer anderen Gruppe zu entfernen, deaktivieren Sie die jeweilige Option in der Spalte **Mitglieder**.

Gruppe einem Plug-In oder einem 4D Client Web Server zuweisen

Sie können einer Gruppe Rechte auf jedes Plug-In zuweisen, das in der Datenbank installiert ist. Dies schließt alle 4D Plug-Ins und Plug-Ins von Drittanbietern ein. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter der Funktion **Set user properties**.

Auf diese Weise können Sie die Verwendung der verfügbaren Lizenzen für Plug-Ins steuern. Benutzer, die nicht der Gruppe mit Zugriffsrecht auf das Plug-In angehören, können dieses Plug-In nicht laden. Hier können Sie auch die Verwendung von **4D Client Web Server** und **SOAP Server** einschränken.

Der Bereich Plug-In auf der Seite Gruppen zeigt alle Plug-Ins an, die in der 4D Anwendung geladen sind. Damit eine Gruppe Zugriff auf ein Plug-In erhält, müssen Sie nur das gewünschte Plug-In markieren:

Plug-in	Zugriff
4D Client Web Server	<input checked="" type="checkbox"/>
4D Client SOAP Server	<input type="checkbox"/>

Über die Einträge **4D Client Web Server** und **4D Client SOAP Server** können Sie für jedes 4D im remote Modus die Veröffentlichung von Web und SOAP (Web Services) steuern. 4D Server behandelt diese Lizenzen wie Lizenzen für Plug-Ins. So können Sie, wie für Plug-Ins, den Zugriff auf Lizenzen des Web Server auf eine bestimmte Benutzergruppe beschränken.

Gruppen laden und sichern

4D ermöglicht dem Administrator, Gruppen und Benutzer, die er erzeugt oder geändert hat, zu speichern und zu öffnen. Wenn Gruppen gespeichert werden, wird alles über die aktuellen Benutzer und Gruppen gespeichert.

Diese Möglichkeit bedeutet, dass der Administrator das Zugriffssystem einer Datenbank speichern und auf eine geänderte Version der gleichen Datenbank oder auf eine neue Datenbank übertragen kann. Da die Gruppen wieder geladen werden können, müssen die Benutzer der Datenbank kein neues Zugriffssystem erlernen.

Alle Benutzernamen, Kennwörter, StartUp-Methodennamen, Gruppen, Gruppeneigentümer und Gruppenzugehörigkeiten bleiben erhalten.

Hinweis: Der Designer kann über den Gruppeneditor Gruppen weder speichern noch öffnen. Er kann jedoch die 4D Befehle **USERS TO BLOB** und **BLOB TO USERS** ausführen.

Um Gruppen zu sichern, die vom Administrator erstellt oder geändert wurden:

1. Öffnen Sie die Datenbank als Administrator und zeigen Sie in der Toolbox die Seite **Gruppen** an.
2. Klicken Sie unter der Gruppenliste auf das Menü **Optionen** und wählen den Eintrag **Anwender und Gruppen speichern**.
4D zeigt ein Dialogfenster zum Erzeugen von Dateien, so dass Sie die Gruppen benennen und sichern können. Gruppen- und Benutzerdateien haben die Endung ".4UG".

Um Gruppen zu laden:

1. Öffnen Sie die Datenbank als Administrator und zeigen in der Toolbox die Seite **Gruppen** an.
2. Klicken Sie unter der Gruppenliste auf das Menü **Optionen** und wählen den Eintrag **Anwender und Gruppen laden**.
4D zeigt ein Dialogfenster zum Öffnen von Dateien, in der Sie die Gruppenseite auswählen können.

Benutzern Zugriff auf die Designumgebung geben

Alle Benutzer haben Zugriff auf den Anwendungsmodus. Den Zugriff auf die Designumgebung können Sie dagegen einschränken. Dazu wählen Sie in den Datenbank-Eigenschaften auf der [Seite Sicherheit](#) die betreffende Gruppe in der Dropdown-Liste **Design-Zugriff** aus.

Dann können nur Benutzer, die zu dieser Gruppe gehören und der Designer die Struktur der Datenbank verändern. Der Designer hat immer Zugriff auf die Designumgebung, auch wenn er nicht explizit zur Gruppe mit Designzugriff gehört.

Alle anderen Benutzer sind normale Benutzer. Wenn ein Benutzer eine Anwendung öffnet, wird diese in der Anwendungsumgebung geöffnet. Der Zugriff eines Benutzers wird durch seine Gruppenzugehörigkeit eingeschränkt.

Datenbankobjekten eine Gruppe zuweisen

Haben Sie Benutzer und Zugriffsgruppen festgelegt, können Sie folgenden Objekten Gruppen zuweisen:

- Formulare,
- Methoden,
- Menübefehle

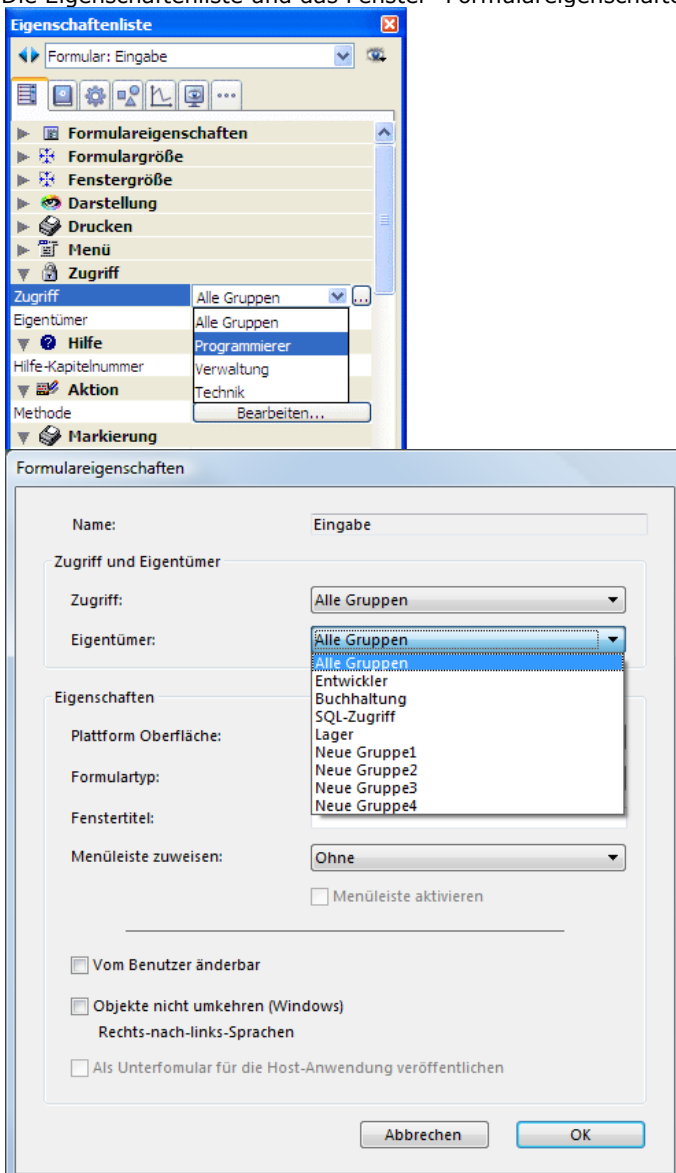
Sie können jedem Objekt nur eine Gruppe zuweisen. Deshalb ist es wichtig, die Zugriffsgruppen so zu gestalten, dass Benutzer mit mehr Rechten in der Hierarchie zu allen Gruppen mit weniger Rechten gehören. Weitere Informationen dazu finden Sie im Abschnitt "Ein Schema für Zugriffshierarchie" unter **Überblick über das Zugriffssystem**.

Gruppe einem Formular zuweisen

Wenn Sie einer Gruppe Zugriffsrechte auf ein Formular zuweisen, können nur Benutzer dieser Gruppe dieses Formular zur Dateneingabe verwenden. Wenn Sie einer Gruppe Eigentümerrechte auf ein Formular zuweisen, können nur Benutzer dieser Gruppe dieses Formular in der Designumgebung ändern.

Um Zugriffs- und Eigentümerrechte zuzuweisen:

1. Öffnen Sie das Formular im Formulareditor und zeigen seine Eigenschaften in der Eigenschaftensliste an ODER
Wählen Sie das Formular im Explorer auf der Seite **Formulare** aus und wählen dann im DropDown-Menü **Optionen** am unteren Rand oder im Kontextmenü den Eintrag **Formulareigenschaften**.
Die Eigenschaftensliste und das Fenster "Formulareigenschaften" zeigen die Listen für Zugriff und Eigentümer:



2. Wählen Sie in der Dropdown-Liste „Zugriff“ eine Gruppe aus.
Wenn Sie keine Gruppe auswählen, erhalten alle Gruppen Zugriffsrechte auf das Formular (Standardeinstellung).

3. Wählen Sie in der Dropdown-Liste „Eigentümer“ eine Gruppe aus.
Wenn Sie keine Gruppe auswählen, erhalten alle Gruppen Eigentümerrechte auf das Formular (Standardeinstellung).

Gruppe einer Projektmethode zuweisen

Wenn Sie einer Gruppe Zugriff auf eine Projektmethode zuweisen, können nur Benutzer dieser Gruppe diese Methode verwenden. Wenn Sie eine Gruppe als Eigentümer für eine Projektmethode zuweisen, können nur Benutzer dieser Gruppe diese Methode in der Designumgebung ändern.

Um Zugriffs- und Eigentümerrechte zuzuweisen:

1. Öffnen Sie die Methode im Methodeneditor und wählen im Menü **Methode** den Befehl **Methodeneigenschaften** ODER
Wählen Sie im Explorer auf der **Seite Methoden** die Projektmethode und wählen dann im DropDown-Menü **Optionen** am unteren Rand oder im Kontextmenü den Eintrag **Methode-Eigenschaften**.
Das Fenster "Methode-Eigenschaften" zeigt die Listen für Zugriff und Eigentümer (siehe **Eigenschaften für Projektmethoden**).
2. Wählen Sie in der Dropdown-Liste „Zugriffsrechte“ eine Gruppe aus.
Wenn Sie keine Gruppe auswählen, erhalten alle Gruppen Zugriffsrechte auf die Methode (Standardeinstellung).
3. Wählen Sie in der Dropdown-Liste „Eigentümer“ eine Gruppe aus.
Wenn Sie keine Gruppe auswählen, erhalten alle Gruppen Eigentümerrechte auf die Methode (Standardeinstellung).

Gruppe einem Menübefehl zuweisen

Sie können einem Menübefehl eine Gruppe zuweisen, so dass nur Benutzer dieser Gruppe den Menübefehl in der Anwendungsumgebung verwenden können.

Um einem Menübefehl eine Gruppe zuzuweisen:

1. Wählen Sie im Menü **Design** den Befehl **Tool Box>Menüs** ODER
Klicken Sie in der Werkzeugleiste auf das Icon **Toolbox** und dann auf das Icon **Menüs**.
Auf dem Bildschirm erscheint der Menüeditor. Weitere Informationen dazu finden Sie im Kapitel **Menüs und Menüleisten**.
2. Wählen Sie eine Menüleiste.
Der Bereich Menüleiste zeigt die Menüs, die zu dieser Menüleiste gehören.
3. Erweitern Sie ein Menü in der Liste der Menüleisten.
Die Menübefehle und Methoden für dieses Menü erscheinen.
4. Markieren Sie den Menübefehl, für den Sie eine Zugriffgruppe festlegen wollen.
5. Wählen Sie in der Dropdown-Liste „Zugriffsrechte“ die gewünschte Gruppe aus.

Kennwortsystem pflegen

Ist ein Kennwortsystem eingerichtet, muss es regelmäßig gepflegt werden. Es müssen Benutzer hinzugefügt werden, Gruppen benötigen neue Mitglieder und Kennwörter müssen geändert werden. Der Designer hat Zugriff auf die Designumgebung und kann erforderliche Änderungen in der Toolbox auf den Seiten Benutzer und Gruppen durchführen.

Administrator und Designer können auch die Benutzungshistorie jedes Benutzers für die Pflege ansehen.

Hinweis: Sie können zur Wartung und zur leichteren Aktualisierung von Datenbanken Benutzer und Gruppen sichern und laden. Weitere Informationen dazu finden Sie im Abschnitt **Benutzer und Gruppen verwalten**.

Zugriff für Wartung

Der Administrator hat nicht zwangsläufig Zugriff auf die Designumgebung (siehe **Benutzern Zugriff auf die Designumgebung geben**). Erstellt der Designer dagegen eine Projektmethode mit dem 4D Befehl **EDIT ACCESS**, können Administrator und Gruppeneigentümer eingeschränkte Rechte zur Kontrolle von Benutzern und Gruppen haben. Der 4D Befehl **EDIT ACCESS** lässt sich in eine Methode einfügen, die einem eigenen Menü beigefügt ist, oder lässt sich durch Auswählen des Eintrags **Methode** im Menü **Start** ausführen. Wird die Methode von einem Benutzer ausgeführt, der weder Administrator noch Gruppeneigentümer ist, hat sie keine Auswirkung.

Wird die Methode ausgeführt, hängt das Ergebnis davon ab, ob der Benutzer Administrator oder Gruppeneigentümer ist.

- Führt der Administrator die Methode aus, die den 4D Befehl **EDIT ACCESS** enthält, zeigt 4D die Toolbox nur mit den Seiten Benutzer und Gruppen. Über diese Editoren kann der Administrator Benutzer und Gruppen anlegen; selbst angelegte Benutzer oder Gruppen bearbeiten, inkl. Ändern der Benutzerkennwörter; und Benutzer aus selbst angelegten Gruppen entfernen oder hinzufügen. Der Administrator kann nicht Gruppen Formularen, Menübefehlen, Methoden oder Plug-Ins zuweisen. Nur der Designer kann diesen Objekten Gruppen zuweisen.
- Führt ein Gruppeneigentümer, der nicht Administrator ist, die Methode aus, die den 4D Befehl **EDIT ACCESS** enthält, zeigt 4D in der Toolbox nur die Gruppen, die dem Gruppeneigentümer gehören. Der Gruppeneigentümer kann Benutzer in diesen Gruppen hinzufügen oder entfernen. Er kann jedoch keine Benutzer definieren, Benutzerinformationen bearbeiten oder Gruppen hinzufügen. Die Menübefehle zum Hinzufügen und Bearbeiten von Benutzern erscheinen in Grauschrift.

Benutzung anzeigen

Die Seite Benutzer in der Toolbox zeigt das Datum der letzten Benutzung der Datenbank durch den Benutzer und die Gesamtanzahl der Benutzungen. Administrator oder Designer können diese Information ansehen, wenn Sie einen Benutzer in der Liste auswählen.

Um die Benutzerinformation anzusehen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox>Anwender** oder klicken Sie in der Werkzeugleiste auf das Icon **Toolbox**
ODER
Führen Sie die Methode aus, die den 4D Befehl **EDIT ACCESS** enthält.
4D öffnet die Toolbox auf der Seite Benutzer.
2. Markieren Sie in der Benutzerliste den gewünschten Benutzernamen.
Das Dialogfenster zeigt das Datum, wann der Benutzer die Datenbank zuletzt benutzt und wie viele Male er sie geöffnet hat.

Letzte Benutzung:	Fr, 24. Sep 2010
Benutzungsanzahl:	1

Kennwort durch Benutzer ändern

Das Dialogfenster Anwender Identifikation enthält standardmäßig die Schaltfläche **Ändern**, über die der aktuelle Benutzer sein eigenes Kennwort ändern kann. Wird die Liste der Benutzer angezeigt, ist der markierte Benutzer der aktuelle Benutzer. Wird die Liste nicht angezeigt, muss im Dialogfenster Anwender Identifikation zuerst der Name des aktuellen

Benutzers eingegeben werden:

Anwender Identifikation

Anwender Identifikation

Anwenderliste

- Administrator
- Antonia
- Designer
- Franz
- Walter

Kennwort

Kennwort merken

Klickt der Benutzer auf die Schaltfläche **Ändern**, erscheint folgendes Dialogfenster:

Änderung des Kennworts

Anwender Identifikation

Ändere Kennwort von Anwender Antonia:

Altes Kennwort:










Neues Kennwort:

Kennwort bestätigen:

Um das Kennwort zu ändern, muss der Benutzer zuerst das alte Kennwort eingeben. Dann gibt er das neue Kennwort einmal ein und noch einmal zum Bestätigen. Klickt er auf die Schaltfläche **OK**, wird bei korrekter Eingabe das alte Kennwort durch das neue ersetzt und in der Datenbank gespeichert. Der Benutzer muss sich dann mit dem neuen Kennwort anmelden, um die Datenbank zu öffnen.

Sie können die Schaltfläche **Ändern** ausblenden, so dass Benutzer ihre Kennwörter nicht ändern können. Diese Option finden Sie in den Datenbank-Eigenschaften auf der [Seite Sicherheit](#).

Menüs und Menüleisten

-  Oberfläche mit Menüs gestalten
-  Menüeditor
-  Menüs erstellen
-  Menüs anfügen
-  Referenzen für Bezeichnung verwenden
-  Aktion für Menübefehle festlegen
-  Menüeigenschaften einrichten
-  Menüleisten verwalten
-  SDI Modus unter Windows

Oberfläche mit Menüs gestalten

Sie können für Ihre Datenbanken und eigenen Lösungen eigene Menüs erstellen. Jede Desktop-Anwendung bietet Dropdown-Menüs als Standard an. Diese Menüs erleichtern die Handhabung Ihrer Anwendung, die Benutzer können sich rascher eingewöhnen. Sie können eigene Menüs und eigene Werkzeugleisten erstellen, so dass Ihre Datenbanken wie eigenständige Anwendungen arbeiten.

Eine eigene Anwendung muss wenigstens eine Menüleiste mit mindestens einem Menü enthalten. Beim Erstellen einer neuen Datenbank legt 4D standardmäßig eine eigene Menüleiste an, so dass Sie auf die Anwendungsumgebung zugreifen können. Weitere Informationen über das Erstellen eigener Anwendungen finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Eine 4D Anwendung aufbauen**.

Überblick

Im allgemeinen enthalten Menüs die Menübefehle, die der Benutzer wählt, um Aufgaben der Datenbank durchzuführen: Datensätze ändern, Datensätze suchen, Berichte drucken usw.. Die folgende Abbildung zeigt dafür ein Beispiel:



Eine Menüleiste ist eine Gruppe von Menüs, die Sie zusammen auf dem Bildschirm darstellen. Jedes Menü einer Menüleiste kann viele Menübefehle inkl. Untermenüs enthalten, sowie Trennlinien, welche die Menübefehle in Gruppen aufteilen, und entsprechende Kürzel. Wählt der Benutzer einen Befehl im Menü oder Untermenü, ruft dieser eine Projektmethode auf, die eine Operation ausführt.

Sie können für jede Datenbank mehrere unterschiedliche Menüleisten anlegen. Sie verwenden z.B. eine Menüleiste mit Menüs für normale Datenbank-Operationen und eine andere, die nur bei Berichten aufgerufen wird. Eine Menüleiste kann ein Menü mit Befehlen zur Eingabe von Datensätzen enthalten. Die Menüleiste, die zusammen mit dem Eingabeformular erscheint, kann das gleiche Menü enthalten, die Menübefehle sind jedoch nicht anwendbar, weil der Benutzer diese für die Dateneingabe nicht benötigt.

Sie verwenden den Menüeditor auch, um eigene Werkzeugleisten zu erstellen. Hierzu verbinden Sie ein Icon mit einem Menübefehl. Das Icon erscheint in der Symbolleiste, der Text des Menübefehls erscheint als Tipp zum Icon.

Sie können auch das gleiche Menü in mehreren Menüleisten oder anderen Menüs verwenden, oder das Menü unverknüpft lassen und nur per Programmierung steuern. Das ist dann ein unabhängiges Menü.

Beachten Sie beim Entwerfen von Menüs folgende Regeln:

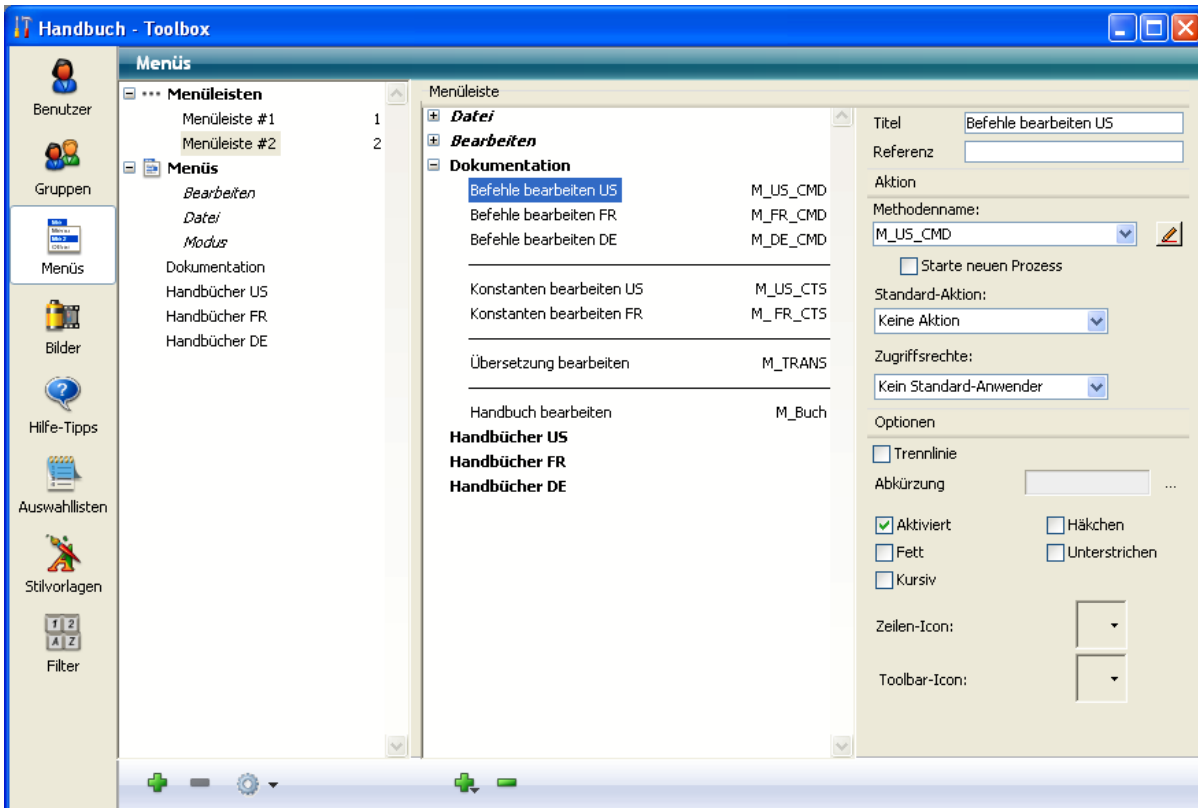
- Verwenden Sie Menüs für Funktionen, die zu Menüs passen: Menübefehle sollen Aufgaben durchführen, wie z.B. einen Datensatz hinzufügen, Datensätze suchen, oder einen Bericht drucken.
- Fassen Sie Menübefehle mit gleichen Funktionen in Gruppen zusammen: Alle Menübefehle z.B., die in Datensätzen blättern, sollten im gleichen Menü liegen; oder Sie fassen alle Operationen für eine bestimmte Tabelle in einem Menü zusammen.

Zum Erstellen von Menüs und Menüleisten verwenden Sie:

- den **Menüeditor**
- Befehle der 4D Programmiersprache
- oder eine Kombination aus beiden

Sie können Menüleisten und Menüs auch komplett per Programmierung anlegen, ohne den Menüeditor zu verwenden. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Menüs**.

Sie wählen den Menüeditor in der Toolbox mit der Schaltfläche **Menüs** aus:



Menüs und Menüleisten erscheinen im linken Bereich des Dialogfensters als zwei Einträge in derselben hierarchischen Liste. So können Sie alle in der Datenbank definierten Menüs auf einmal sehen und müssen nicht zuerst eine Menüleiste auswählen. Die Menüs erscheinen in alphabetischer Reihenfolge.

Sie können jedes Menü an eine Menüleiste oder ein anderes Menü anfügen. Im zweiten Fall wird das Menü zum Untermenü (siehe **Menü an anderes Menü anfügen (Untermenü)**).

Wollen Sie den Inhalt einer Menüleiste oder eines Menüs ansehen, klicken Sie auf den jeweiligen Titel im linken Bereich des Editors. Die Liste der Einträge in der Menüleiste bzw. im Menü erscheinen dann im mittleren Bereich. Um die Eigenschaften eines Menüeintrags anzuzeigen, wählen Sie diesen im mittleren Teil des Fensters aus.

Ein Häkchen markiert das Element, d.h. Menüleiste oder Menü, zu dem das gewählte Menü gehört. Ist ein Menü mehreren Einträgen zugeordnet, erscheinen mehrere Häkchen. Im folgenden Beispiel ist das Menü "Handbücher US" der Menüleiste #2 zugeordnet:



Wird das Menü nicht verwendet (unabhängiges Menü), erscheint kein Häkchen.

Kontextmenü und Menüassistent

Der Menüeditor enthält ein Kontextmenü, über das Sie direkt auf die Operationen zugreifen können, die für den gewählten Eintrag des Menüs bzw. der Menüleiste möglich sind. Über das Kontextmenü können Sie einen Eintrag hinzufügen oder entfernen, die Liste auf- oder zuklappen, sowie jeweils spezifische Aktionen ausführen.

Der Menüeditor enthält am unteren Rand einen Menüassistenten. Er öffnet sich als PopUp-Menü, wenn Sie auf die Schaltfläche unter der linken Liste klicken. Es enthält sowohl allgemeine als auch kontextabhängige Befehle. Die allgemeinen Befehle ermöglichen, eine neue Menüleiste bzw. ein ein neues Menü und das Menü **Bearbeiten** mit den Standardaktionen zu erstellen. Die kontextabhängigen Befehle sind je nach gewähltem Eintrag unterschiedlich und bieten entsprechende Aktionen für die Verwaltung.

Menüs erstellen

Standardmenüleisten

4D erstellt in jeder neuen Datenbank automatisch eine Standardmenüleiste mit der Nummer 1. Sie enthält Standardmenüs und einen Befehl, um in den Designmodus zurückzukehren. So kann der Entwickler schon beim Anlegen der Datenbank auf die Anwendungsumgebung zugreifen. Die Menüleiste #1 wird automatisch aufgerufen, wenn im Menü **Start>Anwendung testen** gewählt wird.

Die Standardmenüleiste enthält die Menüs **Datei/Ablage**, **Bearbeiten** und **Modus**.

- **Datei/Ablage:** Dieses Menü enthält nur den Befehl **Beenden**, dem die Aktion *Automatisch Beenden* zugeordnet ist. Damit können Sie die Anwendung beenden..
- **Bearbeiten** (Standard): Dieses Menü enthält die Standardaktionen Kopieren, Einsetzen, etc. Es lässt sich vollständig verändern.
- **Modus:** Dieses Menü enthält standardmäßig den Befehl **Zurück zur Designumgebung**. Damit können Sie den Anwendungsmodus wieder verlassen.

Hinweis: Menüeinträge erscheinen in Kursivschrift, da dies Referenzen sind und nicht hard-codierter Text. Weitere Informationen dazu finden Sie im Abschnitt **Referenzen für Bezeichnung verwenden**.


Sie können diese Menüleiste beliebig verändern, sowie weitere hinzufügen.

Menüleiste erstellen

Dieser Abschnitt beschreibt das Erstellen einer eigenen Menüleiste.

Hinweis: Sie können Menüs komplett per Programmierung anlegen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Menüs**.

Um eine Menüleiste anzulegen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox>Menüs**.
4D zeigt den Menüeditor. Standardmäßig erscheint Menüleiste # 1 in der linken Spalte, sowie andere Menüleisten, die bereits angelegt wurden.
4D vergibt die Nummern der Menüleisten der Reihe nach - zuerst erscheint Menüleiste # 1. Sie können die Namen der Menüleisten verändern, jedoch nicht die Nummern, da sie von der Programmiersprache verwendet werden.
2. Klicken Sie auf das Pluszeichen  unter der Liste Menüs
ODER
Wählen Sie im Kontextmenü oder im DropDown-Menü unter der Liste den Eintrag **Erstelle neue Menüleiste**.
In der Liste erscheint eine neue Menüleiste mit den Standardmenüs Datei/Ablage und Bearbeiten.
3. Klicken Sie unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahltaste** auf den Namen der Menüleiste, damit er eingebbar wird und geben Sie einen eigenen Namen ein (optional)
ODER
Geben Sie in der rechten Spalte im Bereich Titel einen eigenen Namen ein.
Eigene Namen vereinfachen die Identifizierung von Menüleisten in den verschiedenen Dialogfenstern des Designmodus und in den Befehlen der Programmiersprache. Der Name muss einmalig sein, er kann bis zu 31 Zeichen lang sein.
Jetzt können Sie Anpassungen vornehmen: Sie können die Menüleisten umbenennen oder Menüs zuweisen, den Menüs Befehle hinzufügen, etc.


Menüs erstellen

Sie können Menüs jederzeit erstellen. Sie können einer Menüleiste oder anderen Menüs als Untermenüs zugeordnet sein oder nicht zugeordnet und nur per Programmierung zu steuern sein (unabhängige Menüs).

Sie müssen Menüs nicht in ihrer endgültigen Reihenfolge erstellen. Sie können sie nach dem Erstellen per Drag&Drop umsortieren. Weitere Informationen dazu finden Sie im Abschnitt **Menüs und Menübefehle neu ordnen**.

Hinweis: Sie können Menüs komplett per Programmierung anlegen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Menüs**.

Um ein Menü anzulegen:


1. Öffnen Sie die Seite Menüs der 4D Toolbox.
2. Wählen Sie in der Liste der Ausgangsmenüs den Titel „Menüs“ oder ein vorhandenes Menü und klicken Sie unter der Liste auf das Pluszeichen 
ODER
Wählen Sie im Kontextmenü oder im DropDown-Menü unter der Liste den Eintrag **Neues Menü anlegen**.
4D fügt am Ende der Liste ein neues Menü hinzu.
3. Klicken Sie unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahltaste** auf den Menünamen, damit er eingebbar wird und geben Sie einen eigenen Namen ein
ODER
Geben Sie in der rechten Spalte im Bereich Titel einen eigenen Namen ein.
Sie können den Menünamen hard-codiert oder als Referenz für eine Variable, eine Ressource oder ein XLIFF Element eingeben. Weitere Informationen dazu finden Sie im Abschnitt **Referenzen für Bezeichnung verwenden**.
Vermeiden Sie bei Eingabe eines Namens Steuerzeichen, welche die Anzeige des Menüs stören könnten (siehe Absatz unten).

4. Wiederholen Sie die Schritte 2 und 3, um weitere Menüs hinzuzufügen (optional).
Sie können jederzeit eine Vorschau des Menüs sehen, das Sie gerade anlegen. Wählen Sie dazu das Quellmenü und klicken dann in den Vorschaubereich im rechten Teil des Editors.

Menübefehle hinzufügen

Für jedes Menü in der Menüleiste erstellen Sie eigene Menübefehle. Diese erscheinen, wenn Sie die Menüs öffnen. Sie können den Einträgen Methoden oder Standardaktionen zuweisen oder andere Menüs hinzufügen (Untermenüs). Weitere Informationen zum Hinzufügen hierarchischer Untermenüs finden Sie im Abschnitt [Menüs anfügen](#).

Um einen Menübefehl hinzuzufügen:

1. Wählen Sie in der Liste der Ausgangsmenüs das Menü, dem Sie einen Menübefehl anfügen wollen.
Hat das Menü bereits Befehle, erscheinen sie in der mittleren Liste. Um einen neuen Befehl hinzuzufügen, markieren Sie den Befehl, über dem er erscheinen soll. Sie können das Menü später noch umsortieren.
2. Wählen Sie im Kontextmenü (rechter Mausclick in die mittlere Liste) oder im DropDown-Menü unter der linken Liste den Eintrag **Neue Zeile für Menü „Menü #“**
ODER
Klicken Sie unter der mittleren Liste auf das Pluszeichen .
4D fügt eine neue Zeile mit dem Standardnamen „Zeile X“ hinzu, wobei X, die Nummer, sich nach den bereits angelegten Befehlen richtet.
3. Klicken Sie unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahltaste** auf den Befehlsnamen, oder doppelklicken Sie auf den Befehlsnamen, damit er eingebbar wird und geben einen eigenen Namen ein.
ODER
Schreiben Sie im rechten Bereich den eigenen Namen in den Bereich „Titel“.
Sie können den Befehlsnamen hard-codiert oder als Referenz für eine Variable, ein Ressource oder ein XLIFF Element eingeben. Weitere Informationen dazu finden Sie im Abschnitt [Referenzen für Bezeichnung verwenden](#).
Vermeiden Sie bei Eingabe eines Namens Steuerzeichen, welche die Anzeige des Menüs stören könnten (siehe nächster Absatz).
4. (Optional) Geben Sie im rechten Bereich eine eigene Referenz im Bereich „Referenz“ ein.
Die Befehle der Programmiersprache können diese Referenz nutzen. Weitere Informationen dazu finden Sie im Abschnitt [Eigene Parameter](#).
5. Wiederholen Sie die Schritte 1 bis 4, um weitere Menübefehle einzufügen.

Steuerzeichen in Menübezeichnungen

Sie können die Eigenschaften der Menüzeilen über Steuerzeichen (Meta-Zeichen) direkt in den Bezeichnungen der Menübefehle setzen.

Sie können zum Beispiel unter Windows einer Menüzeile die Tastaturkürzel **Strg+G**, auf Mac OS **Befehlstaste+G** zuordnen, indem Sie in ihrer Bezeichnung die Zeichen /G einfügen. Steuerzeichen erscheinen nicht in den Bezeichnungen der Menübefehle. Verwenden Sie diese Zeichen nicht bei der Namensvergabe, damit keine unerwünschten Nebenwirkungen eintreten. Es gibt folgende Zeichen:

- (= sich öffnende Klammer, für Trennungslinie
- < kleiner als, für Schriftstil
- ! Ausrufezeichen, für Markierung vor der Zeile (nur Macintosh)
- ^ Circumflex Zeichen, für Icon
- / Schrägstrich, für Tastaturkürzel

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter dem Befehl [APPEND MENU ITEM](#).


Menüs und Menübefehle neu ordnen

Nachdem Sie die Menüs für eine Menüleiste und die Menübefehle für ein Menü erstellt haben, können Sie diese per Drag&Drop neu ordnen. Um einen Menübefehl an anderer Stelle in der Reihenfolge einzufügen, ziehen Sie den Menübefehl bei gedrückter Maustaste an die neue Stelle. Um ein Menü zu bewegen, ziehen Sie dieses Menü bei gedrückter Maustaste an die neue Stelle in der aktuellen Liste der Menüs.

Menüs und Untermenüs entfernen

Sie können jederzeit eine Menüleiste, ein Menü oder einen Menübefehl wieder entfernen. Beachten Sie, dass jedes Menü bzw. jede Menüleiste nur eine Referenz hat. Ist ein Menü mit verschiedenen Menüleisten oder mit Menüs als Untermenü verknüpft (siehe [Menüs anfügen](#)), wirkt sich Ändern oder Löschen sofort auf alle Vorkommen dieses Menüs aus.

Um eine Menüleiste, ein Menü oder einen Menübefehl zu löschen:

- Wählen Sie den entsprechenden Eintrag und klicken Sie am unteren Rand der jeweiligen Liste auf das Minuszeichen .
- Wählen Sie im Kontextmenü oder im DropDown-Menü unter der 1. Liste den Eintrag **Lösche Menüleiste ""**, **Lösche Menü ""** oder **Lösche Zeile ""**.

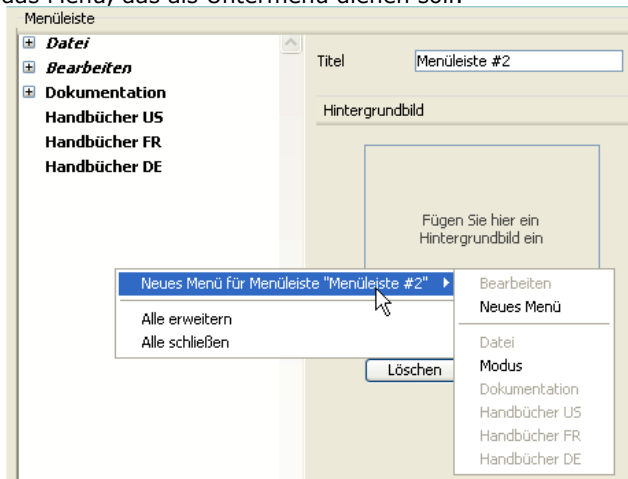
Hinweis: Menüleiste #1 lässt sich nicht löschen.

Menüs anfügen

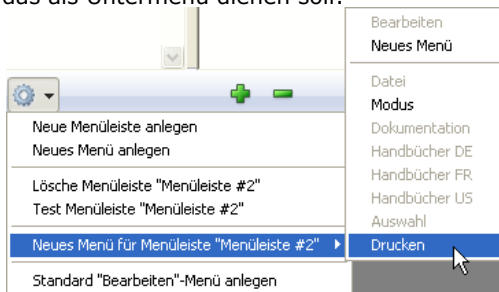
Menü an Menüleiste anfügen

Haben Sie ein Menü angelegt, können Sie es einer Menüleiste zuweisen. Sie können es per Drag-and-Drop, über das Drop-Down Menü am unteren Rand oder über das Kontextmenü des mittleren Bereichs zuweisen.

- **Drag-and-Drop:** Klicken Sie auf eine Menüleiste, um ihren Inhalt im mittleren Bereich anzuzeigen; wählen Sie ein Menü im linken Bereich aus und ziehen es an die gewünschte Stelle im mittleren Bereich.
- **Kontextmenü:** Klicken Sie auf eine Menüleiste, um ihren Inhalt im mittleren Bereich anzuzeigen; klicken Sie mit der rechten Maustaste in diesen Bereich, markieren den Eintrag **Neues Menü für Menüleiste „Name“** und wählen das Menü, das als Untermenü dienen soll:

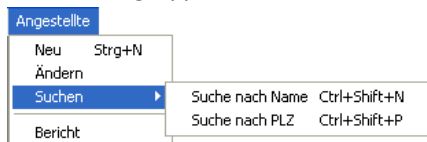


- **DropDown-Menü „Assistent“:** Wählen Sie eine Menüleiste im linken Bereich und klicken unter der Liste auf das DropDown-Menü „Assistent“; markieren Sie den Eintrag **Neues Menü für Menüleiste „Name“** und dann das Menü, das als Untermenü dienen soll:



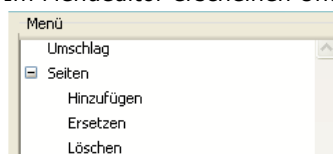
Menü an anderes Menü anfügen (Untermenüs)

Sie können auch Menüs an andere Menüs anfügen, d.h. hierarchische Untermenüs einrichten. Auf diese Weise können Sie Funktionen gruppieren, die zu einem Objekt innerhalb des Menüs gehören. Hierzu ein Beispiel:



Untermenüs und deren Einträge können dieselben Attribute wie die Menüs selbst haben, z.B. Aktionen, Methoden, Tastenkürzel, Icons.

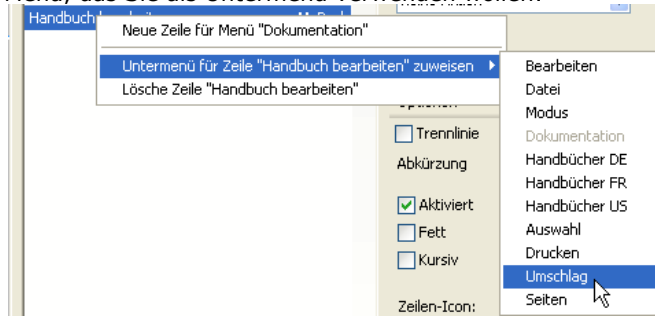
Im Menüeditor erscheinen Untermenüs als Einträge einer hierarchischen Liste:



Um ein Untermenü zu erstellen, verknüpfen Sie einfach ein vorhandenes Menü mit dem Eintrag eines anderen Menüs. Es gibt zwei Wege:

- **Drag-and-Drop:** Wählen Sie ein Menü im linken Bereich aus und ziehen es auf den Eintrag im mittleren Bereich, dem Sie das Untermenü zuordnen wollen.
- **Kontextmenü:** Klicken Sie im mittleren Bereich mit der rechten Maustaste auf den Eintrag, dem Sie das Menü zuordnen wollen, wählen Sie im Kontextmenü den Eintrag **Untermenü für Zeile „Name“ zuweisen** und dann das

Menü, das Sie als Untermenü verwenden wollen:



Das angefügte Menü wird so zum Untermenü. Der Menütitel wird beibehalten, er lässt sich jedoch verändern. Der Originaltitel des Untermenüs wird ignoriert.

Die Einträge des Untermenüs behalten ihre ursprünglichen Merkmale und Eigenschaften bei. Das Untermenü funktioniert genauso wie das Standardmenü.

Sie können auch in einem Untermenü weitere Untermenüs erstellen. Klappen Sie dazu das Untermenü im mittleren Bereich auf und fügen an den gewünschten Eintrag ein Menü an. Theoretisch können Sie Untermenüs in unbegrenzter Verschachtelung einrichten. Damit die Oberfläche übersichtlich bleibt, sollten Sie jedoch nicht mehr als zwei Ebenen mit Untermenüs einrichten.

Menü/Untermenü entfernen

Sie können jederzeit ein Menü aus einer Menüleiste bzw. ein Untermenü aus einem Menü wieder entfernen. Das entfernte Menü/Untermenü steht dann in der Menüleiste/im Menü selbst nicht mehr zur Verfügung, jedoch weiterhin in der Liste der Menüs.

Um ein Menü zu entfernen, klicken Sie mit der rechten Maustaste im mittleren Bereich auf das entsprechende Menü/Untermenü und wählen dann im Kontextmenü den Eintrag **Menü „Name“ von Menüleiste „Name“ entfernen** bzw. **Untermenü von Zeile „Name“ entfernen**.

Unabhängige Menüs

Sie können auch "unabhängige" Menüs einrichten; d.h. Menüs, die weder einer Menüleiste noch einem spezifischen Menü zugeordnet sind. Diese Menüs lassen sich im Menüeditor konfigurieren, müssen aber mit Befehlen der 4D Programmiersprache verwaltet werden (siehe Kapitel **Menüs**).

Um ein unabhängiges Menü zu erstellen, klicken Sie unter der linken Liste auf das Pluszeichen und wählen den Eintrag **Menü:**



Referenzen für Bezeichnung verwenden

Sie können für Menübezeichnungen statt hard-codiertem Text eine Referenz verwenden. Das vereinfacht die Übersetzung einer Anwendung. Es gibt drei Arten von Referenzen:

- Eine **XLIFF Ressource** vom Typ `:xliff:MeinName` oder `:10115,3` (Kompatibilität). Weitere Informationen dazu finden Sie unter **PICTURE TO BLOB**.
- Eine **Interprozessvariable** plus Nummer, z.B.: `<>vlang,3`. Ändern Sie den Inhalt dieser Variablen, erscheint bei der nächsten Anzeige des Menüs der geänderte Name. In diesem Fall ruft die Bezeichnung je nach Übersetzungsarchitektur der Datenbank eine XLIFF oder STR# Ressource auf:
 - *XLIFF Architektur*: Der Wert der Variablen `<>vlang` entspricht dem id Attribut des group Elements. Der zweite Wert (in diesem Beispiel 3) bezeichnet das id Attribut des trans-unit Elements.
 - *STR# Architektur*: Die Zeichenkette in der dritten Zeile der STR# Ressource, deren Nummer in der Interprozessvariablen `<>vlang` enthalten ist, erscheint als Menübezeichnung.
- Eine **STR# Ressource**: Geben Sie z.B. `:2000,3` ein, erscheint der Text in der 3. Zeile der Ressource STR# 20000 als Menübezeichnung. Ändern Sie den Inhalt dieser Ressource per Programmierung (siehe *4D Programmiersprache*, Kapitel **Ressourcen**) oder oder unter Mac OS über den Ressourcen-Editor, wird der Menüname entsprechend geändert und erscheint bei der nächsten Anzeige mit dem neuen Namen.

Hinweis zur Kompatibilität: 4D unterstützt Ressourcen noch zur Wahrung der Kompatibilität. Dieser Mechanismus ist jedoch überholt und sollte nicht länger verwendet werden. Wir empfehlen, Ihre dynamischen Oberflächen auf Variablen oder XLIFF Architektur aufzubauen.

Aktion für Menübefehle festlegen

Damit ein Menübefehl seine Funktion durchführt, müssen Sie ihm entweder eine Projektmethode oder eine Standardaktion zuweisen. Der Menübefehl **Monatsbericht** ruft z.B. eine Projektmethode auf, die aus der Tabelle mit den Finanzdaten einen monatlichen Bericht erstellt. Der Menübefehl **Ausschneiden** ruft die Standardaktion *Ausschneiden* auf, um die Auswahl in die Zwischenablage zu legen und aus dem Fenster im Vordergrund zu entfernen. Wird ein Menübefehl gewählt, führt 4D die zugehörige Projektmethode bzw. Standardaktion aus.

Die Wahl einer Standardaktion oder Projektmethode für einen Menübefehl richtet sich nach dem gewünschten Ergebnis. In der Regel sollten Sie Standardaktionen den Vorzug geben, da ihre Arbeitsweise optimiert ist.

Sie können einen eigenen Menübefehl auch mit einer Standardaktion und einer Projektmethode verknüpfen. In diesem Fall wird die Standardaktion nie ausgeführt; 4D verwendet dagegen die Aktion, um den Menübefehl je nach Kontext zu aktivieren/deaktivieren. Wird ein Menübefehl deaktiviert, kann die dazugehörige Projektmethode nicht ausgeführt werden.

Die Projektmethode erstellen Sie im **Methodeneditor**. Sie erstellen diese entweder bevor, oder nachdem Sie ihr einen Menübefehl zuweisen. Haben Sie einem Menübefehl im Menüeditor eine Methode zugewiesen, können Sie die dazugehörige Methode öffnen, wenn Sie im rechten Bereich auf das Icon [...] neben dem Eingabebereich für Methode klicken.

Weisen Sie dem Menübefehl weder eine Standardaktion, noch eine Methode zu, kehrt 4D automatisch in den Designmodus zurück, wenn dieser Befehl im Anwendungsmodus ausgewählt wird (sofern verfügbar).

Projektmethode zuweisen

Um einem Menübefehl eine Projektmethode zuzuweisen:

1. Erstellen oder wählen Sie den Menübefehl.
Der rechte Bereich zeigt die Eigenschaften des gewählten Menübefehls.
2. Gibt es die Projektmethode bereits in der Datenbank, wählen Sie diese in der Combobox *Methodenname* aus.
ODER
Existiert die Projektmethode noch nicht, geben Sie ihren Namen in der Combobox *Methodenname* ein und klicken auf das Icon [...].
Im zweiten Fall zeigt 4D das Dialogfenster zum Erstellen der Projektmethode im Methodeneditor.
Hinweis: Wenn Sie den Namen einer Methode ändern, die in einem Menü verwendet wird, müssen Sie den Namen hier im Menüeditor aktualisieren.
3. Im Normalfall wird eine zu einem Menübefehl gehörende Methode innerhalb des aktuellen Prozesses ausgeführt, es sei denn, Sie rufen in Ihrem Code ausdrücklich einen neuen Prozess auf. Sie starten den neuen Prozess einfacher, indem Sie das Kontrollkästchen **Starte neuen Prozess** aktivieren.
Wenn Sie das Kontrollkästchen **Starte neuen Prozess** aktivieren, erstellt 4D einen neuen Prozess, sobald Sie den Menübefehl gewählt haben. 4D weist dem neuen Prozess in der Prozessliste einen voreingestellten Namen zu und verwendet hierfür das Format *ML_ProcessNumber*. Der Name von aus einem Menü gestarteten Prozess besteht aus "ML_" plus der Prozessnummer.
Weitere Informationen über Prozesse finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Prozesse**.

Standardaktion zuweisen

Um einem Menübefehl eine Standardaktion zuzuweisen:

1. Erstellen oder wählen Sie den Menübefehl.
4D hebt den gewählten Menübefehl hervor. Der rechte Bereich zeigt die Eigenschaften des gewählten Menübefehls.
2. Wählen oder schreiben Sie in der ComboBox "Zugewiesene Standardaktion" die gewünschte Aktion.
Die Liste der Standardaktionen für Menüs ähnelt den Aktionen für Schaltflächen. Sie sind im 4D **Formulareditor** in den Eigenschaften für Schaltflächen verfügbar. Die meisten Aktionen können sowohl für Menübefehle als auch für Schaltflächen verwendet werden. Einige Aktionen, wie z.B. *Automatischer Splitter* kann keinem Menübefehl zugewiesen werden. Deshalb erscheint sie nicht in der Combobox. Sie können dagegen für den Bereich jede gewünschte Aktion, die unterstützt wird, und optional Parameter eingeben. Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.
Hinweis für macOS: Auf macOS werden eigene Menübefehle, denen die Aktionen **Einstellungen** und **Beenden** zugewiesen sind, automatisch in das Anwendungsmenü gelegt (gemäß den Oberflächennormen dieser Plattform).
3. (Optional) Wählen Sie die Option **Ausführen ohne Prüfung**.
Ist diese Option markiert, führt 4D vor Ausführen der zugewiesenen Aktion keine „Bestätigung“ des Datenfeldes aus, in dem der Cursor liegt.
Diese Option dient hauptsächlich für Befehle des Menüs Bearbeiten. 4D durchläuft und bearbeitet standardmäßig den Inhalt eines Feldes vor Ausführen einer Standardaktion (via Menübefehl oder Tastenkürzel), d.h. das Formularereignis *On Data Change* wird erzeugt. Dies kann die Funktionsweise von Befehlen des Typs Kopieren oder Einsetzen stören, da beim Aufrufen unerwartet das Formularereignis *On Data Change* erzeugt wird. In diesem Fall ist es sinnvoll, die Option **Ausführen ohne Prüfung** zu markieren.

Menüeigenschaften einrichten

Im rechten Teil des **Menüeditor** können Sie verschiedene Eigenschaften für Menüeinträge setzen, wie z.B. Schriftstil, Trennlinien, Tastenkürzel oder Icons.

Eigene Parameter

Sie können jedem Menüeintrag einen eigenen Parameter zuweisen. Dieser besteht aus einer Zeichenkette, deren Inhalt frei wählbar ist.

Parameter für Menüeinträge sind hauptsächlich zur Menüverwaltung per Programmierung sinnvoll, insbesondere mit den 4D Funktionen **Dynamic pop up menu**, **Get menu item parameter** und **Get selected menu item parameter**.

Trennlinien

Sie trennen Gruppen von Menübefehlen in einem Menü durch eine Trennlinie. Auf diese Weise können Sie Menübefehle nach Funktionen gruppieren.



Um eine Trennlinie hinzuzufügen, erstellen Sie einen Menübefehl und geben keinen Text ein, sondern aktivieren im rechten Teil unter Optionen das Kontrollkästchen **Trennlinie**. Anstelle von Text erscheint eine Linie im Bereich der aktuellen Menüleiste.

Hinweis: Auf Mac OS können Sie den Bindestrich "-" als erstes Zeichen einer Menüzeile verwenden, sie erscheint dann als Trennlinie. Das ist besonders hilfreich beim Einsetzen des 4D Befehls **INSERT MENU ITEM**.

Hinweis: Ist die Option **Trennlinie** markiert, bleiben andere Eigenschaften ohne Wirkung.

Tastenkürzel zuweisen

Sie können jedem Menübefehl entsprechende Tastenkürzel hinzufügen. Wenn zu einem Menübefehl eine solche Tastenkombination gehört, sieht der Benutzer das neben dem Befehl. Die Kombination **Strg-Taste + Buchstabe C** unter Windows oder **⌘+C** auf Mac OS erscheint im Menü **Bearbeiten** neben dem Befehl **Kopieren**.

Sie können Tastenkürzeln für Menübefehle auch die **Umschalttaste** und unter Windows die **Alt-Taste**, auf Mac OS die **Wahltaste** zuordnen. Das vervielfältigt die Anzahl der Tastenkürzel, die Sie für selbst erstellte Menüleisten verwenden können. Sie können also folgende Tastenkombination definieren:

- Unter Windows:
 - Strg+Buchstabe
 - Strg+Umschalttaste+Buchstabe
 - Strg+Alt+Buchstabe
 - Strg+Umschalttaste+Alt+Buchstabe
- Auf Mac OS:
 - Befehl+Buchstabe
 - Befehl+Umschalttaste+Buchstabe
 - Befehl+Wahltaste+Buchstabe
 - Befehl+Umschalt- +Wahltaste+Buchstabe

Hinweis: Wir empfehlen, die standardmäßigen Tastenkombinationen für Standardaktionen beizubehalten.

Sie können jede alphanumerische Taste als Tastenkürzel verwenden, mit Ausnahme der für Standardmenübefehle definierten Zeichen in den Menüs **Bearbeiten** und **Datei/Ablage** und Tasten die für 4D Menübefehle reserviert sind.

Die folgende Tabelle zeigt die reservierten Tastenkombinationen:

Tasten	Aktion
Strg+C	Kopieren
Strg+Q	Beenden
Strg+V	Einfügen
Strg+X	Ausschneiden
Strg+Z	Rückgängig
Strg+. (Punkt)	Aktion anhalten

Auf Mac OS verwenden Sie statt **Strg** die **Befehlstaste**.

Um ein Tastenkürzel zuzuweisen:

1. Wählen Sie den Menübefehl, dem Sie ein Tastenkürzel zuweisen wollen.
2. Klicken Sie im rechten Teil nach dem Eingabebereich für die Abkürzung auf die Schaltfläche [...]. Auf dem Bildschirm erscheint folgendes Dialogfenster:



3. Geben Sie ein Zeichen ein und aktivieren bei Bedarf die Zusatztasten **Shift** bzw. **Alt/Option**.

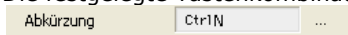
Hinweise:

- Sie können auch direkt die Tasten für die gewünschte Kombination drücken.
- Die **Strg-/Befehlstaste** lässt sich nicht deaktivieren, da sie für Tastenkürzel von Menüs zwingend ist.

Um erneut zu starten, klicken Sie auf die Schaltfläche **Löschen**.

4. Klicken Sie auf die Schaltfläche **OK**, um die Eingabe zu bestätigen.

Die festgelegte Tastenkombination erscheint jetzt im Eingabebereich:



Hinweis: Auch ein aktives Objekt kann ein Tastenkürzel haben. Wenn die Zuweisung mit der **Strg-** bzw. **Befehlstaste** einen Konflikt ergibt, hat das aktive Objekt Vorrang. Weitere Informationen dazu finden Sie im Abschnitt **Tastaturkürzel**.

Aktiver Menübefehl

Sie können festlegen, ob ein Menübefehl aktiviert oder deaktiviert ist. Einen aktiven Menübefehl kann der Benutzer anwählen, ein inaktiver Befehl erscheint in Grauschrift und ist nicht auswählbar. Solange Sie nichts anderes festlegen, aktiviert 4D jeden Befehl, den Sie im eigenen Menü hinzufügen.

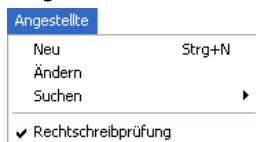
Sie können einen Befehl z.B. deaktivieren, damit er nur per Programmierung, d.h. über die 4D Befehle **ENABLE MENU ITEM** und **DISABLE MENU ITEM** aktivierbar ist.

Ist das Kontrollkästchen **Aktiviert** nicht markiert, erscheint der Befehl in Grauschrift, d.h. er lässt sich nicht auswählen.

Häkchen zuweisen

Sie können im rechten Teil unter Optionen das Kontrollkästchen **Häkchen** markieren, um einen Menüeintrag standardmäßig mit einem Häkchen zu versehen und dann seine Anzeige über die 4D Befehle **SET MENU ITEM MARK** und **Get menu item mark** zu steuern.

Häkchen dienen im allgemeinen für Menübefehle mit fortlaufenden Aktionen und geben an, dass die Aktion gerade ausgeführt wird:



Schriftstil

In 4D können Sie den Menübefehlen unterschiedliche Schriftstile geben, und zwar mit den Stilarten Fett, Kursiv und Unterstrichen.

Verwenden Sie die Schriftstile mit Bedacht, denn zuviele verwirren den Benutzer und Ihre Anwendung wirkt überladen. Um einen Schriftstil zuzuweisen, wählen Sie den gewünschten Menübefehl und markieren im rechten Teil unter Optionen das entsprechende Kontrollkästchen.

Zeilen-Icon

Mit der Option **Zeilen-Icon** können Sie dem ausgewählten Befehl ein Icon zuordnen. Das Icon muss zuvor in der 4D **Bildbibliothek** gespeichert werden. Klicken Sie auf den Bereich neben der Option, erscheint ein DropDown-Menü, um ein passendes Bild aus der **Bildbibliothek** auszuwählen:



Hinweis: Ist die Bildbibliothek leer, erscheint der Eintrag **Bildbibliothek** nicht.

Haben Sie ein Bild gewählt, erscheint es in der Vorschau:



Im Menü erscheint es dann direkt vor dem Eintrag:

Handbuch

Umschlag

 Seiten

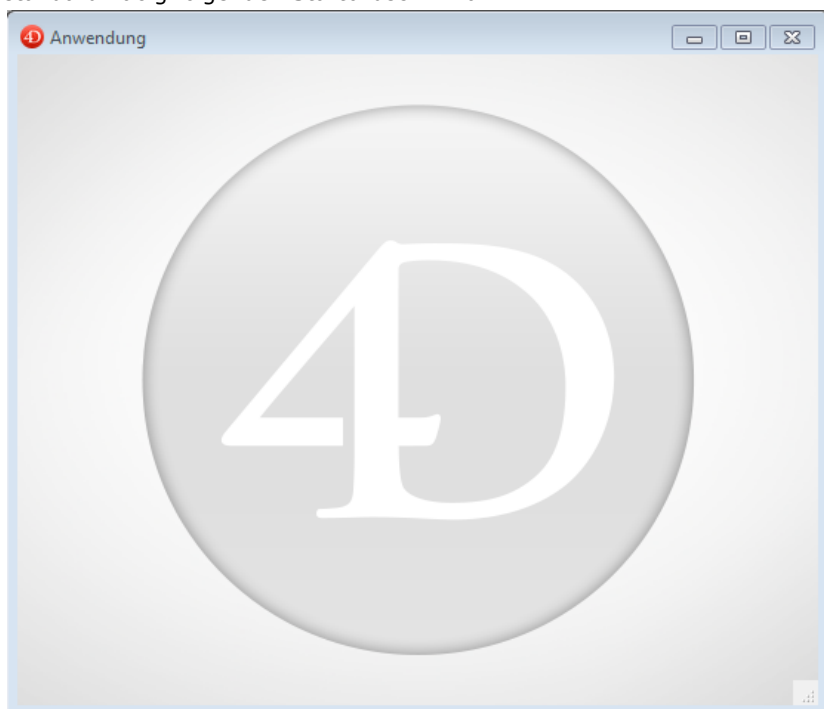
Um das Icon wieder zu entfernen, wählen Sie im DropDown-Menü den Eintrag **Kein Icon**.

Menüleisten verwalten

4D ermöglicht, jeder Menüleiste ein eigenes Bild für den Startbildschirm zuzuweisen und diese Menüleiste jederzeit als Vorschau anzuzeigen.

Startfenster einrichten

Sie können die Darstellung der Menüleiste durch ein eigenes Startfenster verbessern. Es wird unter der Menüleiste angezeigt, sobald sie erscheint. Ein Startfenster kann ein Firmenlogo oder andere Designelemente enthalten. 4D zeigt standardmäßig folgenden Startbildschirm an:



Ein eigenes Bild für das Startfenster kann von einem beliebigen Grafikprogramm kommen. Sie können das Bild aus der Zwischenablage übertragen, ein Bild aus der Bildbibliothek oder ein Bild von Ihrer Festplatte verwenden. Alle von 4D unterstützten Bildformate sind möglich.

Um das Bild des Startfensters zu ändern:

1. Wählen Sie im Menüeditor die Menüleiste, der Sie das Bild zuordnen wollen.
Beachten Sie den Bereich "Hintergrundbild" im rechten Teil des Fensters.
2. Um ein Bild direkt von Ihrer Festplatte zu öffnen, klicken Sie im rechten Teil auf die Schaltfläche **Öffnen**.
ODER
Klicken Sie in den Bereich für Hintergrundbild.
Auf dem Bildschirm erscheint ein PopUp-Menü mit den verschiedenen Möglichkeiten zum Einsetzen des Bildes:
 - Um ein Bild aus der Zwischenablage einzusetzen, wählen Sie **Kopieren**.
 - Um ein Bild aus der Bildbibliothek auszuwählen, markieren Sie **Bildbibliothek** (Ist die Bildbibliothek leer, erscheint dieser Eintrag nicht).
 - Um ein Bild auf der Festplatte zu öffnen, markieren Sie **Öffnen**.
3. Wählen Sie **Öffnen**, erscheint ein Standard-Öffnen Dialog, um das Bild an entsprechender Stelle auszuwählen. Ist ein Bild ausgewählt, erscheint es als Miniaturansicht im Bereich und wird der gewählten Menüleiste zugeordnet.



Sie sehen das Endergebnis, wenn Sie die Menüleiste testen (siehe nächster Abschnitt). Im Anwendungsmodus erscheint das Bild im Startbildschirm im Format „Abgeschnitten (zentriert)“.

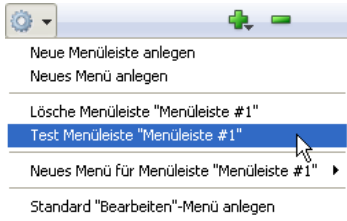
Hinweis: Sie können dieses Fenster in den Datenbank-Eigenschaften auf der **Seite Oberfläche** ein- oder ausblenden.

Wollen Sie ihr eigenes Startfenster löschen und wieder das Standardfenster von 4D anzeigen, klicken Sie in den Bereich „Hintergrundbild“ und wählen im PopUp-Menü den Eintrag **Löschen**.

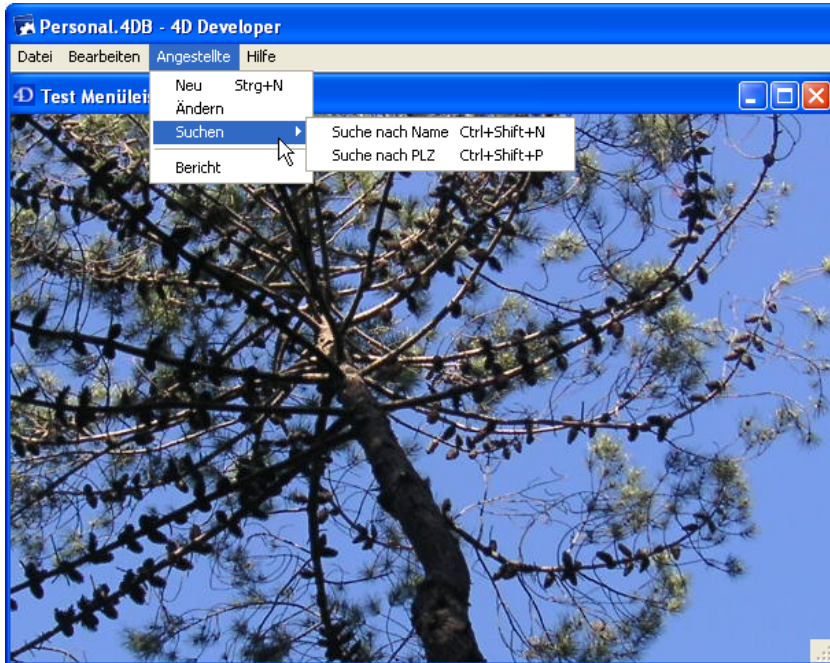
Voransicht der Menüleiste

Sie können im Menüeditor jederzeit die eigenen Menüs und den Startbildschirm anzeigen, ohne die Seite der Toolbox zu verlassen.

Markieren Sie dazu die Menüleiste und wählen im Kontextmenü oder im DropDown-Menü unter der Liste den Eintrag **Test Menüleiste „Menüleiste #“**.



4D zeigt eine Vorschau der Menüleiste und des Startbildschirms:



Sie können die Menüs und Untermenüs aufklappen, um sich den Inhalt anzusehen. Um die Funktionsweise von Menüs und Symbolleisten zu testen, müssen Sie im Menü **Start** den Befehl **Anwendung testen** aufrufen.

Menüs und eigene Anwendungen

Menüleisten sind die wichtigsten Schnittstellen für eigene Anwendungen. Sie müssen für jede eigene Anwendung mindestens eine Menüleiste mit wenigstens einem Menü erstellen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Eine 4D Anwendung aufbauen**.

Sie erstellen Menüleisten für den Anwendungsmodus, gleichgültig, ob Sie eine eigene Anwendung anlegen oder lediglich Menüs für die Verwendung in der Anwendungsumgebung erstellen. Standardmäßig wird die Menüleiste #1 im Anwendungsmodus angezeigt. Mit dem 4D Befehl **SET MENU BAR** können Sie bestimmen, welche Menüleiste angezeigt wird.

Wenn Sie einen Menübefehl festlegen, ohne ihm eine Methode zuzuweisen, wird beim Auswählen dieses Befehls der Anwendungsmodus verlassen und in die Designumgebung gewechselt - sofern der Zugriff auf diesen Modus erlaubt ist. Bei Verwendung mit 4D Desktop wird bei Verlassen des Anwendungsmodus die Anwendung beendet.

Wenn Sie die komplette 4D Anwendung verwenden, können Sie ein Zugriffssystem mit Kennwort einrichten und so den Zugriff auf den Designmodus steuern und schützen, wenn ein Benutzer den Anwendungsmodus verlässt. Sie können auch in den Datenbank-Eigenschaften eine Gruppe mit Zugriffsberechtigung auf den Designmodus definieren. Weitere Informationen dazu finden Sie im Abschnitt **Benutzern Zugriff auf die Designumgebung geben**. Benutzer ohne Zugriffsberechtigung auf den Designmodus gelangen vom Anwendungsmodus aus nicht in den Designmodus, weder über einen Menübefehl, noch über die Tastenkombination **Alt+Umschalttaste+rechte Maustaste** unter Windows oder **Ctrl-+Wahltaste+rechte Maustaste** auf Mac OS, welche das PopUp-Menü zum Bearbeiten anzeigt (siehe Handbuch *4D Programmiersprache*, Abschnitt **Fenster Debugger**). Versucht ein Benutzer ohne Berechtigung, in den Designmodus zu wechseln, wird 4D beendet. Designer und Administrator haben immer Zugriff auf die Designumgebung, selbst wenn sie nicht zur Gruppe mit Zugriffsrecht auf die Designumgebung gehören. Weitere Informationen dazu finden Sie in Kapitel **Benutzer und Gruppen**.

Überblick

Unter Windows können 4D Entwickler ihre 4D Anwendungen in 64-bit mit einkompilierter Engine als SDI (Single-Document Interface) Anwendungen einrichten. In SDI Anwendungen ist jedes Fenster von den anderen unabhängig und kann eine eigene Menüleiste haben. SDI Anwendungen sind das Gegenteil von MDI (Multiple Documents Interface) Anwendungen, wo alle enthaltenen Fenster vom Hauptfenster abhängen.

Hinweis für macOS: Das Konzept SDI/MDI gibt es nicht auf macOS. Dieses Feature gilt nur für Windows Anwendungen und damit zusammenhängende Optionen werden auf macOS ignoriert.

Verfügbarkeit

Der SDI Modus ist nur in folgender Ausführungsumgebung verfügbar:

- Windows
- Eigenständige oder Client 4D Anwendung mit einkompilierter Engine
- 4D Anwendung in der 64-bit Version

SDI Modus aktivieren

Zum Aktivieren und Einsetzen des SDI Modus in Ihrer Anwendung sind folgende Schritte erforderlich:

1. Markieren Sie auf der Seite **Oberfläche** des Dialogfensters Datenbank-Eigenschaften die Option **Verwende SDI Modus für Windows** (siehe [Seite Oberfläche](#)).
2. Erstellen Sie eine doppelklickbare Anwendung (Einzelplatz und/oder Client Anwendung, siehe [Eigenständige Anwendung erstellen und weitergeben](#)).

Die doppelklickbare Anwendung läuft dann bei Ausführen im entsprechenden Kontext automatisch im SDI Modus.

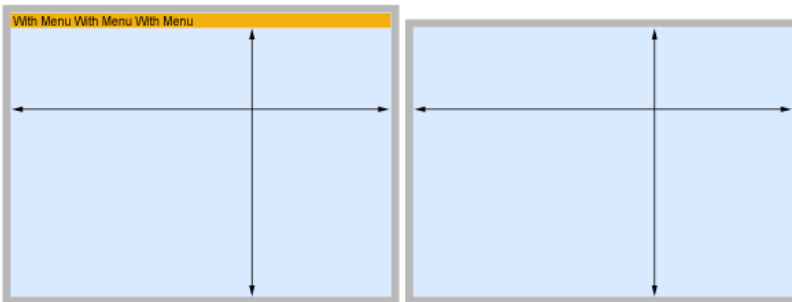
Anwendungen im SDI Modus verwalten

Zum Ausführen einer 4D Anwendung im SDI Modus sind keine spezifischen Implementierungen erforderlich: Vorhandene Menüleisten werden automatisch in die eigenen SDI Fenster gelegt. Sie müssen jedoch ein paar Eigenheiten beachten, die nachfolgend aufgelistet werden.

Menüs in Windows

Im SDI Modus erscheint die Menüleiste des Prozesses automatisch in jedem Dokument vom Typ Fenster, das während der Lebensdauer des Prozesses geöffnet wird (ausgeschlossen sind beispielsweise Palettenfenster). Ist die Menüleiste des Prozesses nicht sichtbar, bleiben die Tastenkürzel der Menübefehle weiterhin aktiv.

Menüs werden oben im Fenster hinzugefügt, die Größe des Inhalts wird dabei nicht verändert:



Fenster lassen sich also im MDI oder SDI Modus verwenden, es ist keine Neuberechnung die Position der Objekte notwendig.

Hinweis zum Hauptfenster:

- Ist für die Anwendung die Oberflächenoption **Hauptfenster** ausgewählt (siehe [Seite Oberfläche](#)), enthält das Hauptfenster alle Menüs, die auch im MDI Fenster erscheinen würden. Beachten Sie, dass Schließen des Hauptfensters - analog zum MDI Modus- auch das Verlassen der Anwendung bewirkt.
- Ist die Option **Hauptfenster** nicht für die Anwendung ausgewählt, erscheinen die Menüs nur in den geöffneten Fenstern, gemäß der Wahl des Programmierers.

Automatisch Beenden

Beim Ausführen im MDI Modus beendet die 4D Anwendung einfach, wenn der Benutzer das Anwendungsfenster (MDI Fenster) schließt. Dagegen haben 4D Anwendungen beim Ausführen im SDI Modus kein Anwendungsfenster und außerdem bedeutet Schließen des zuletzt geöffneten Fensters nicht unbedingt, dass der Benutzer die Anwendung beenden möchte (beispielsweise können Prozesse ohne Oberfläche laufen) -- obwohl das auch der Fall sein könnte.

Für diesen Fall enthalten 4D Anwendungen, die im SDI Modus ausgeführt werden, einen Mechanismus zum automatischen Beenden (Aufrufen des Befehls **QUIT 4D**), wenn folgende Bedingungen zutreffen:

- Der Benutzer kann nicht mehr mit der Anwendung interagieren

- Es gibt keine laufenden Benutzerprozesse
- 4D Prozesse oder Worker Prozesse warten auf ein Ereignis
- Der Web Server ist nicht gestartet






Hinweis: Bei Aufrufen eines Menüs mit einer zugewiesenen Standardaktion *Beenden* wird die Anwendung beendet und alle Fenster werden geschlossen, egal von wo aus das Menü aufgerufen wird.

Programmiersprache

Auch wenn 4D den SDI Modus transparent verwaltet, gibt es hier bei der Oberflächenverwaltung der Anwendung ein paar kleinere Abweichungen:

Befehl/Feature	Abweichung im SDI Modus unter Windows
Open form window	Enthält Optionen zur Unterstützung von Palettenfenstern in SDI (<u>Controller form window</u>) und zum Entfernen der Menüleiste (<u>Form has no menu bar</u>)
Menu bar height	Gibt die Höhe einer Zeile der Menüleiste in Pixel zurück, auch wenn die Menüleiste aus mehreren Zeilen besteht. Gibt 0 zurück, wenn der Befehl von einem Prozess ohne Formularfenster aufgerufen wird.
SHOW MENU BAR / HIDE MENU BAR	Gilt nur für das aktuelle Formularfenster, d.h. in dem der Code ausgeführt wird
MAXIMIZE WINDOW	Das Fenster wird auf Bildschirmgröße maximiert
CONVERT COORDINATES	<u>XY Screen</u> ist das globale Koordinatensystem, bei dem der Hauptbildschirm bei (0,0) liegt. Links davon oder oberhalb liegende Bildschirme können negative Koordinaten haben, rechts davon oder unterhalb liegende Bildschirme können Koordinaten haben, die größer sind als die von Screen height oder Screen width zurückgegebenen Werte.
GET MOUSE	Globale Koordinaten sind relativ zum Bildschirm
GET WINDOW RECT	Wird im Parameter <i>Fenster</i> -1 übergeben, gibt der Befehl 0;0;0;0 zurück
Datenbankmethode On Drop	Wird nicht unterstützt

Bildbibliothek

-  Überblick
-  Bilder verwalten
-  Bildeigenschaften
-  Thumbnails erstellen
-  Drag-and-Drop aus der Bildbibliothek

Überblick

In der Bildbibliothek speichern Sie Bilder, die Sie als Grafikelemente in Formularen, wie z.B. Bildschaltflächen, PopUp-Menüs vom Typ Bild, kleine Symbole in Listen und eigene Symbole in der Symbolleiste verwenden können. Sie können dasselbe Bild an verschiedenen Stellen Ihrer Datenbank verwenden und müssen es nur einmal abspeichern. Ändern Sie ein Bild in der Bildbibliothek, werden Bezüge auf dieses automatisch in der ganzen Datenbank aktualisiert.

So verringern Sie die Größe Ihrer Strukturdatei und vereinfachen die Mehrfachverwendung eines Bildes.

4D unterstützt die meisten gängigen Formate native. Dazu gehören die Formate JPEG, SVG, PNG, BMP und TIFF. Die Bilder werden im jeweiligen Originalformat angezeigt und gespeichert, ohne Interpretation durch 4D. Die Eigenheiten der verschiedenen Formate, wie Schatten, transparente Bereiche, etc. werden beibehalten und ohne Veränderung angezeigt. Über den 4D Befehl **PICTURE CODEC LIST** können Sie die auf dem Rechner vorhandenen native Typen erhalten.

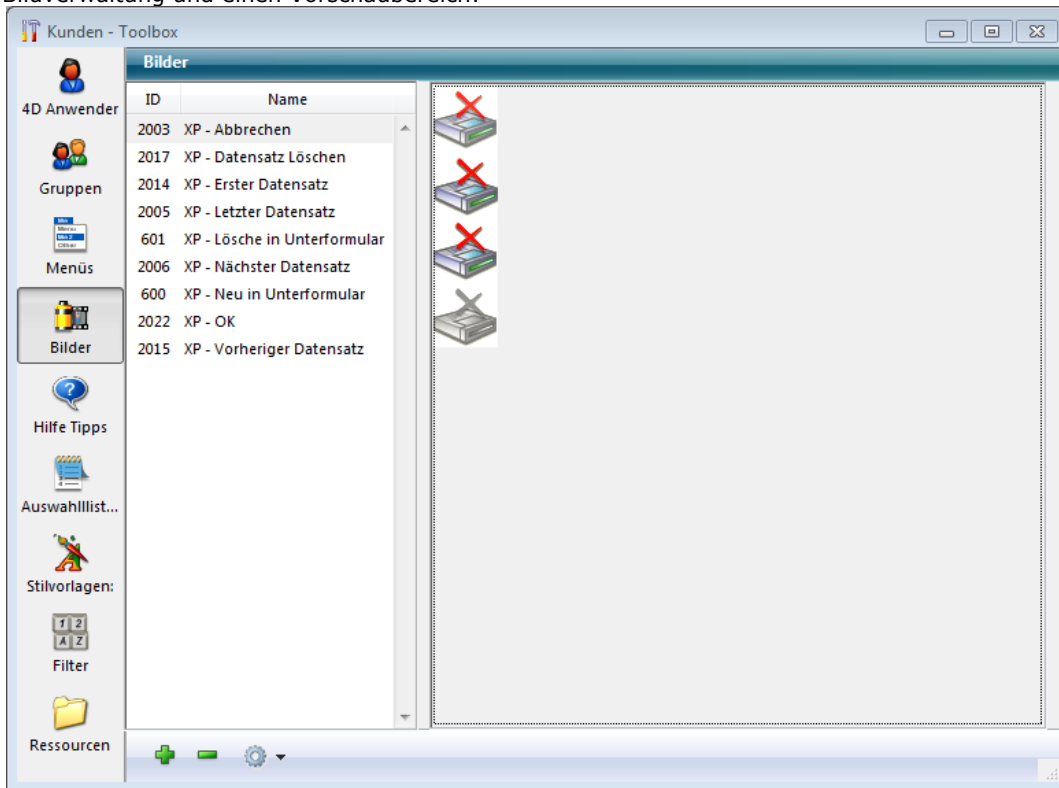
Die Bildbibliothek enthält Funktionen, um Tabellen mit Thumbnails einzurichten oder zu bearbeiten. Damit erstellen Sie Bildschaltflächen oder PopUp-Menüs vom Typ Bild.

Die Bildbibliothek wird über die Toolbox von 4D aufgerufen. Um die Bildbibliothek anzuzeigen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox>Bildbibliothek**

ODER

Klicken Sie in der Werkzeugleiste auf das Icon **Toolbox** und dann im Dialogfenster am linken Rand auf die Schaltfläche **Bilder**. Die Seite Bilder zeigt die Liste der in der Datenbank gespeicherten Bilder, Befehle zur Bildverwaltung und einen Vorschaubereich:



Über das Kontextmenü unter der Bilderliste können Sie Bilder nach Name (Standard) oder nach ID sortieren:

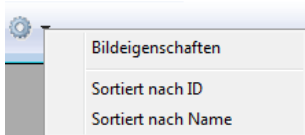



Bild importieren

4D kann die gängigsten Bildformate importieren und anzeigen. Dazu gehören die Formate JPEG, SVG, PNG, BMP, GIF und TIFF.

Um ein neues Bild in die Bildbibliothek zu importieren:

1. Klicken Sie unter der Bilderliste auf das Icon  oder mit rechtem Mausklick in die Bilderliste und wählen **Neu**. Auf dem Bildschirm erscheint ein Standard-Öffnen-Dialog. Unter Dateityp sehen Sie die von 4D unterstützten Formate.
2. Wählen Sie das Bild zum Import und klicken auf die Schaltfläche **Öffnen**
ODER
3. Ziehen Sie eine externe Bilddatei per Drag-and-Drop in die Bilderliste.


Ist das Bild gültig, erscheint das Dialogfenster Bildeigenschaften. Die Ausmaße des Bildes werden automatisch an die Ausmaße des importierten Bildes angepasst. Weitere Informationen dazu finden Sie im Abschnitt **Bildeigenschaften**.

Ändern Sie bei Bedarf Name und ID Nummer sowie andere Eigenschaften und klicken dann auf die Schaltfläche **OK**, um das Bild in der Bildbibliothek zu erstellen.

Warnung: Sie können die ID_Nummer nur ändern, wenn diese in der Bildbibliothek erstellt wurde.

Bild löschen

Um ein Bild aus der Bildbibliothek zu entfernen:

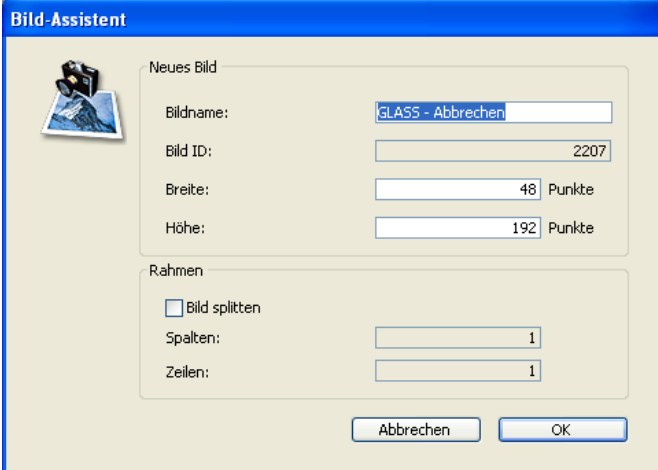
Wählen Sie das entsprechende Bild in der Liste aus und klicken am unteren Rand auf das Minuszeichen 

ODER

klicken Sie mit rechter Maustaste in das Bild und wählen im Kontextmenü den Befehl **Löschen**.

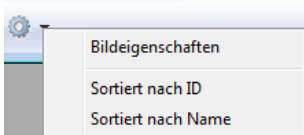
Bildeigenschaften

Im Dialogfenster Bildeigenschaften definieren oder betrachten Sie den Bildnamen, die Ausmaße sowie Eigenschaften der Bildrahmen. Auch die ID-Nummer des Bildes wird angezeigt, sie lässt sich jedoch nicht ändern.



Egal wie Sie ein Bild erstellen, es erscheint immer der Dialog zum Festlegen der Bildeigenschaften.

Sie können diesen Dialog auch jederzeit aufrufen, wenn Sie ein Bild auswählen und im Kontextmenü unter der Bilderliste auf den ersten Eintrag Bildeigenschaften klicken.



Es gibt folgende Bildeigenschaften:

- **Bildname:** Hier ändern Sie den Namen des Bildes. Sie können denselben Namen mehrmals vergeben. Die Bilder unterscheiden sich durch ihre ID Nummer.
- **Bild ID:** Hier geben Sie dem Bild eine einmalige Nummer. Das ist die Referenznummer des Bildes. Diese Nummer verwenden Sie, wenn Sie eigene Bildschaltflächen, PopUp-Menüs vom Typ Bild oder Werkzeugleisten erstellen oder Bilder aus der Bildbibliothek per Programmierung verwalten.
Hinweis: Sie können diese Nummer beim Erstellen des Bildes festlegen, sie dann später aber nicht mehr ändern.
- **Breite** und **Höhe:** Hier bestimmen Sie die Bildgröße. Diese Werte sind bereits berechnet, wenn Sie ein Bild aus einer Datei oder der Zwischenablage importieren. Splitten Sie das Bild (siehe unten), entsprechen die Werte der Größe des einzelnen Rahmens.
- **Rahmen:** Hier erstellen Sie aus einem Bild Thumbnails, um Schaltflächen oder Bildmenüs anzulegen. Weitere Informationen dazu finden Sie im Abschnitt **Thumbnails erstellen**.

Thumbnails erstellen

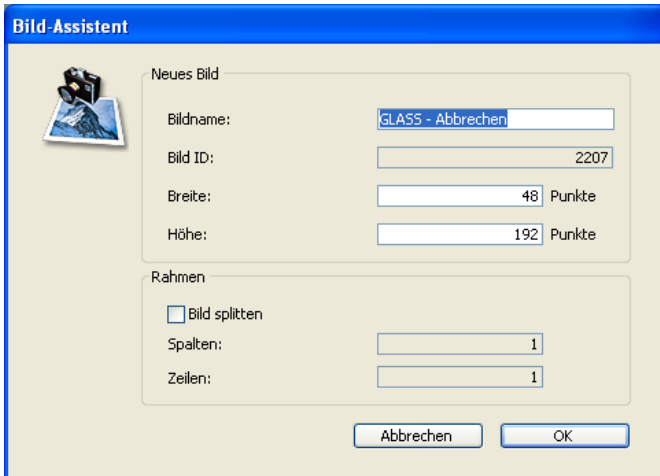
Die Bildbibliothek enthält eine Reihe Funktionen, um eine Zeile über Spaltenarrays von Bildern zur Verwendung von Bildschaltflächen oder Bildmenüs einzurichten. Das Array kann entweder ein- oder zweidimensional sein. In der Bildbibliothek heißt solch ein Array "Rahmen." An anderer Stelle in 4D kann ein Array mit Bildern auch "Thumbnails" oder "Bildertabelle" heißen. Dahinter steckt die Idee, ein Bild in Zeilen und Spalten zu splitten, so dass Zellen entstehen. Jede Zelle ist dann ein Rahmen bzw. ein „Thumbnail“. 4D zeigt je nach den festgelegten Parametern in der Bildschaltfläche oder dem Bildmenü den richtigen Ausschnitt. Weitere Informationen dazu finden Sie in den Abschnitten **Bildschaltflächen** und **PopUp-Menü vom Typ Bild**).

Sie können beim Erstellen des Bildes oder auch später eine Rahmenfolge definieren.

Rahmenfolge erstellen

Sie können eine Rahmenfolge in einem Bild erstellen, das schon in der Bildbibliothek existiert oder das sie neu hinzufügen. In beiden Fällen legen Sie im Dialogfenster **Bildeigenschaften** fest, wie das Bild aufgeteilt werden soll. Erstellen Sie ein Bild, erscheint der Dialog automatisch. Sonst doppelklicken Sie auf das Bild oder wählen das Bild aus und wählen im DropDown-Menü unter der Liste den Eintrag **Bild-Assistent**.

Im Bereich Rahmen legen Sie die Anzahl der Spalten und Zeilen Ihrer Rahmenfolge fest. Dazu müssen Sie zuerst die Option **Bild splitten** markieren:



The screenshot shows the 'Bild-Assistent' dialog box with the following fields and options:

- Neues Bild**
 - Bildname: GLASS - Abbrechen
 - Bild ID: 2207
 - Breite: 48 Punkte
 - Höhe: 192 Punkte
- Rahmen**
 - Bild splitten
 - Spalten: 1
 - Zeilen: 1

Buttons: Abbrechen, OK

Rahmengröße

4D berechnet automatisch die Rahmengröße. Definieren Sie die Rahmenfolge, ändern sich die Werte in "Breite" und "Höhe". Die Größe des einzelnen Rahmen wird angezeigt.

Wollen Sie die Rahmengröße später verändern, müssen Sie nur in "Breite" und "Höhe" neue Werte eingeben, ohne sich um die Gesamtgröße des Bildes zu kümmern. Jeder resultierende Rahmen wird automatisch in der neuen Größe zentriert, wenn er größer ist. Das Bild wird dabei nicht verzerrt. Ist die neue Größe kleiner, wird jeder resultierende Rahmen abgeschnitten.

Drag-and-Drop aus der Bildbibliothek



Ist die Option **Bild splitten** markiert (siehe **Thumbnails erstellen**), können Sie es über Tastaturkürzel als **Bildschaltflächen** oder **PopUp-Menü vom Typ Bild** in Ihre Formulare einfügen.

- Um eine Bildschaltfläche zu erstellen, setzen Sie das Bild per Drag-and-Drop aus der Bildbibliothek in das Formular.
- Um ein PopUp-Menü vom Typ Bild zu erstellen, setzen Sie das Bild bei gedrückter **Umschalttaste** per Drag-and-Drop aus der Bildbibliothek in das Formular.

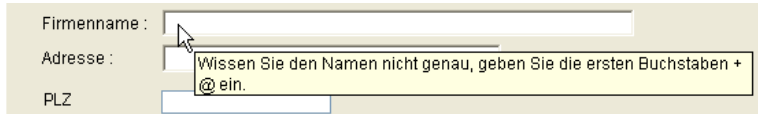
Setzen Sie ein Bild per Drag-and-Drop ein, das nicht als Rahmenfolge angelegt wurde, wird es als Standardbild eingesetzt.

Hinweis: Im Formulareditor können Sie mit der Eigenschaftenliste zwischen zwei Bildtypen unterscheiden: Bild aus der Bibliothek (wird dynamisch aktualisiert, wenn das Ursprungsbild in der Bibliothek geändert wird.) und Statisches Bild (ist nicht dem Ursprungsbild der Bibliothek zugeordnet). Weitere Informationen dazu finden Sie im Abschnitt "Eingefügtes Bild vom Ursprungsbild unabhängig machen" unter **Statische Bilder verwenden**.

Hilfetipps

-  Überblick
-  Hilftetipp Editor

Sie können Datenfeldern und aktiven Objekten in Ihrem Formular eine Hilfmeldung (Tipp) hinzufügen, um den Benutzern das Arbeiten mit Ihrer Datenbank zu erleichtern. Der Tipp erscheint im Formular immer, wenn die Maus über das Datenfeld oder Objekt bewegt wird.



The screenshot shows a form with three input fields: 'Firmenname', 'Adresse', and 'PLZ'. A mouse cursor is hovering over the 'Adresse' field, which has triggered a tooltip. The tooltip contains the text: 'Wissen Sie den Namen nicht genau, geben Sie die ersten Buchstaben + @ ein.'

Hinweise:

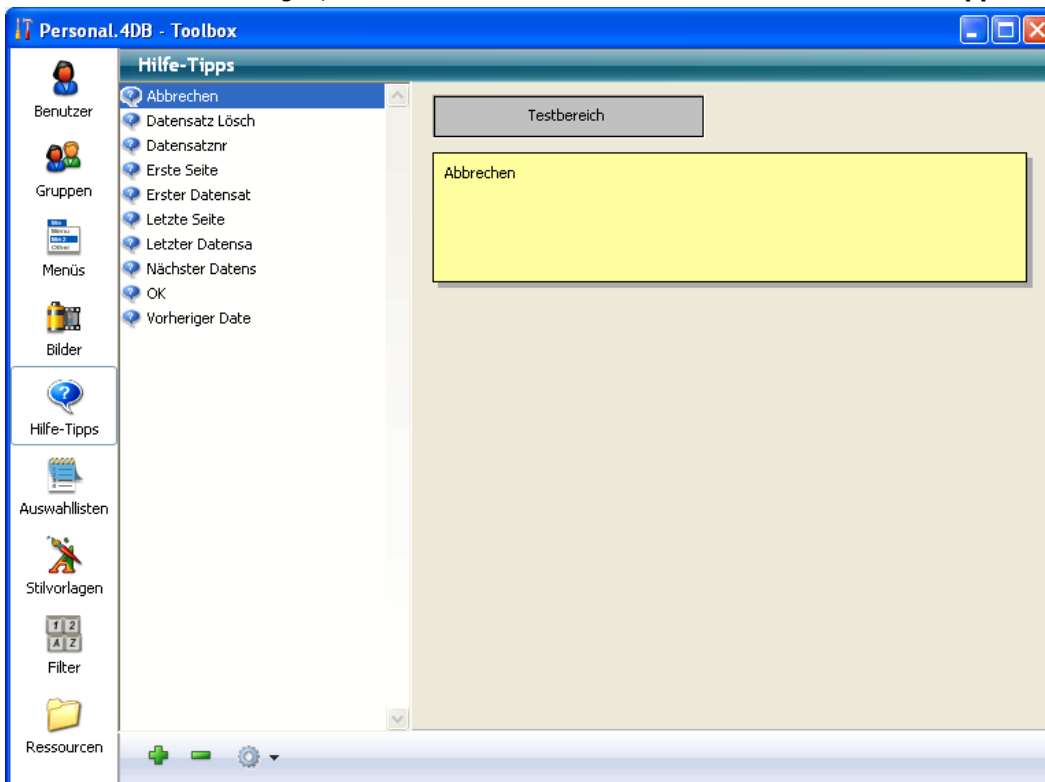
- Über den Selektor `Tips enabled` des Befehls **SET DATABASE PARAMETER** lassen sich Hilfetipps für die gesamte Anwendung deaktivieren oder aktivieren.
- Schriftstil und -größe für Hilfetipps, die während Ausführen des Formulars angezeigt werden, können Sie in den Datenbank-Eigenschaften auf der **Seite Allgemein** setzen.

Sie weisen Hilfmeldungen einem Datenfeld oder Objekt im Formulareditor direkt über die Eigenschaftenliste oder über eine Referenz auf einen Hilfetipp hinzu, die in der **Hilfetipp Editor** erstellt wurde. Weitere Informationen dazu finden Sie im Abschnitt "Hilfmeldungen" unter **Eingabekontrollen**.

Hinweis: Sie können Hilfetipps auch auf der Strukturebene setzen (siehe Abschnitt **Feldeigenschaften**) oder dynamisch über den Befehl **OBJECT SET HELP TIP** für den Prozess.

Hilfetipp Editor

Mit dem Hilfetipp-Editor erstellen und testen Sie Hilfetipps, die Sie dann Datenbankobjekten zuordnen können. Um diesen Editor anzuzeigen, klicken Sie in der 4D Toolbox auf die Schaltfläche **Hilfetipps**:

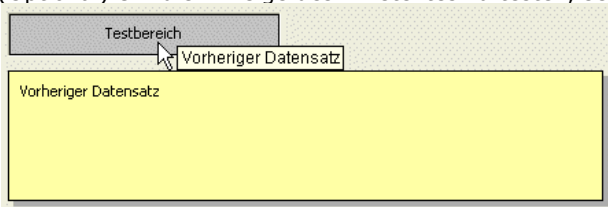


- Die Liste im linken Teil enthält die Tipps, welche für Standardformulare oder über den Formularassistenten automatisch angelegt werden.
- Im grauen Bereich testen Sie die Hilfetipps
- Im gelben Bereich geben Sie den Text der Hilfetipps ein


Hilfetipps erstellen

Um einen Tipp über den Hilfetipp-Editor zu erstellen:

1. Klicken am unteren Rand auf das Pluszeichen oder wählen Sie im Kontextmenü den Eintrag **Hinzufügen**. Es erscheint, wenn Sie mit der rechten Maustaste in die Liste klicken. In der Liste wird ein neuer Eintrag mit Namen „Meldung 1“ hinzugefügt.
2. (Optional) Überschreiben Sie „Meldung 1“ mit einem eigenen Namen. Der Name des Tipps dient in der Anwendung als Referenz, er ist für den Benutzer nicht sichtbar. Er kann bis zu 15 Zeichen lang sein. Wollen Sie den Namen später wieder ändern, drücken Sie unter Windows die **Alt-Taste**, auf Macintosh die **Wahltaste** und klicken dann in der Liste auf den Namen der Meldung. Sie können auch im Kontextmenü der Liste den Eintrag **Umbenennen** verwenden. Ändern Sie nicht den Standardnamen, erstellt 4D Tipps mit den Namen Meldung 1, Meldung 2, etc. **Hinweis:** Hilfemeldungen, die direkt in der Eigenschaftenliste eingegeben werden, erscheinen nicht in der Liste der Toolbox. Die Namen werden in alphabetischer Reihenfolge angezeigt. Fügen Sie neue Hilfemeldungen hinzu, wird der Inhalt der Liste neu geordnet.
3. Betätigen Sie die **Tabulatortaste** oder klicken Sie in den Eingabebereich und tragen den Hilfetext ein. Sie können bis zu 255 Zeichen eingeben. Auch dynamische Referenzen sind möglich - mit Ausnahme von XLIFF Referenzen (siehe **Referenzen in statischem Text verwenden**). Beachten Sie, dass dynamische Referenzen vom Typ Feld oder Variable nicht im Testbereich des Editors angezeigt werden.
4. (Optional) Um die Anzeige des Hilfetextes zu testen, setzen Sie den Mauszeiger ohne Klicken auf den Testbereich.



Auf diese Weise können Sie sehen, wie der Hilfetipp in der Datenbank erscheint.


5. Wollen Sie einen neuen Hilfetext erstellen, klicken Sie am unteren Rand auf das Pluszeichen  oder wählen Sie im Kontextmenü den Eintrag **Hinzufügen**. Es erscheint, wenn Sie mit der rechten Maustaste in die Liste klicken.

ODER





Wollen Sie eine neue Meldung auf der Basis einer bereits vorhandenen erstellen, markieren Sie diesen Hilfetext und wählen im Kontextmenü oder im Menü **Optionen** unter der Liste den Befehl **Duplizieren**.

Hilfetipp ändern oder löschen

Sie können jeden Hilfetext wieder ändern, indem Sie ihn auswählen und den Inhalt verändern. Um die Änderung zu bestätigen, drücken Sie die **Tabulatortaste** oder klicken außerhalb des Bereichs.

Wollen Sie einen Hilfetext löschen, markieren Sie diesen und klicken am unteren Rand auf das Minuszeichen  oder wählen Sie im Kontextmenü den Befehl **Löschen**.

Auswahllisten

-  Überblick
-  Listen zur Dateneingabe gestalten
-  Listen erstellen und ändern
-  Listeneigenschaften definieren

Eine Liste ist ein Satz möglicher Werte. Sie können eine Liste für folgende Zwecke verwenden:

- Verschiedene Wahlmöglichkeiten anbieten, aus denen der Benutzer einen Eintrag für ein Datenfeld oder ein eingebbares Objekt auswählen kann.
- Die möglichen Werte auf die Einträge in der Liste beschränken
- Die in der Liste aufgeführten Einträge von der Eingabe ausschließen

Sie können auch hierarchische Listen erstellen, d.h. jedem Element der Hauptliste ist eine Unterliste zugeordnet.

4D ermöglicht das Verknüpfen eines Icons mit jedem Zeileneintrag in einer Liste oder einer hierarchischen Liste. Bei Bedarf wird das Icon vor dem Zeileneintrag angezeigt. Sie können zum Beispiel die Icons in rollbaren Bereichen anzeigen (siehe **PopUp-Menüs, Dropdown-Listen**), für **Registerkarten** und für hierarchische Listen (siehe **Hierarchische PopUp-Menüs und hierarchische Listen**).

Wenn eine Liste als Auswahlliste für ein Datenfeld oder ein eingebbares Objekt verwendet wird, kann der Benutzer die Eingabe in der Liste auswählen, statt sie einzutippen. Sie möchten zum Beispiel eine Auswahlliste für die Eingabe von Stellenbeschreibungen in einer Personaldatenbank erzeugen.

Sie können Listen auch zur Beschränkung von Dateneingaben verwenden. Eine Liste kann die erforderlichen Werte für ein Datenfeld enthalten und alle anderen ausschließen. Eine andere Liste kann die ausgeschlossenen Werte für ein Datenfeld enthalten und damit verhindern, dass diese Werte in der Liste eingegeben werden.

Ihre Listen können pro Datenbank bis zu 8.000 Wahlmöglichkeiten enthalten, jedes Element kann bis zu 2 Milliarden Zeichen lang sein.

Weitere Informationen zum Hinzufügen einer Auswahlliste als Eigenschaft eines Datenfeldes finden Sie im Abschnitt "Auswahlliste" unter **Feldeigenschaften**. Weitere Informationen zur Verwendung von Listen mit Eingabekontrollen finden Sie im Abschnitt **Eingabekontrollen**.

Listen werden oft in Methoden verwendet. Zum Beispiel ist eine Liste praktisch, um die Elemente eines Arrays abzulegen. Ein Array speichert eine Liste von Werten. Sie können Listen zum Speichern von Elementen aus PopUp-Menüs, hierarchischen Listen, Combo Boxen, Registerkarten und anderen Objekten der Benutzeroberfläche mit mehreren Werten verwenden. Sie übertragen die Inhalte der Liste auf das Oberflächenobjekt mit der Methode des Objekts oder durch Zuweisen der Liste zum Objekt in der Eigenschaftsliste.

Sie erzeugen 4D Listen im Listeneditor der Toolbox.

Hinweis für 4D Server: Objekte werden gesperrt, wenn zwei oder mehr Benutzer versuchen, die gleiche Liste zum selben Zeitpunkt zu ändern. Wenn ein Benutzer eine Liste in der Designumgebung ändert, wird die Liste gesperrt. Andere Benutzer können die Liste, den Listennamen oder Einträge in der Liste erst ändern, wenn der erste Benutzer die Liste durch Schließen freigibt.

Listen zur Dateneingabe gestalten

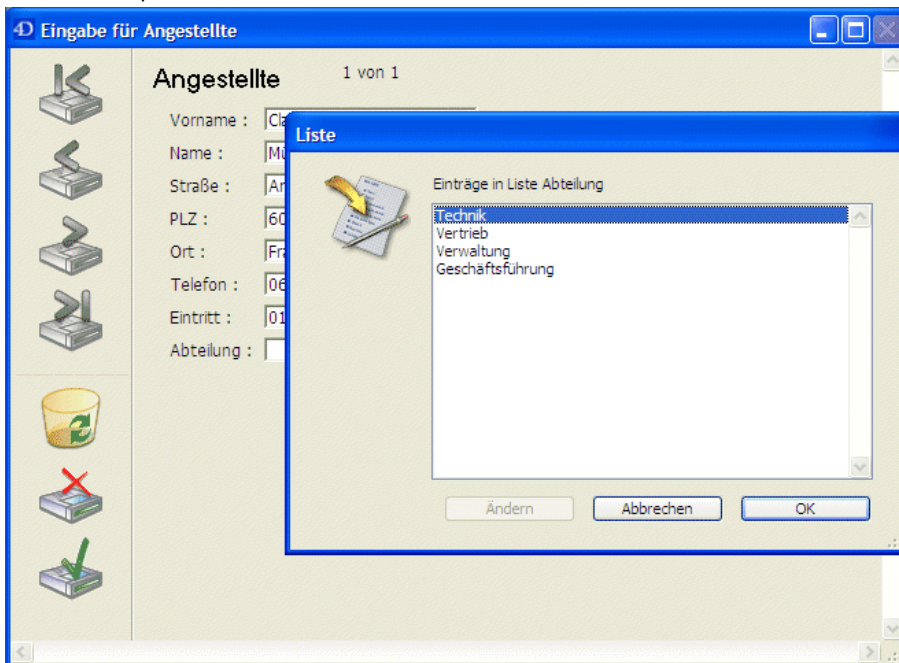
Ein Verwendungszweck von Listen ist, dem Benutzer Werte anzubieten, aus denen er bei der Dateneingabe wählen kann. Es gibt folgende Möglichkeiten:

- Eine Liste für jedes oder für ausgewählte Formulare verfügbar machen
- Lange Listen in mehrere kleinere Listen aufteilen
- Die möglichen Eingaben auf die Einträge der Liste beschränken oder zulassen, dass der Benutzer andere Einträge eingibt
- Erlauben oder ausschließen, dass der Benutzer die Liste ändert
- Ein Feld oder eine Variable mit Liste über PopUp/DropDown Liste oder Combobox direkt im Formular darstellen

Tipp: Ist die Anzahl der Elemente in einer Liste beschränkt, ist eine Auswahlliste nicht immer die sinnvollste Lösung. So ist beispielsweise für die beiden Kriterien „männlich“ und „weiblich“ ein Feld vom Typ Boolean besser geeignet, das einem Optionsfeld bzw. Biloptionsfeld zugewiesen ist. Selbst bei drei oder vier Kriterien sind Optionsfelder u.U. passender.

Sie können eine Liste mit einem Datenfeld als eine Datenfeldeigenschaft verknüpfen, die sie auf Strukturebenen definieren. Dann erscheint die Liste immer in einem Fenster, wenn dieses Datenfeld bei der Dateneingabe ausgewählt wird. Das kann im Listenmodus oder Seitenmodus des Formulars sein. Der Benutzer kann in der Liste einen Wert auswählen. Ist die Liste sortiert, wird automatisch weitergeblättert, wenn der Benutzer Buchstaben eintippt. Gibt er zum Beispiel "N" ein, blättert die Liste zum ersten Zeileneintrag, der mit "N" beginnt. Erscheint der gewünschte Wert, kann er mit dem Eintippen aufhören und den Eintrag für die Eingabe auswählen.

Hier ein Beispiel mit Auswahlliste:



Wenn Sie eine Liste über die Feldeigenschaften im Structureditor mit dem Datenfeld verknüpfen, erscheint die Liste auch, wenn das Datenfeld im Sucheditor ausgewählt wird. Weitere Informationen dazu finden Sie im Abschnitt "Auswahlliste" unter **Feldeigenschaften**.

Sie können die Liste dem Datenfeld oder der Variablen auch als Eingabekontrolle in einem Formular zuweisen. Diese Liste erscheint nur, wenn das Datenfeld oder die Variable in diesem Eingabeformular ausgewählt wird.

In diesem Fall können Sie die Eingabe in ein Feld auch direkt über **PopUp-Menüs**, **Dropdown-Listen** (nicht änderbare Auswahllisten) oder **Combo Boxen** (nicht änderbare Auswahllisten) steuern. Dazu weisen Sie die Liste dem Objekt zu und geben dann den Feld- oder Variablennamen direkt im Bereich **Variablenname** der Eigenschaftenliste ein. Weitere Informationen dazu finden Sie im **PopUp-Menüs**, **Dropdown-Listen**.



Hierarchische Listen

Mit 4D können Sie hierarchische Listen erstellen. Wählen Sie einen Eintrag in der Hauptliste, erscheint dann eine Unterliste.

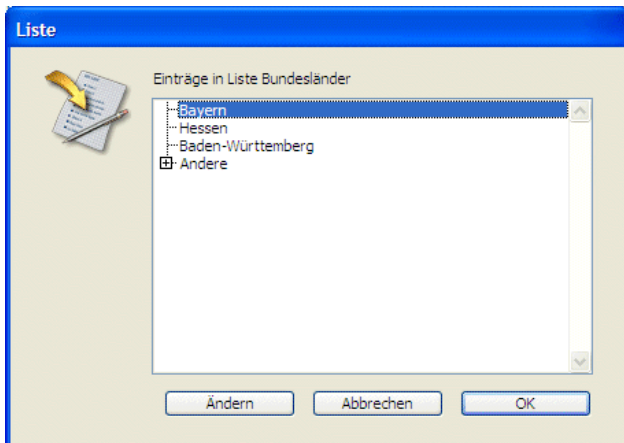
Es gibt verschiedene Möglichkeiten, eine Unterliste einzusetzen. Gehen wir z.B. von einer Liste mit den deutschen Bundesländern und ihren Regionen aus; ohne hierarchische Gliederung würde sie an die 100 Werte enthalten, was die Auswahl verlangsamt. Sie können diese Liste nach verschiedenen Kriterien hierarchisch unterteilen.

Hierarchische Listen sind eine wertvolle Hilfe und Kontrolle bei der Eingabe. Sie sind für folgende Objekte möglich:

- **Registerkarten**
- **Hierarchische PopUp-Menüs und hierarchische Listen**
- **Listboxen**

Hinweis: Sie können hierarchische Listen auch **PopUp/DropDown-Menüs** oder **Combo Boxen** zuordnen. In diesem Fall wird aber nur die erste hierarchische Ebene angezeigt.

Oft genügt es, eine Liste in zwei Listen zu unterteilen. Gelten z.B. 80 % der Werte für ganz bestimmte Bundesländer, können Sie diese in eine Liste setzen und den Rest in eine Unterliste.



Mit dieser Einteilung kann der Benutzer die Eingabe überwiegend mit der ersten Liste durchführen. Bei einem Bundesland, das nicht in dieser Liste enthalten ist, klickt er auf den Eintrag **Andere** und erhält eine Unterliste mit den restlichen Bundesländern.

Eine andere Möglichkeit wäre, die Liste in Bundesländer und Städte einzuteilen. Der Benutzer wählt dann in der Hauptliste das Bundesland, in der jeweils untergeordneten Liste die dazugehörige Stadt aus. Das ist noch schneller als ein Bundesland aus einer längeren Liste auszuwählen.

Zwingende und ausgeschlossene Werte

Einige Dateneingaben können zu Problemen führen. Sie haben unter Umständen eine Anwendung, in der nur bestimmte Werte eingegeben werden dürfen. Andere Werte würden vielleicht ernstzunehmende Konsequenzen wie z.B. die verspätete Zahlung von Rechnungen haben.

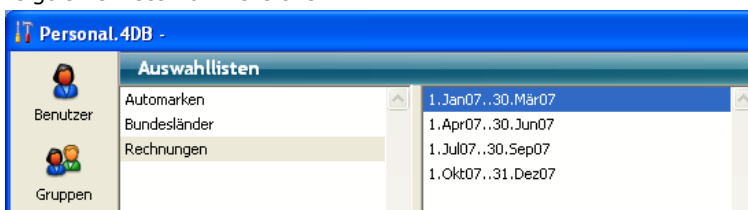
4D ermöglicht Ihnen, die Werte einer Liste für die Eingabe im Formular zwingend zu machen. Diese Art der Eingabekontrolle verhindert, dass der Benutzer unzulässige Werte eingibt. Zum Beispiel könnte Ihr Unternehmen bestimmte Stellenbezeichnungen haben, die in einer Personaldatenbank zugelassen sind.

Eine andere Dateneingabekontrolle ermöglicht es, Werte in einer Liste auszuschließen. Der Benutzer kann nur einen Wert eingeben, der nicht in der Liste erscheint. Ihr Unternehmen darf vielleicht Geschäfte in einigen Ländern nicht betreiben. Wenn diese Länder in eine Ausgeschlossen-Liste aufgenommen werden, können sie nicht eingegeben werden.

Zulässige Wertebereiche

Eine effektive Eingabekontrolle ist die Festlegung von Maximum- und Minimumwerten für ein Datenfeld vom Typ Zahl, Datum oder Zeit. So wird verhindert, dass ein Benutzer einen Wert außerhalb dieses Bereichs eingibt

Angenommen, Sie haben drei zugelassene Bereiche für das Datenfeld. Für jeden Bereich können Sie eine Liste mit zwingenden Werten erstellen. Dann werden Werte außerhalb der drei Bereiche nicht akzeptiert. Die folgende Abbildung zeigt eine Liste von Bereichen:



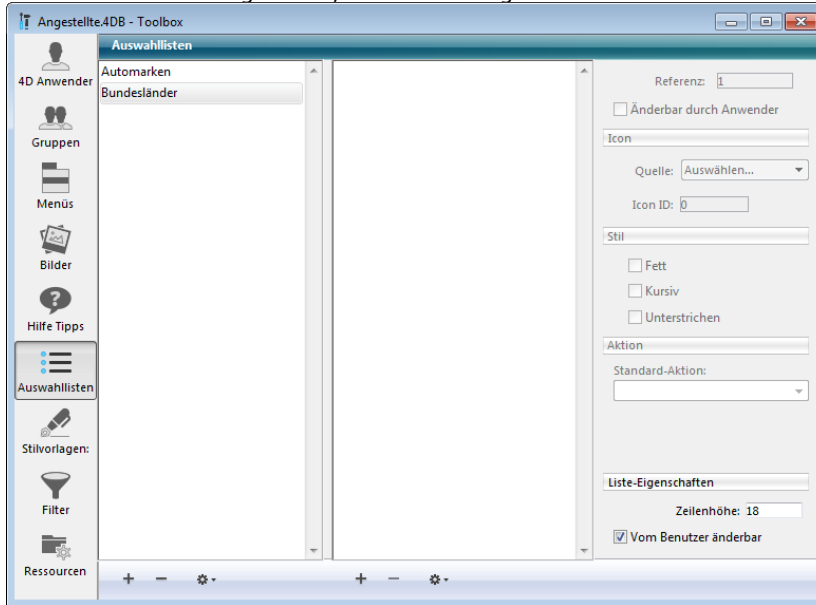
Umgekehrt können Sie auch eine Liste mit nicht zulässigen Werten erzeugen und für ein Datenfeld zur Ausgeschlossen-Liste machen. Diese Werte lassen sich dann nicht eingeben.

Listen erstellen und ändern


Liste erstellen

Sie erzeugen Listen im Listeneditor der Toolbox. Hier können Sie eine angelegte Liste auch jederzeit wieder ändern. Um eine Liste zu erstellen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox** und klicken am linken Rand auf den Icon **Auswahllisten**. Der Listeneditor wird geöffnet; standardmäßig ist er leer:

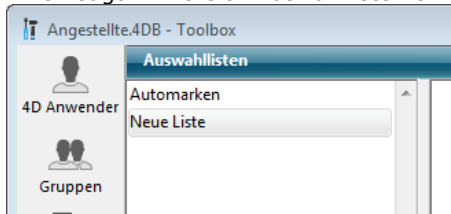


Der Listeneditor zeigt die Namen der bestehenden Listen im linken Bereich. Der mittlere Bereich zeigt die Elemente der aktuellen Liste, der rechte Bereich die Eigenschaften des aktuellen Eintrags und der aktuellen Liste.

2. Klicken Sie unter den Auswahllisten auf das Pluszeichen 

ODER

Klicken Sie mit der rechten Maustaste in die Auswahllisten und wählen im Kontextmenü den Eintrag **Neu**. 4D erzeugt im Bereich Auswahllisten ein neues Element mit Namen *Neue Liste N*.



Hinweis: Gibt es bereits eine Auswahlliste im linken Bereich, ist im Kontextmenü der Befehl **Duplizieren** verfügbar. Damit können Sie rasch eine neue Auswahlliste mit den Eigenschaften der bereits bestehenden Liste erzeugen.


3. Ändern Sie den Namen der neuen Liste und drücken die **Tabulatortaste**, um die Eingabe zu bestätigen. Sie können eine Liste jederzeit umbenennen. Doppelklicken Sie dazu auf die Auswahlliste, wählen Sie im DropDown-Menü unter der Liste oder im Kontextmenü den Eintrag **Umbenennen** oder unter Windows die Tastenkombination **Alt-Klick**, auf macOS **Wahltaste+Klick**.

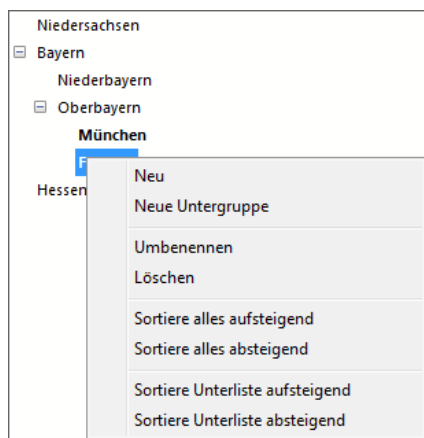
Sie haben eine neue leere Liste erzeugt. Als nächstes erstellen Sie die Elemente in dieser Liste.

In Listen Einträge hinzufügen

Wenn Sie Einträge hinzufügen, können Sie diese am Ende der Liste anhängen oder an beliebiger Stelle in der bestehenden Liste einfügen.

Um Elemente anzufügen:

1. Wählen Sie den Namen der entsprechenden Liste aus. Enthält die Liste bereits Einträge, erscheinen sie im mittleren Bereich. Wollen Sie einen Wert zwischen bereits vorhandenen Elementen einfügen, markieren Sie den Wert davor. Der neue Wert wird dann nach diesem Wert eingefügt.
2. Klicken Sie auf das Pluszeichen  unter der mittleren Liste
ODER
Wählen Sie im Kontextmenü oder im DropDown-Menü unter der mittleren Liste den Eintrag **Neu**



4D erzeugt ein neues Element mit Namen *Neuer Eintrag # N*.

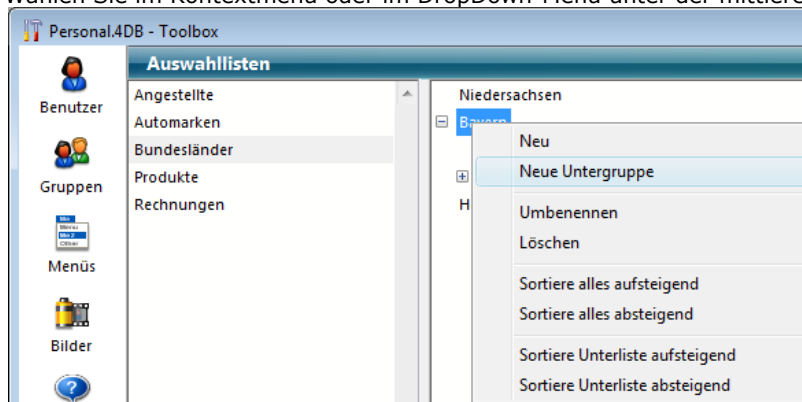
- Geben Sie einen anderen Namen ein und drücken die **Tabulatortaste**, um die Eingabe zu bestätigen. Sie können ein Element jederzeit umbenennen. Doppelklicken Sie dazu auf den entsprechenden Namen oder wählen Sie im Kontextmenü oder im Menü **Optionen** am unteren Rand den Eintrag **Umbenennen**.
- Um weitere Elemente hinzuzufügen, wiederholen Sie die vorhergehenden Schritte. Sie können eingetragene Werte per Drag-and-Drop beliebig umorganisieren und auch in alphabetischer Reihenfolge sortieren. Weitere Informationen dazu finden Sie im nachfolgenden Abschnitt "Liste sortieren".

Hierarchische Liste erzeugen

Sie können jedem Listenelement eine Liste hinzufügen. Die Anzahl der hierarchischen Ebenen ist unbegrenzt.

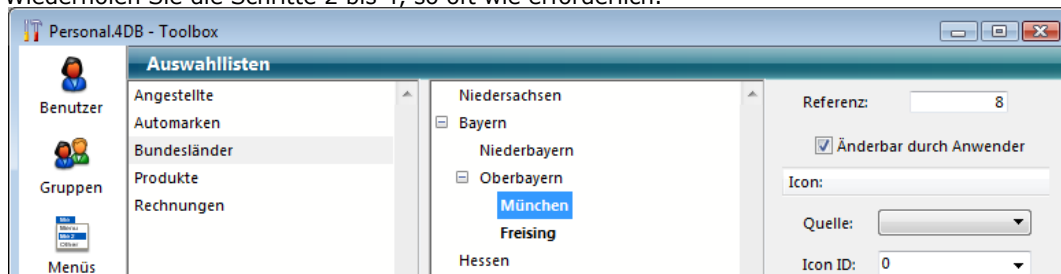
Um eine hierarchische Liste zu erstellen:

- Wählen Sie den Listeneintrag aus, mit dem die Unterliste verknüpft werden soll.
- Wählen Sie im Kontextmenü oder im DropDown-Menü unter der mittleren Liste den Eintrag **Neue Untergruppe**.



4D klappt das ausgewählte Element auf und fügt in der Unterliste ein neues Element mit Namen *Neuer Eintrag # N* hinzu.

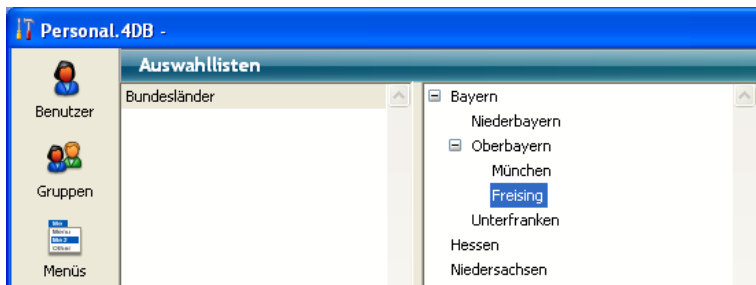
- Geben Sie den Wert ein. Sie können einen Wert jederzeit umbenennen. Doppelklicken Sie dazu auf den Eintrag oder wählen Sie im DropDown-Menü unter der Liste bzw. im Kontextmenü den Eintrag **Umbenennen**.
- Um weitere Einträge in die Unterliste einzufügen, markieren Sie den Eintrag und wählen das Pluszeichen **+** unter der Liste.
ODER
Wählen Sie den Haupteintrag und erneut den Befehl **Neue Untergruppe**. Wählen Sie **Neue Untergruppe**, wenn ein Eintrag der Unterliste ausgewählt ist, erstellen Sie eine weitere Unterebene in der Hierarchie (siehe unten).
- Wiederholen Sie die Schritte 2 bis 4, so oft wie erforderlich.



Bei Bedarf können Sie Unterlisten mit Untereinträgen verknüpfen, um die Hierarchie fortzusetzen.

Um eine Unterliste einem Untereintrag hinzuzufügen:

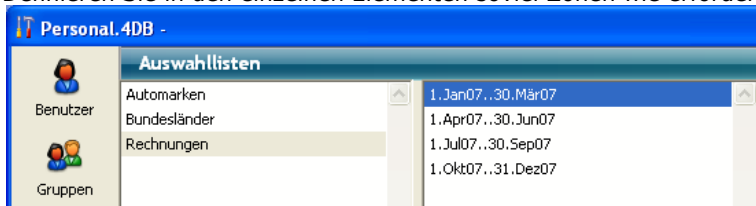
- Wählen Sie das Element der Unterliste aus.
- Wählen Sie im DropDown-Menü unter der mittleren Liste oder im Kontextmenü den Eintrag **Neue Untergruppe**.
- Tragen Sie den Wert ein und wiederholen die Schritte zum Erstellen von Unterelementen, so oft wie erforderlich. Die folgende Abbildung zeigt eine dreistufige Hierarchie:



Zonen in einer Liste festlegen

4D ermöglicht Ihnen, Zonen für Zahlen, Datum und Zeiten in einer Liste festzulegen. Sie können diese Zonen als Datenübernahmebereiche einsetzen, indem Sie die Liste in einem Formular zwingend oder ausgeschlossen machen. Um Zonen in einer Liste zu erstellen:

1. Erzeugen Sie die Liste, die Sie für Zonen verwenden wollen.
2. Geben Sie für jedes Element den Minimumwert der Zone, zwei Punkte (..) und den Maximumwert ein.
Ein Beispiel: 01.Jan07..30.Mär07
Damit wird eine Datumszone vom 1. Januar bis einschließlich 30. März definiert.
3. Definieren Sie in den einzelnen Elementen soviel Zonen wie erforderlich. Hierzu ein Beispiel:



Liste sortieren

4D behält die Reihenfolge der Listenzeilen so bei, wie sie eingegeben wurden. Sie können die Liste und auch Unterlisten alphabetisch sortieren, so dass die Eingaben für den Benutzer leichter zugänglich sind. Da eine sortierte Liste automatisch nach den eingegebenen Buchstaben blättert, erleichtert das Sortieren die Dateneingabe.

Um eine Liste oder Unterliste zu sortieren:

1. Wählen Sie im Listeneditor die Liste aus, die die Auswahl zum Sortieren enthält.
2. Klicken Sie unter dem mittleren Bereich auf das Icon für Sortieren und wählen Sie in der DropDown-Liste die passende Option aus. Es gibt folgende Möglichkeiten:
 - **Sortiere alles aufsteigend:** 4D sortiert die Auswahlliste und evtl. vorhandene Unterlisten in aufsteigender Reihenfolge, d.h. von A bis Z.
 - **Sortiere alles absteigend:** 4D sortiert die Auswahlliste und evtl. vorhandene Unterlisten in absteigender Reihenfolge, d.h. von Z bis A.
 - **Sortiere Unterliste aufsteigend:** 4D sortiert nur Unterlisten in aufsteigender Reihenfolge.
 - **Sortiere Unterliste absteigend:** 4D sortiert nur Unterlisten in absteigender Reihenfolge.

Einträge und Listen löschen

Sie können Einträge auf jeder Ebene der Hierarchie löschen. Bedenken Sie jedoch, dass Sie den Löschvorgang nicht rückgängig machen können.

Um ein Element zu löschen:

1. Wählen Sie die Auswahlliste, die das gewünschte Element enthält.
2. Wählen Sie im mittleren Bereich das Element aus, das Sie löschen wollen.
Erweitern Sie bei Bedarf die einem Element zugewiesene Unterliste.
3. Klicken Sie unter dem mittleren Bereich auf das Minuszeichen
ODER
Wählen Sie im Kontextmenü der entsprechenden Liste den Eintrag **Löschen**.
4D löscht das Element in der Liste.

Um eine Liste zu löschen:

1. Markieren Sie die Liste, die Sie löschen wollen.
2. Klicken Sie unter dem Bereich auf das Minuszeichen
ODER
Wählen Sie im Kontextmenü der entsprechenden Liste den Eintrag **Löschen**.
4D zeigt eine Meldung, um die Operation zu bestätigen oder abzubrechen. Klicken Sie auf die Schaltfläche **Löschen**, wird die Liste gelöscht.

Listeneigenschaften definieren

Dem Eintrag eine Referenznummer hinzufügen

Der rechte Bereich enthält einen Eingabebereich für die Referenznummer des Elements. Diese Nummer dient zur eindeutigen Identifizierung des Elements. Sie ist nur von Belang, wenn Sie Listen mit Methoden verwalten oder für ein Feld bzw. eine Variable mit Liste die Option **Sichern als Referenz** aktiviert haben.



Benötigen Sie die Programmiersprache, um festzulegen, welchen Eintrag in einer Liste ein Benutzer auswählt (z.B. welche Zeile eines hierarchischen Menüs), können Sie dies über die Referenznummer herausfinden. Weitere Informationen finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Hierarchische Listen**.

Option Änderbar durch Anwender

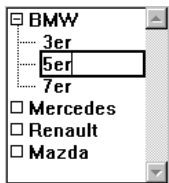
Der Listeneditor bietet für jedes Element einer Liste die Option **Änderbar durch Anwender**. Sie ist standardmäßig markiert.

Diese Option gilt nur für Listen in hierarchischer Form oder für Registerkarten mit jeweils unterschiedlicher Aktion.

Hinweis: Ist die Liste einer Combobox zugeordnet (siehe **Combo Boxen**), sind die Elemente immer änderbar.

Hierarchische Liste

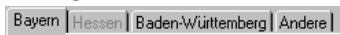
Eine Liste kann auch hierarchisch aufgebaut sein. In diesem Fall können Sie für jeden einzelnen Eintrag festlegen, ob der Benutzer ihn bearbeiten kann. Ist ein Listeneintrag änderbar, kann er per Doppelklick oder unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahl taste** auf den Eintrag klicken und ihn ändern. Hierzu ein Beispiel:



Mit der Option **Änderbar durch Anwender** erlauben Sie dem Benutzer, den Eintrag in der hierarchischen Liste zu ändern.

Registerkarte

Haben Sie Registerkarten eine Auswahlliste zugeordnet, entspricht jede Karte einem Eintrag aus der Liste. Sie können jede Karte aktivieren oder deaktivieren. Eine inaktive Karte erscheint im Formular in Grauschrift. Im folgenden Beispiel ist die Registerkarte „Hessen“ inaktiv:



Mit der Option **Änderbar durch Anwender** kann der Benutzer die einem Eintrag zugeordnete Registerkarte verändern.

Icon mit einem Eintrag verknüpfen

Sie können mit einem Element in einer Liste ein Icon verknüpfen. Wird die Auswahlliste in einem rollbaren Bereich angezeigt, erscheint dieses Icon vor dem Wert.

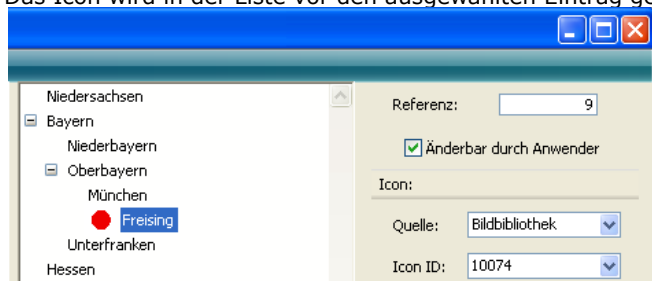
Diese Icons sind in der Bildbibliothek gespeichert.

Hinweis zur Kompatibilität: Die Verwendung von Icons, die auf "Ressourcen" basieren, wird in 4D nicht mehr empfohlen und wird in 64-bit Versionen von 4D nicht unterstützt.

Um ein Icon einem Eintrag zuzuordnen:

1. Wählen Sie die gewünschte Liste und dann den Eintrag aus.
2. Wählen Sie im rechten Bereich im DropDown-Menü **Quelle** die Option **Bildbibliothek** aus.
3. Wählen Sie einen Wert in der Combobox **Icon ID** aus.

Das Icon wird in der Liste vor den ausgewählten Eintrag gesetzt:



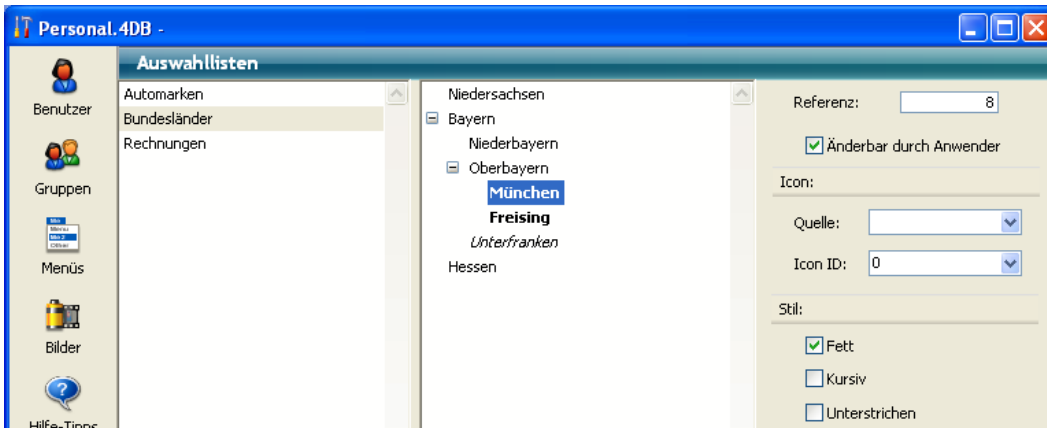
Hinweis: Je nach Größe des Icon müssen Sie u.U. die Höhe der Liste verändern. Weitere Informationen dazu finden Sie im unteren Abschnitt "Mindesthöhe" festlegen.

Wollen Sie die Referenz auf ein Icon löschen, wählen Sie den Wert in der Combobox „Icon-ID“ aus und betätigen Sie die **Rückschritttaste** oder geben Sie 0 ein.

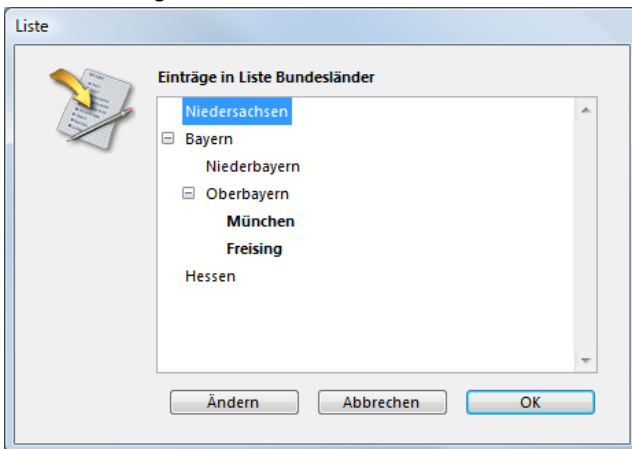
Schrifteigenschaften festlegen

Wenn eine Liste als Auswahlliste verwendet wird, können Sie Einträge fett, kursiv oder unterstrichen anzeigen und auch miteinander kombinieren.

Wählen Sie dazu den Wert in der Liste aus und markieren im rechten Teil unter Stil die Optionen **Fett**, **Kursiv** bzw. **Unterstrichen**. Sie können die Stilarten auch miteinander kombinieren. Nachfolgend sehen Sie ein Beispiel:



Im Anwendungsmodus erscheint die Auswahlliste mit dem definierten Schriftstil:



Eine Standardaktion zuweisen

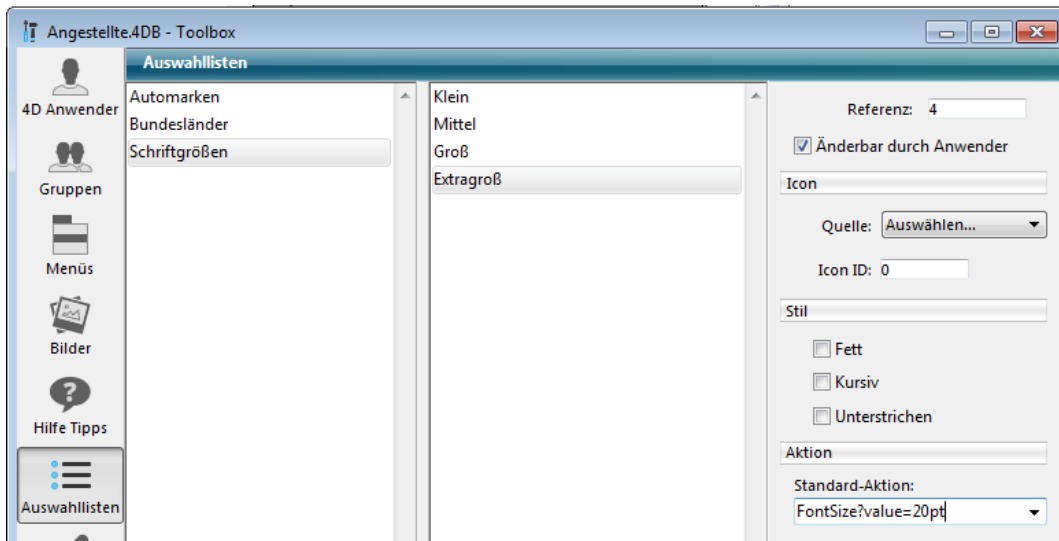
Sie können einem Listeneintrag eine Standardaktion zuweisen (nur erste Ebene), dann wird beim Auswählen durch den Benutzer die Standardaktion in Echtzeit ausgeführt. Weitere Informationen dazu finden Sie im Abschnitt **Standardaktionen**.

Standardaktionen mit Listeneinträgen werden nur unter folgenden Bedingungen berücksichtigt:

- Die Liste ist einem Formularobjekt **PopUp Menü/DropDown-Liste** oder einem **Hierarchischen PopUp-Menü** als Standard Auswahlliste zugewiesen.
- Jede Standardaktion ist über einen String mit einem Wert definiert, z.B. "color?value=#FF0000".
- Der gesamte Satz der Einträge muss dieselbe Art Standardaktion aufrufen und eine eigene Liste mit Werten anzeigen, welche die automatische Liste für die übergeordnete Aktion überschreibt (zugewiesen auf der Ebene der Formularobjekte).

Sie können z.B. eine eigene Liste "Schriftgrößen" mit einem PopUp-Menü mit verschiedenen Schriftgrößen anlegen, um Schriftgrößen in einem Textbereich mit Mehrfachstil oder einem 4D Write Pro Bereich zu verwalten:

1. Sie erstellen die Liste "Schriftgrößen" und weisen jedem Eintrag im Bereich *Standardaktion* eine Standardaktion zu, z.B. "fontSize?value=9pt", "fontSize?value=12pt"...;

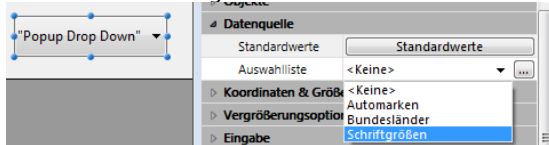


2. Sie weisen die Standardaktion "fontSize" dem Objekt PopUp-Menü/DropDown-Liste auf Formularebene zu:

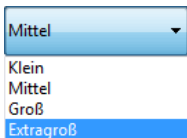


Hinweis: Weisen Sie dem Formularobjekt keine generische Standardaktion zu, können Sie nicht das automatische Aktivieren/Deaktivieren des Eintrags je nach Kontext nutzen.

3. Sie weisen "Schriftgrößen" als Auswahlliste für das Objekt PopUp-Menü/DropDown-Liste zu:



In Echtzeit läuft das PopUp-Menü dann automatisch mit den selbst definierten Größen:

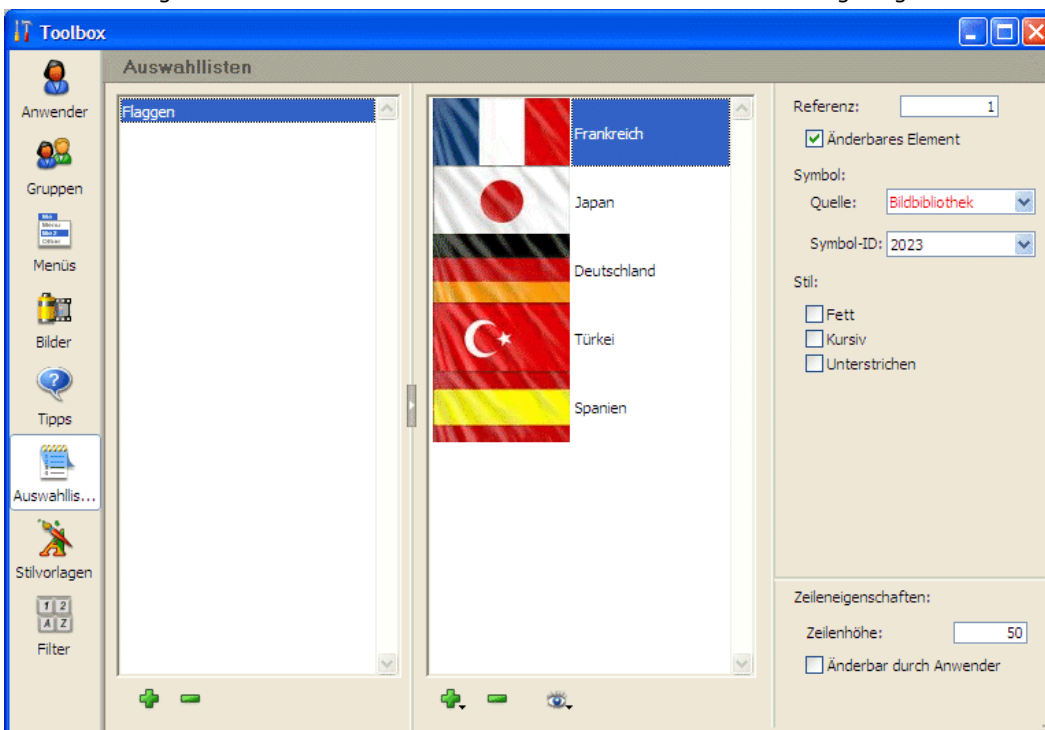


Mindesthöhe festlegen

Zeigt 4D eine Liste als Auswahlliste an, bestimmt die Schriftgröße der Einträge in der hierarchischen Liste den vertikalen Abstand zwischen aufeinanderfolgenden Elementen. Sie können aber auch einen größeren vertikalen Abstand definieren. Das ist besonders zur richtigen Darstellung von Icons notwendig, die den Einträgen zugeordnet sein können. Sie können diese Option auch verwenden, um die Listeneinträge auseinanderzuziehen.

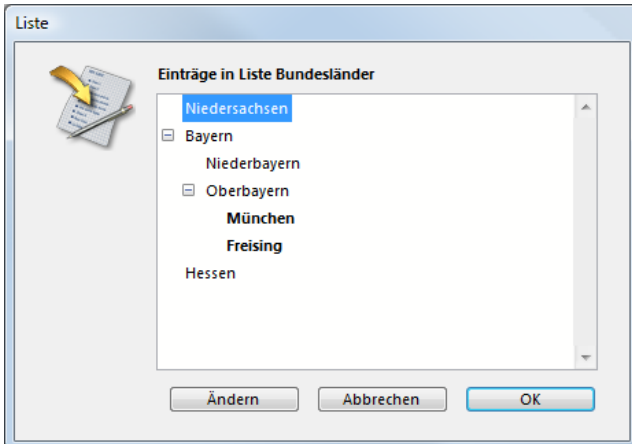
Um eine Mindesthöhe festzulegen, tragen Sie im rechten Bereich unter **Liste-Eigenschaften > Zeilenhöhe** einen Wert in Punkt ein.

Die Auswirkungen dieses Werts werden unmittelbar im mittleren Bereich angezeigt. Hierzu ein Beispiel:

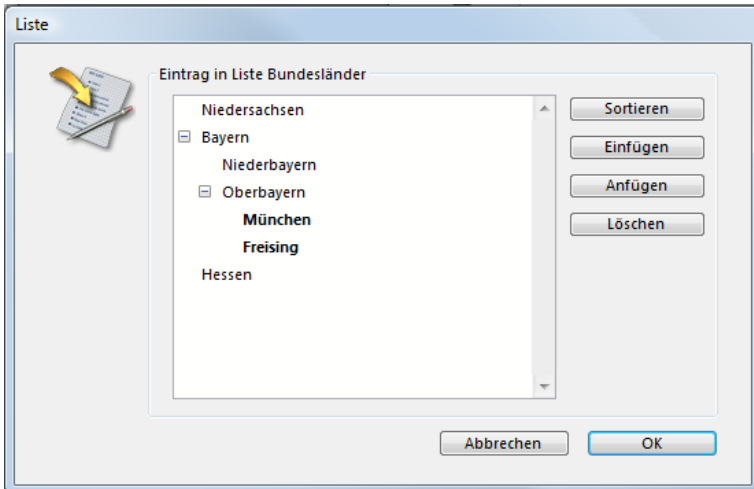


Auswahlliste änderbar machen

Sie können festlegen, ob eine Auswahlliste durch den Benutzer änderbar ist. Eine Liste ist standardmäßig nicht änderbar. Machen Sie eine Liste für den Benutzer änderbar, hat der Benutzer beim Verwenden der Datenbank Zugriff auf einen speziellen Listeneditor. Dieser ermöglicht nur, die zugewiesene Liste zu ändern. Der Benutzer kann weder Listen hinzufügen, Listen löschen noch andere Listen ändern. Ist eine Liste änderbar, ist im Anwendungsmodus die Schaltfläche **Ändern** im Dialogfenster Liste aktiv:

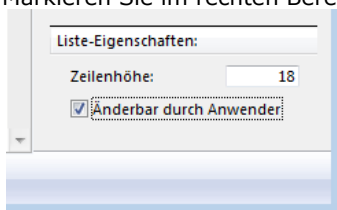


Klickt der Benutzer auf diese Schaltfläche, erscheint der Benutzer-Listeneditor:



Um eine Auswahlliste vom Benutzer änderbar zu machen:





1. Wählen Sie die Liste, die Sie änderbar machen wollen.
2. Markieren Sie im rechten Bereich unter Liste-Eigenschaften die Option **Änderbar durch Anwender**:



Die Liste kann jetzt von Benutzern der Datenbank geändert werden.

Um die Liste für den Benutzer nicht änderbar zu machen, wählen Sie die Liste aus und deaktivieren die Option **Änderbar durch Anwender**.

Stilvorlagen

-  Überblick
-  Stilvorlage erstellen
-  Stilvorlage anwenden
-  Automatische Stilvorlagen

Eine Stilvorlage ist eine Kombination von Schrifteigenschaften — Schriftart, Schriftgröße und Stil. Sie können Stilvorlagen zum Setzen von Schrifteigenschaften für Objekte im Formulareditor verwenden.

Eine Stilvorlage kann für jede von 4D unterstützte Plattformoberfläche spezifische Eigenschaften vorgeben. So kann zum Beispiel eine Stilvorlage für Schaltflächen auf OS X die Schrift Lucida Grande, unter Windows die Schrift Segoe UI verwenden. Schriftgröße und -stil lassen sich ebenfalls für jede Plattform anders definieren.

Stilvorlagen sorgen für eine einheitliche Oberfläche Ihrer Anwendung. Sie haben aber noch weitere Vorteile:

- **Zeitsparende Entwicklung:** Sie können mehrere Einstellungen in einer Operation ausführen.
- **Einfache Pflege und Veränderung:** Stilvorlagen gelten für alle Objekte, die sie verwenden. Verändern Sie z.B. die Schriftgröße einer Stilvorlage, ändert sich die Schrift für alle Objekte, die diese Stilvorlage einsetzen.
- **Steuerbare Entwicklung auf verschiedenen Plattformen:** Setzen Sie eine Stilvorlage ein, verwendet 4D automatisch die Parameter der Plattform, auf welcher das Formular angezeigt wird, wenn für das Objekt als Darstellung *System* eingestellt ist.

Hinweis: Verwendet ein Formularobjekt eine andere Schrift als von der Plattform vorgegeben, wird seine Höhe automatisch so angepasst, dass sie ein Vielfaches der Höhe der Schrift für die aktuelle Plattform ist. Das müssen Sie beim Zeichnen von Objekten berücksichtigen.

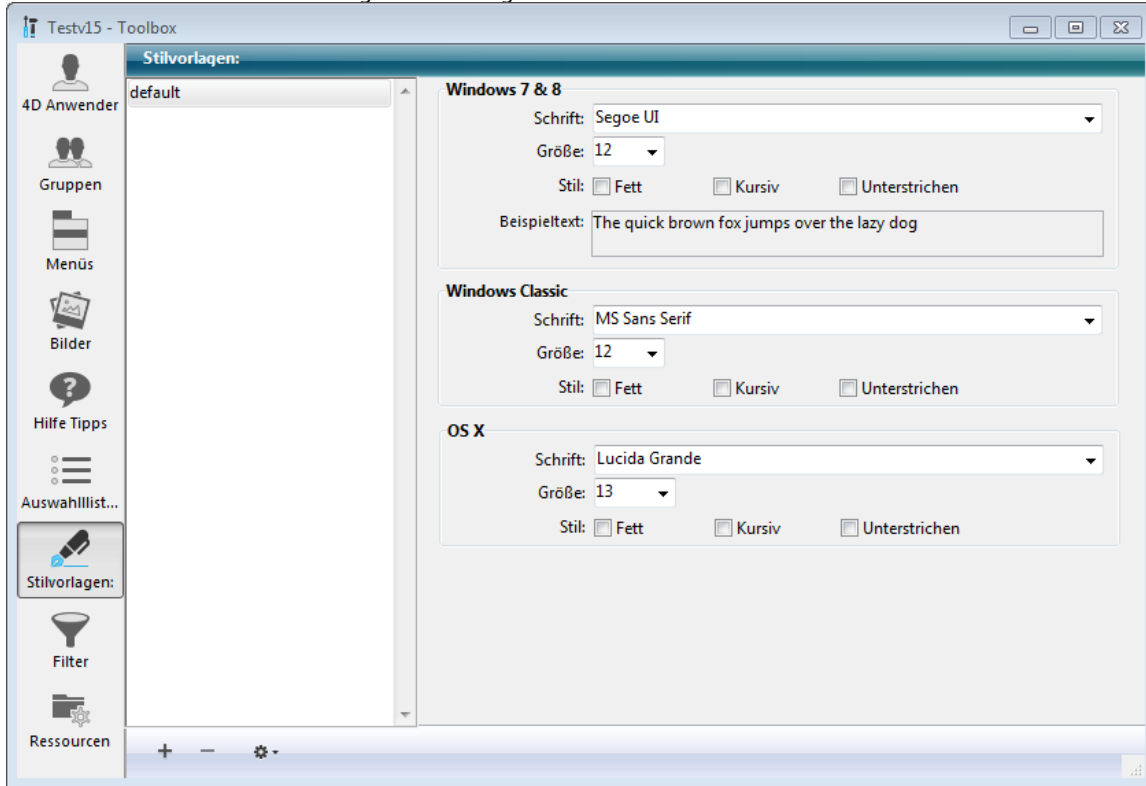
4D bietet einen Satz **Automatische Stilvorlagen**, mit denen Sie Formulare in Übereinstimmung mit den Regeln für die Oberfläche der verschiedenen Plattformen erstellen können.

Stilvorlage erstellen

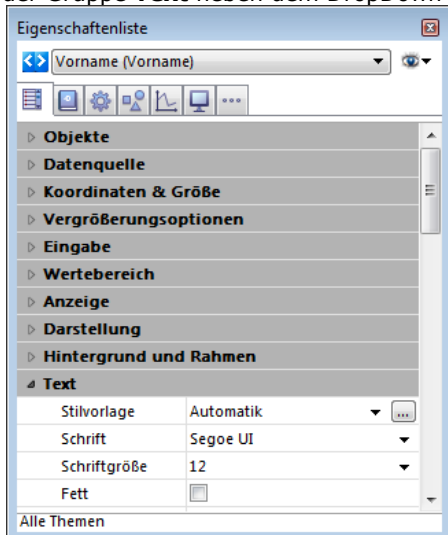
Sie erstellen Stilvorlagen mit dem Stilvorlageneditor in der Toolbox von 4D.
Um eine Stilvorlage anzulegen:

1. Wählen Sie im Menü **Design** den Befehl **Toolbox>Stilvorlagen**

Auf dem Bildschirm erscheint folgendes Dialogfenster:



Sie können dieses Fenster auch im Formulareditor öffnen. Gehen Sie dazu in die Eigenschaftensliste und klicken in der Gruppe **Text** neben dem DropDown-Menü **Stilvorlage** auf die Schaltfläche [...]:



Die Reihenfolge der Anzeige richtet sich nach dem aktuellen Betriebssystem von 4D: Läuft 4D z.B. auf OS X, erscheint oben auf der Seite der OS X Bereich mit dem Beispieltext.

Standardmäßig ist nur die Stilvorlage „Default“ verfügbar. Sie können diese Stilvorlage verändern.

2. Klicken Sie am unteren Rand des Editors auf das Pluszeichen **+**

ODER

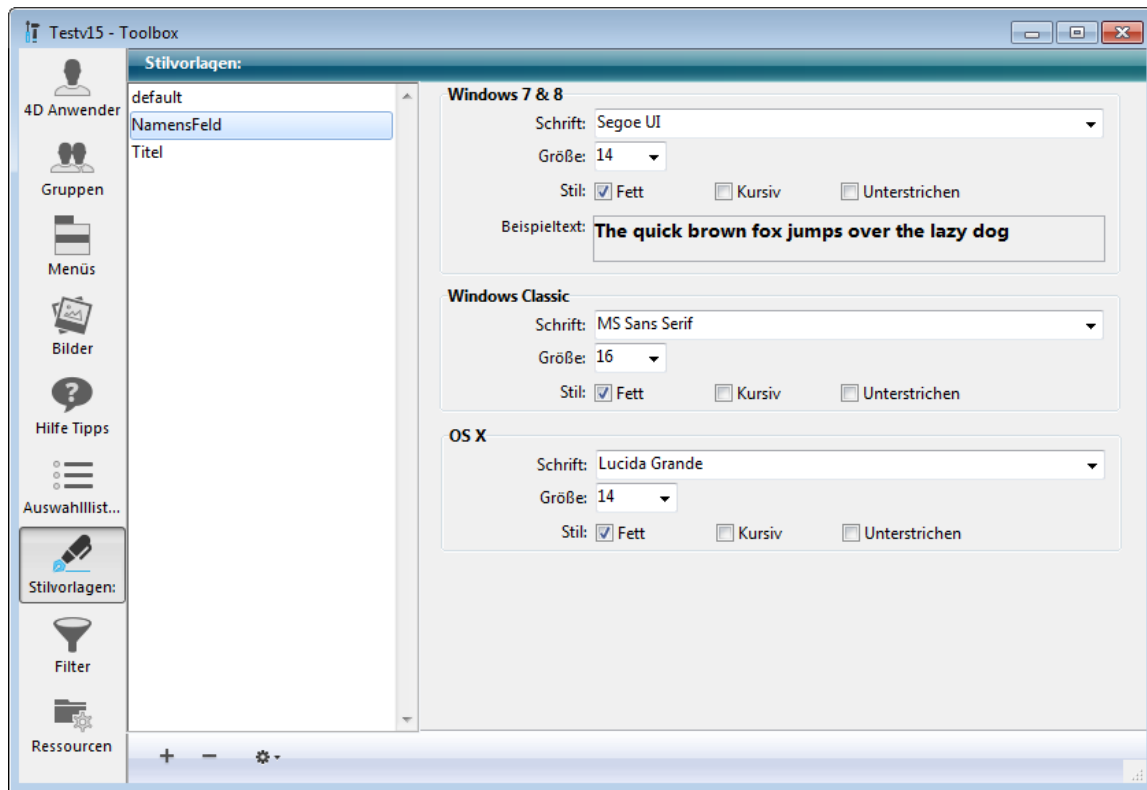
Klicken Sie mit der rechten Maustaste in die Liste der Stilvorlagen und wählen im Kontextmenü den Eintrag **Neu**.

In der Liste der Stilvorlagen erscheint eine neue Stilvorlage mit Namen *Stilvorlage X*.

3. Markieren Sie die Stilvorlage und vergeben einen eigenen Namen.

Sobald der Name bestätigt ist, wird die Liste der Stilvorlagen in alphabetischer Reihenfolge umsortiert.

4. Wählen Sie im rechten Bereich für jede Plattform die gewünschten Optionen für Schrift, Größe und Stil.
Die Änderungen werden automatisch gespeichert. Der Bereich Beispieltext zeigt eine Textprobe



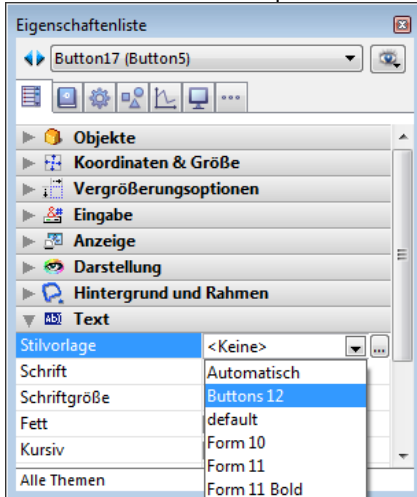
Sie können die Definition einer neuen Stilvorlage auch durch Duplizieren einer bestehenden beginnen. So können Sie gemeinsame Eigenschaften gleich übernehmen. Klicken Sie dazu mit der rechten Maustaste auf den Namen der vorhandenen Stilvorlage und wählen im Kontextmenü den Eintrag **Duplizieren**. Sie können auch die Stilvorlage markieren und im DropDown-Menü unter der Liste den Eintrag **Duplizieren** wählen.

Stilvorlage anwenden

Vordefinierte Stilvorlagen beschleunigen das Anlegen von Datenbanken, da Sie nicht jedes Attribut einzeln zuweisen müssen.

Sie können die angelegten Stilvorlagen in der Eigenschaftensliste des **Formulareditor** aufrufen. Sie lassen sich für jedes Objekt mit Text verwenden, unabhängig ob statisch oder dynamisch, also **Schaltflächen**, **Get list item font**, **Registerkarten**, **Listboxen**, etc.

Um einem Objekt eine Stilvorlage zuzuweisen, wählen Sie das Objekt und dann in der Eigenschaftensliste unter dem Thema **Text** aus der DropDown-Liste den Namen der Stilvorlage aus:



Die gewählte Stilvorlage bestimmt Schriftart, Schriftgröße und Schriftstil der ausgewählten Bezeichnung bzw. des Objekts.

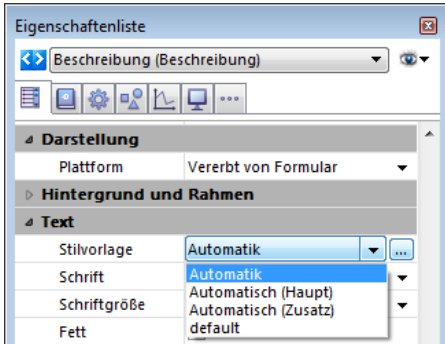
Hinweis: In multi-plattform Kontexten kann die Anwendung von Stilvorlagen mit plattformspezifischen Vorgaben die Höhe von Objekten leicht variieren. Der Grund dafür ist, dass die Objekthöhe automatisch angepasst wird, so dass sie ein Vielfaches der für die aktuelle Plattform angegebenen Schrift ist.

Hinweis: Über die Standardfunktion Suchen/Ersetzen können Sie Stilvorlagen global ersetzen, da sie Referenzen auf Stilvorlagen unterstützt. Weitere Informationen dazu finden Sie im Abschnitt **Suchen und Ersetzen in der Struktur**.

Automatische Stilvorlagen

Beschreibung

In der Eigenschaftsliste sind verschiedene automatische Stilvorlagen verfügbar.



Hinweis: Diese Stilvorlagen sind nur in der Eigenschaftsliste verfügbar. Sie erscheinen nicht in der Liste der Stilvorlagen in der Toolbox.

Im Gegensatz zu anderen Stilvorlagen haben die automatischen Stilvorlagen keine vordefinierten Eigenschaften, sondern bestimmen **Schrift**, **Schriftgröße** und **-farbe** für das Objekt dynamisch gemäß den Parametern des Betriebssystems. Diese richten sich nach:

- der Plattform
- der Sprache des Systems
- und dem Typ des Formularobjekts

Diese automatische Funktionsweise wird bei jeder Verwendung des Formulars implementiert, im Designmodus oder im Anwendungsmodus. Über diese Stilvorlagen ist gewährleistet, dass Titel immer in Einklang mit den aktuellen Oberflächenstandards angezeigt werden. Lediglich die Größe kann von einem Rechner zum anderen variieren.

Automatische Stilvorlagen verwalten **Schrift**, **Schriftgröße** und **-farbe**. Verändern Sie eine dieser Optionen im Formulareditor, arbeitet diese Stilvorlage nicht mehr dynamisch. Die Funktionsweise bleibt dagegen erhalten, wenn Sie eigene Stileigenschaften, wie **fett**, **kursiv** oder **unterstrichen** zuweisen.

Automatik

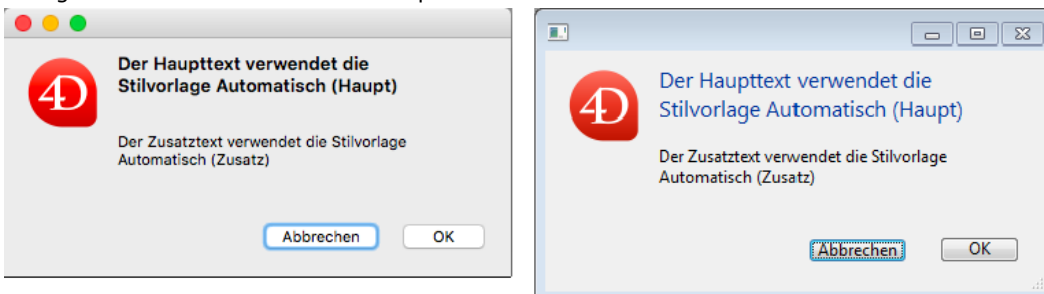
Für ein neu angelegtes Objekt im Formulareditor wird standardmäßig die Stilvorlage **Automatik** angewandt.

Automatisch (Haupt) und Automatisch (Zusatz)

Folgende Objekte unterstützen die Stilvorlagen **Automatisch (Haupt)** und **Automatisch (Zusatz)**:




- Statischer Text
- Felder
- Variablen

Diese Stilvorlagen dienen hauptsächlich zum Gestalten von Dialogboxen. Damit können Sie unterschiedliche Schriftstile für Haupttext und zusätzliche Information in den Fenstern Ihrer Oberfläche verwenden. Hier sehen Sie zwei typische Dialogfenster mit automatischem Haupt- und Zusatztext:



Die Eigenschaften Schrift, Schriftgröße und -farbe werden im Design- oder Anwendungsmodus gemäß Plattform und Sprache des Systems definiert.

Filter und Formate

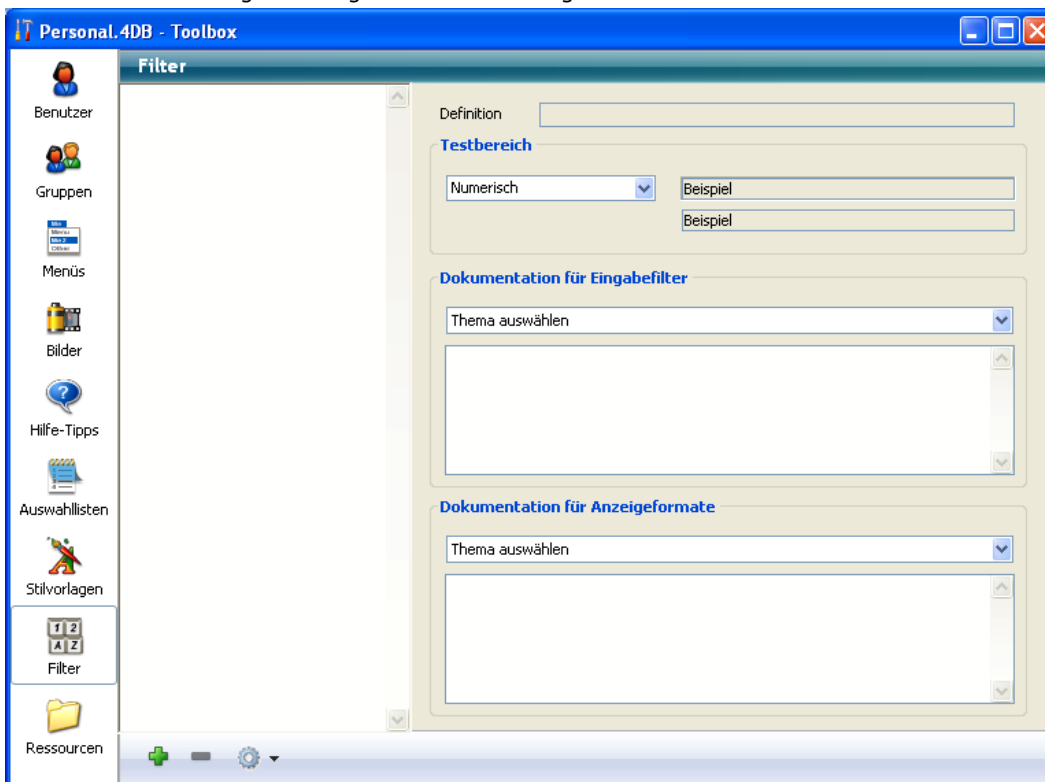
-  Eigene Filter und Formate
-  Format- und Filtereditor
-  Codes für Filter und Formate

Eigene Filter und Formate

4D bietet viele standardmäßige **Anzeigeformate** und Eingabefilter (siehe **Eingabekontrollen**), die Sie zum Konfigurieren der Dateneingabe und zum Anzeigen in Ihren Formularen nutzen können. Bei Bedarf können Sie ein Format oder Filter auch direkt in der Eigenschaftenliste eines Formulars ändern.

Sie können eigene Anzeigeformate und Eingabefilter erzeugen und sich dann über den Namen in jedem Formular darauf beziehen. Eigene Formate und Filter sind sinnvoll, wenn Sie die gleichen Anzeigeformate oder Eingabefilter an verschiedenen Stellen benötigen. Wenn Sie Datenfelder mit dem gleichen Eingabefilter in mehreren Formularen verwenden, können Sie den Eingabefilter einmal erzeugen und darauf verweisen, wo immer Sie diesen brauchen. Außerdem müssen Sie ihn nur einmal ändern, wenn Sie ein Format oder einen Eingabefilter verändern wollen. Er wird dann automatisch bei der Verwendung aktualisiert.

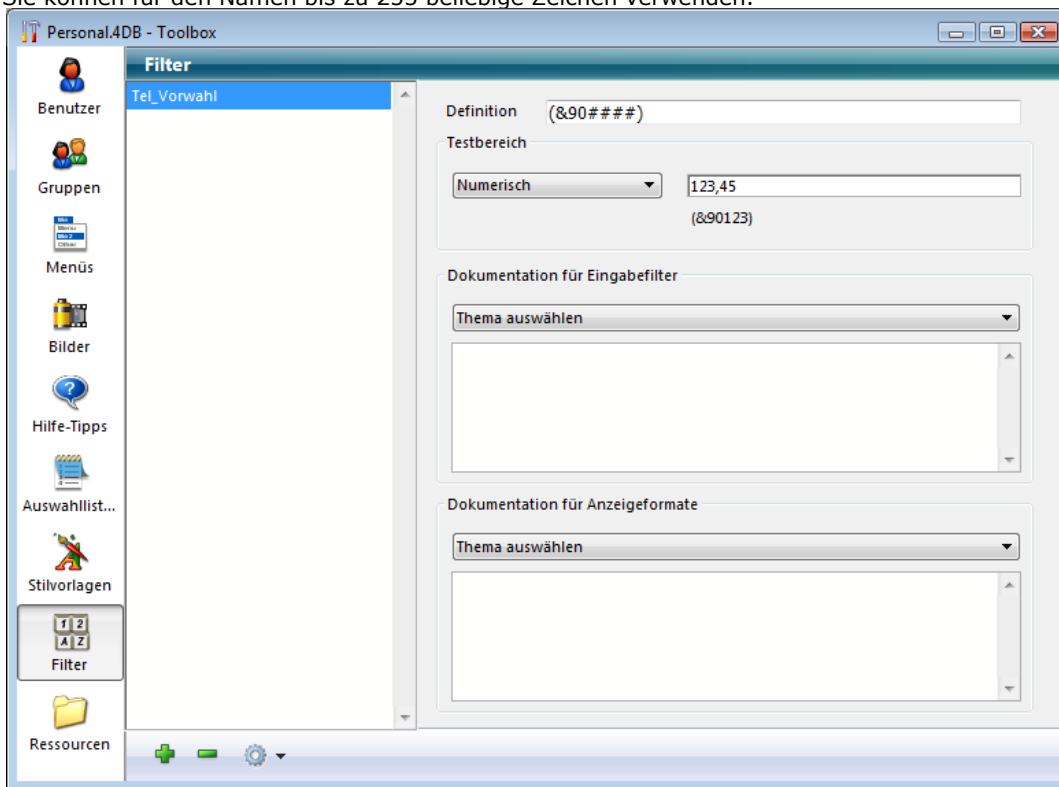
Sie können auch Anzeigeformate erzeugen, die den Eingabefiltern entsprechen und auch Stile zum Installieren verwenden. Sie erzeugen Anzeigeformate bzw. Eingabefilter auf der Seite **Filter** der 4D Toolbox:



Filter oder Format erstellen und setzen

Meist erstellen Sie Filter und Formate als Paar an — einmal für den Eingabefilter und einmal für das Anzeigeformat. Um ein eigenes Format oder einen eigenen Filter anzulegen:

1. Wählen Sie im 4D Menü **Design>Toolbox > Filter und Formate** aus.
ODER
klicken Sie in der Eigenschaftensliste neben dem DropDown-Menü zum Auswählen eines Filters auf die Schaltfläche [...] Auf dem Bildschirm erscheint die Seite **Filter** (standardmäßig leer).
2. Klicken Sie am unteren Rand auf das Pluszeichen +
ODER
Wählen Sie im Kontextmenü des Listenbereichs den Befehl **Hinzufügen** (rechter Mausklick).
Im Listenbereich erscheint ein neuer Eintrag mit Namen „FilterX“.
3. Geben Sie den Namen für das Format oder den Filter ein.
Wollen Sie den Namen später ändern, klicken Sie unter Windows bei gedrückter **Alt-Taste**, auf Mac OS bei gedrückter **Wahltaste** auf den Namen des Formats bzw. Filters, damit er editierbar wird.
Sie können für den Namen bis zu 255 beliebige Zeichen verwenden.



Hinweis: Integrieren Sie in die Bezeichnungen die Namen „Filter“ und „Format“, damit Sie beide voneinander unterscheiden können.

4. Geben Sie im Bereich *Definition* den Code für das Format oder den Filter ein.
 - Wollen Sie zum Beispiel ein Format für eine lokale Telefonnummer erstellen, schreiben Sie: ##-##-##
 - Wollen Sie zum Beispiel einen Eingabefilter für die Artikelnummer mit dem Format XA-654-1 erzeugen, schreiben Sie: !X&"A-Z"##-!0&"0-9"###-#
Das entsprechende Anzeigeformat lautet "##-##-#."

Weitere Informationen dazu finden Sie im Abschnitt **Codes für Filter und Formate**.

Hinweis: Sie können diesen Bereich auch füllen, wenn Sie im unteren Bereich des Fensters auf ein Beispiel doppelklicken. Weitere Informationen dazu finden Sie im unteren Abschnitt "Beispiele verwenden".

5. Wollen Sie weitere Filter bzw. Formate erzeugen, klicken Sie am unteren Rand auf das Pluszeichen + wählen Sie im Kontextmenü des Listenbereichs den Befehl **Hinzufügen** (rechter Mausklick).
ODER
Wollen Sie Filter bzw. Formate erstellen, die auf bereits vorhandenen basieren, wählen Sie im Kontextmenü des Listenbereichs oder im DropDown-Menü am unteren Rand den Eintrag **Duplizieren**.

Sie können jeden Filter oder jedes Format durch Auswählen und Ändern des Namens oder Codes bearbeiten. Sie können jeden Stil markieren und durch Klick auf das Minuszeichen - am unteren Rand oder über den Befehl **Löschen** im Kontextmenü des Editors löschen.

Formate oder Filter, die Sie über den Filter- und Formateditor der Toolbox selbst angelegt haben, werden automatisch mit einem senkrechten Strich (|) vor dem Eintrag gekennzeichnet (siehe **Eingabekontrollen** und **Anzeigeformate**). Ein eigenes Format können Sie auf dieselbe Weise wie ein vorgegebenes Format auswählen.

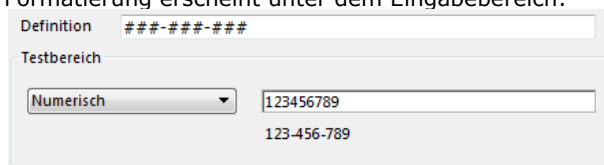
Testbereich verwenden

Im Testbereich können Sie das Format bzw. den Filter prüfen.

Zuerst müssen Sie im DropDown-Bereich den Datentyp bestimmen, für den der Filter bzw. das Format gelten soll: Numerisch, Alphanumerisch, Datum oder Zeit.

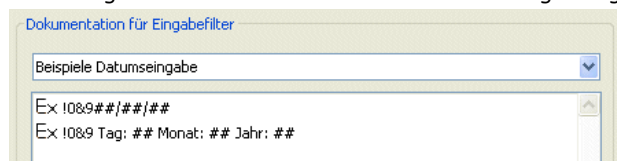
Dann geben Sie im dazugehörigen Bereich einen Testwert ein:

- Eingabefilter prüfen die Zeichen bei der Eingabe
- Bei Anzeigeformaten drücken Sie nach der Eingabe die **Eingabetaste** bzw. die **Zeilenschaltung**. Das Ergebnis der Formatierung erscheint unter dem Eingabebereich:



Beispiele verwenden

Die Bereiche **Dokumentation für Eingabefilter/Anzeigeformate** enthalten jeweils ein DropDown-Menü und einen Anzeigebereich. Hier können Sie ein Thema wählen, über das Sie Beispiele und Informationen erhalten möchten. Im Anzeigebereich darunter erscheinen die dazugehörigen Beispiele mit Erläuterungen:



Doppelklicken Sie auf ein Beispiel, wird es direkt im Bereich Definition eingefügt.

Eingabefilter:

- *Führende Anzeige-Zeichen*: Ändern von Platzhaltern
- *Führende Zeichen*: Auslöser für Filterzeichen
- *Start Codes*: Startcodes für Filter
- *Abkürzungen*: Filterkürzel
- *Platzhalter*: Zeichen, die Platz reservieren
- *Stumme Zeichen*: Zeichen, die in Filtern ignoriert werden
- *Beispiele Datumseingabe*
- *Beispiele Zeiteingabe*
- *Beispiele Bankleitzahl-Eingabe*
- *Beispiel US-Sozialversicherungsnummer*
- *Beispiele andere Eingabefilter*

Anzeigeformate:

- *Platzhalter*: Zeichen, die Platz reservieren
- *Stumme Zeichen*: Zeichen, die in Formaten ignoriert werden
- *Beispiele Datumsanzeige*
- *Beispiele Zeitanzeige*
- *Beispiele Bankleitzahlen-Anzeige*
- *Kreditkartennummer*

Vorbemerkung: Dieser Abschnitt beschreibt die Zeichen zum Definieren von alphanumerischen Eingabefiltern und Anzeigeformaten. Die Zeichen für numerische Anzeigeformate werden im Abschnitt **Anzeigeformate** beschrieben. Ein **Eingabefilter** besteht aus drei Teilen in folgender Reihenfolge:

Initiator "Argument" Platzhalter

- Der Initiator weist 4D an, die nachfolgenden Zeichen bei der Dateneingabe in ein Feld als Eingabefilter zu verwenden
- Argumente definieren die erlaubten Zeichen
- Platzhalter definieren die für die Zeichen verfügbaren Stellen

Zum Beispiel lässt folgender Eingabefilter nur zu, dass die Buchstaben "a," "b," "c" oder "g" zwei Stellen eingegeben werden:

```
&"a;b;c;g"##
```

Bei diesem Beispiel ist das kaufmännische Und-Zeichen (&) der Initiator, "a;b;c;g" das Argument und die Nummernzeichen (#) die Platzhalter. Der Filter kann folgendermaßen interpretiert werden: "Lasse die Buchstaben 'a', 'b', 'c' oder 'g' an zwei Stellen zu". Also kann der Benutzer "ag," "gc," "ba," "ab," "aa," "ac" oder andere Kombinationen der vier erlaubten Zeichen eingeben.

Eingabefilter lassen sich auch miteinander kombinieren. Der folgende Eingabefilter lässt nur zu, dass die Buchstaben "a," "b," "c" oder "g" an zwei Stellen, gefolgt von den Zahlen 1, 3 oder 8 n einer Stelle eingegeben werden:

```
&"a;b;c;g"##&"1;3;8"#
```

Der Benutzer muss zwei der erlaubten Buchstaben verwenden, gefolgt von einer der erlaubten Zahlen.

Ein **Anzeigeformat** kombiniert Platzhalter und Standardzeichen.

Zeichen, die einen Filter starten

Zwei Zeichen starten einen Filter: Das kaufmännische Und (&) bzw. die Tilde (~). Diese Zeichen weisen 4D an, das direkt darauf folgende Argument als Filter für die nachfolgenden Platzhalter zu verwenden.

Die Tilde (~) weist 4D zusätzlich an, alle Buchstaben großzuschreiben. Es verhindert nicht das Eintippen von Kleinbuchstaben; es wandelt sie lediglich in Großbuchstaben um.

Die folgenden Eingabefilter haben dieselbe Auswirkung:

```
&"P"#  
~"P"#
```

Der Unterschied ist, dass der Filter mit dem kaufmännischen Und kein kleingeschriebenes "p" zulässt. Der Filter mit der Tilde (~) akzeptiert das kleingeschriebene "p", konvertiert es jedoch in Großbuchstaben.

Da keine Buchstaben verwendet werden, sind die folgenden Eingabefilter gleich:

```
&"1;5;8"#  
~"1;5;8"#
```

Argumente

Ein Filterargument folgt auf den Initiator und definiert die Zeichen, die in den nachfolgenden Platzhaltern erlaubt sind. Um ein Filterargument zu erzeugen, müssen Sie die gültigen Zeichen zwischen Anführungszeichen setzen.

Argumente bestehen aus Kleinbuchstaben, Großbuchstaben, Zahlen, Satzzeichen und Sonderzeichen (!@#%&*()[]":';?><.,/\`~). Setzen Sie einen Kleinbuchstaben im Argument, kann der Benutzer nur die Kleinbuchstabenform verwenden. Setzen Sie einen Großbuchstaben, kann er nur die Großbuchstabenform verwenden.

- Ein Argument kann ein einzelnes Zeichen sein (ein Buchstabe oder eine Zahl), zum Beispiel "j," "J" oder "6."
- Ein Argument kann eine Reihe von Zeichen sein, getrennt durch ein Strichpunkt, zum Beispiel "a;r;t" oder "1;5."
- Ein Argument kann Zeichenbereiche enthalten. Ein Bereich wird durch das erste Zeichen, einen Strich und das letzte Zeichen bestimmt. Beispiele hierfür sind "a-c" und "1-5." Das Argument "a-c" ist dasselbe wie "a;b;c" und "1-5" ist dasselbe wie "1;2;3;4;5."
- Ein Argument kann einzelne Buchstaben, einzelne Zahlen und einen oder mehrere Bereiche enthalten, zum Beispiel "a;m-z;3;5-9."

Die folgende Tabelle zeigt hilfreiche Abkürzungen für Argumente. Sie werden in Filtern *ohne Anführungszeichen* verwendet:

Zeichen	Bedeutung	Entsprechung
9	Erlaube Zahlen	"0-9"
a	Erlaube Klein- und Großbuchstaben	"a-z;A-Z"
A	Erlaube Großbuchstaben	"A-Z"
@	Erlaube alphanumerische Zeichen	"a-z;A-Z;0-9"

Folgende Eingabefilter sind gleichbedeutend:

&9#
&"0-9"#
&"1;2;3;4;5;6;7;8;9;0"#

Folgende Eingabefilter sind gleichbedeutend:

&a#
&"a-z;A-Z"#

Folgende Eingabefilter sind gleichbedeutend:

&A#
&"A-Z"#

Platzhalter

Das Nummernzeichen (#) ist der einzige Platzhalter für alphanumerische Filter und Formate (für numerische Filter und Formate sind weitere Zeichen zulässig). Sie verwenden ein Nummernzeichen für jedes Zeichen, das der Benutzer im Datenfeld eingeben kann.

Der folgende Eingabefilter erlaubt dem Benutzer, Buchstaben an vier Stellen einzugeben:

&a####

Der folgende Eingabefilter erlaubt dem Benutzer Großbuchstaben an drei Stellen, gefolgt von Zahlen an zwei Stellen einzugeben:

&A###&9##

Geben Sie keine Platzhalter an, erlaubt der Filtercode beliebig viele Zeichen. Folgender Eingabefilter erlaubt dem Benutzer nur, Zahlen einzugeben, jedoch in unbeschränkter Länge:

&9

Hinweis: Im Struktureditor können Sie die maximale Anzahl zugelassener Zeichen in einem alphanumerischen Datenfeld festlegen (siehe **4D Datenfeldtypen**).

Anzeigezeichen

Wenn ein Datenfeld mit einem Eingabefilter zur Dateneingabe ausgewählt wird, zeigt 4D einen Unterstrich (_) für jeden Platzhalter. Wenn der Benutzer einen gültigen Wert eingibt, wird der jeweilige Unterstrich hervorgehoben und durch das eingegebene Zeichen ersetzt.

Sie können 4D anweisen, welches Zeichen für den Unterstrich gesetzt werden soll, wenn Sie den Eingabefilter mit einem Ausrufezeichen (!) und dem entsprechenden Zeichen beginnen.

Sie können anstelle des Unterstrichs auch ein anderes Zeichen verwenden. Wenn Sie zum Beispiel "XXXX" anzeigen lassen und der Benutzer gibt nur zwei der erlaubten Zeichen ein (zum Beispiel "AA"), zeigt das Datenfeld nach dem Speichern "AAXX".

Die folgende Abbildung zeigt ein ausgewähltes Datenfeld mit Null und Unterstrichen.

Vorname :	Nachname :	Gehalt :
Anton	Schmidt	1500
Michael	Schwarz	2100
Vinzenz	Lang	1800




Tote Zeichen

Alle Zeichen, Satzzeichen und Leerzeichen können als tote Zeichen verwendet werden. Tote Zeichen werden bei der Dateneingabe angezeigt, jedoch von der Eingabemarke übersprungen und nicht als Teil der Daten eingegeben.

Die Zeichen, die Sie als tote Zeichen verwenden wollen, werden vor, nach und zwischen Platzhaltern eingefügt. Sie erscheinen zur Klarheit während der Dateneingabe.

Der Eingabefilter für die Sozialversicherungsnummer (&9###-##-####) verwendet Bindestriche als tote Zeichen. Hat der Benutzer die Stelle unmittelbar vor einem toten Zeichen eingegeben, springt die Eingabemarke zum ersten Zeichen nach dem toten Zeichen.

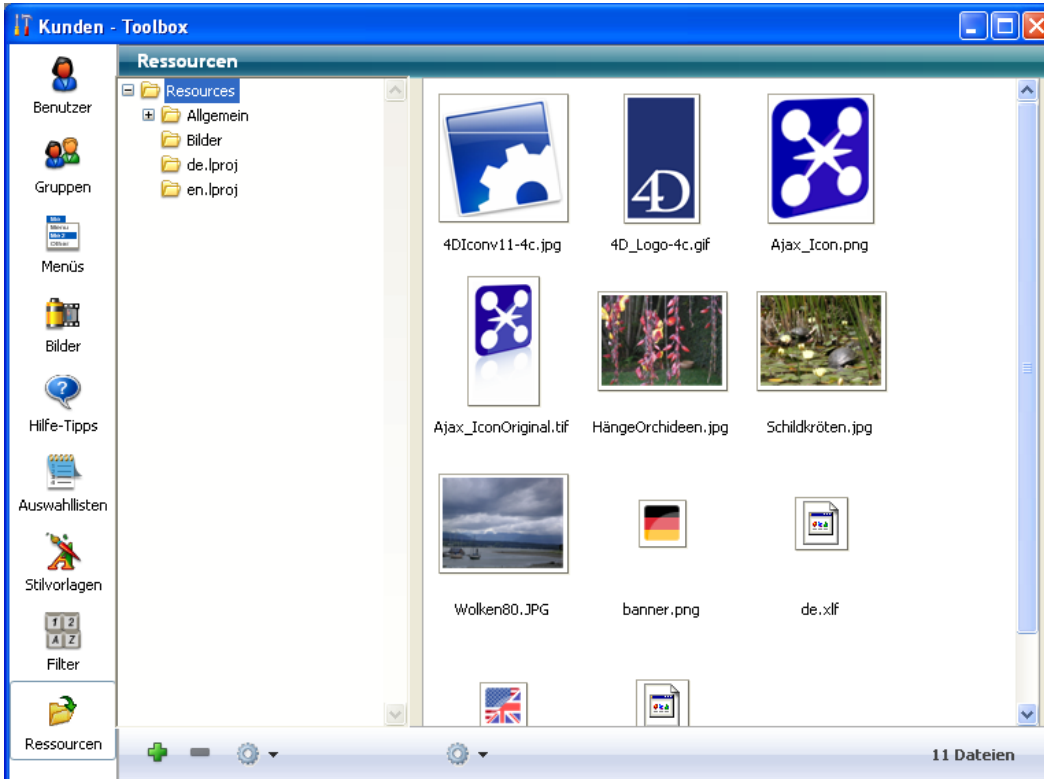
Ressourcen Explorer

-  Überblick
-  Ressourcen Explorer verwenden
-  Dynamische Synchronisation im Client-Server Betrieb

Überblick

Ab 4D v11 SQL basiert die Ressourcen Architektur auf dem Ordner **Resources**, der neben der Strukturdatei der Datenbank liegt (.4db oder .4dc, siehe **Beschreibung der 4D Dateien**). In diesem Ordner können Sie alle Ressourcen bzw. Dokumente einer Anwendung speichern. Das sind Dateien zum Lokalisieren, zum Anpassen der Oberfläche (Bilddateien, Textdateien, XLIFF Dateien, etc.) oder für Operationen ganz allgemein.

Um die Verwaltung dieses Ordners in einer Client-/Server Architektur zu vereinfachen, wird mit 4D Version 11.2 das Tool Ressourcen Explorer eingeführt. Sie können es über die 4D Tool Box aufrufen:



Der Ressourcen Explorer zeigt den Inhalt des Ordners **Resources** der aktuellen Datenbank als hierarchische Liste an. Er enthält auch weitere Funktionen zum Verwalten des Ordnerinhalts, wie Einträge hinzufügen und löschen, Vorschau, etc.

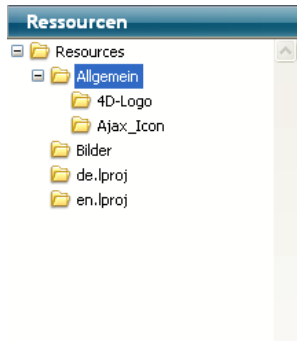
Im Remote-Modus ist der Ressourcen Explorer besonders hilfreich: Denn er ermöglicht, den Inhalt des Ordners **Resources** zwischen allen Rechnern, die an den Server angemeldet sind, aufeinander abzustimmen, d.h. über den Ressourcen Explorer können Sie die gemeinsame Nutzung der Ressourcen in der Client-/Server-Umgebung verwalten. Ein Meldungssystem ermöglicht, die Client-Rechner bei Änderungen im Ordner **Resources** zu informieren. Jeder Client-Rechner kann sich dann selbst mit dem Server synchronisieren.

Ressourcen Explorer verwenden





Über den Ressourcen Explorer können Sie den Inhalt des Ordners **Resources** der aktuellen Datenbank ansehen und verändern. Am unteren Rand gibt es Schaltflächen zum Hinzufügen, Löschen, Ansehen oder Suchen von Ressourcen Elementen.

Liste der Ordner

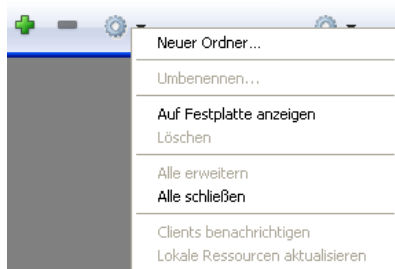
Die Liste zeigt die Hierarchie der Ordner im Ordner **Resources** der Datenbank.



Sie können einzelne Ordner auswählen, auf- und zuklappen. Klicken Sie auf einen Ordnernamen, erscheinen alle darin enthaltenen Dateien in der Vorschau im rechten Teil.

Die Steuerleiste am unteren Rand enthält die Schaltflächen  und  sowie ein DropDown-Menü mit Aktionen. Die Schaltfläche  legt im ausgewählten Ordner bzw. auf der obersten Ebene einen neuen Ordner an. Die Schaltfläche  löscht den ausgewählten Ordner, sowie seinen Inhalt.

Sie können in der Ordnerliste auch das Kontextmenü aufrufen. Es enthält dieselben Befehle wie das DropDown-Menü.



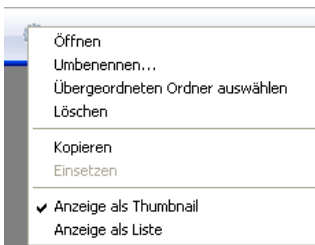
Das DropDown-Menü enthält folgende Optionen:

- **Neuer Ordner...:** Erstellt einen Ordner im ausgewählten Ordner bzw. auf der obersten Ebene, wenn kein Ordner ausgewählt ist. Auf dem Bildschirm erscheint ein Dialogfenster, um den Namen des Ordners einzutragen. Da dieser Ordner tatsächlich auf der Festplatte angelegt wird, müssen Sie darauf achten, dass der Name keine vom System nicht-erlaubten Zeichen enthält (: oder /).
- **Umbenennen...:** Öffnet ein Dialogfenster, um den ausgewählten Ordner umzubenennen.
Hinweis: Der Ordner **Resources** selbst lässt sich nicht umbenennen.
- **Auf Festplatte anzeigen** (nur im lokalen Modus aktiv): Zeigt den Ordner in einem Fenster des Betriebssystems.
- **Löschen:** Löscht den ausgewählten Ordner samt seinem Inhalt.
Hinweis: Der Ordner **Resources** selbst lässt sich nicht löschen.
- **Alle erweitern/Alle schließen:** Klappt alle Ordner in der Liste auf oder zu.
- **Clients benachrichtigen** (nur im Remote-Modus aktiv): Damit veranlassen Sie, dass die anderen Client-Rechner eine Meldung erhalten, dass der Inhalt des Ordners **Resources** aktualisiert werden muss. Die Clients können dann, je nach Konfiguration, den eigenen Ordner **Resources** sofort oder später synchronisieren (siehe Abschnitt **Update der Client Rechner**).
Verwenden Sie diesen Befehl, wenn Sie im Ordner **Resources** Änderungen gemacht haben und die anderen Clients unmittelbar synchronisiert werden sollen.
- **Lokale Ressourcen aktualisieren** (nur im Remote-Modus aktiv): Damit veranlassen Sie, dass der lokale Ordner **Resources** mit dem Ordner auf dem Server Rechner synchronisiert wird.
Verwenden Sie diesen Befehl, wenn Sie die Meldung erhalten, dass der Ordner **Resources** auf dem Server Rechner aktualisiert wurde (siehe Abschnitt **Meldung für Client-Rechner** und die Synchronisation der Ordner nicht automatisch erfolgt).
Um den gesamten Inhalt des lokalen Ordners zu aktualisieren (Download des Server Ordners und Ersetzen des lokalen Ordners), wählen Sie den Befehl **Lokale Ressourcen aktualisieren** mit gedrückter **Umschalttaste** aus.

Vorschau

Der rechte Teil des Ressourcen Explorers zeigt die Vorschau auf die Dateien und Unterdateien im ausgewählten Ordner. Wird ein anderer Ordner in der Liste ausgewählt, ändert sich auch die Vorschau.

Über das DropDown Menü unter der Vorschau können Sie verschiedene Aktionen ausführen:

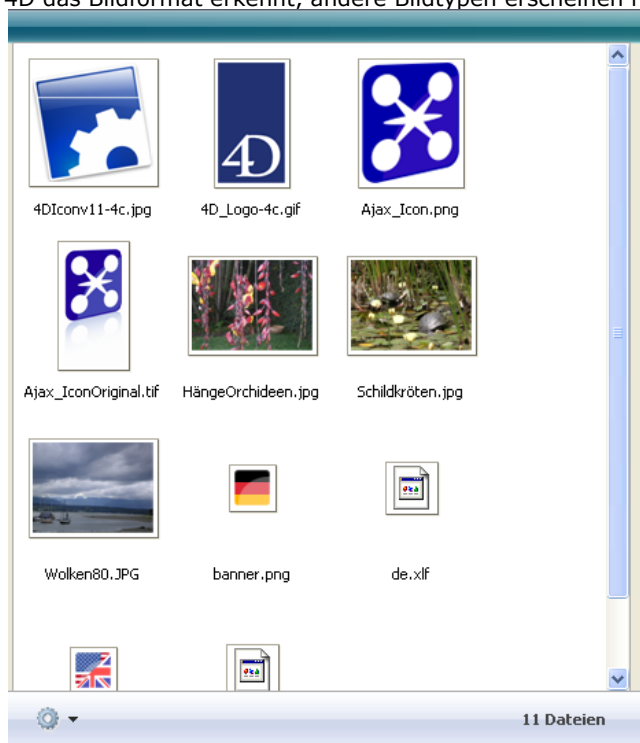


- **Öffnen:** Öffnet die standardmäßig im Editor ausgewählte Datei - sofern vorhanden. Sie können auch auf die Datei doppelklicken, um sie zu öffnen.
- **Umbenennen...:** Öffnet ein Dialogfenster, in dem Sie die ausgewählte Datei umbenennen können.
- **Übergeordneten Ordner auswählen:** Wählt in der Ordnerliste den übergeordneten Ordner zur ausgewählten Datei. So können Sie den genauen Speicherort der Datei innerhalb der hierarchischen Struktur des Ordners **Resources** sehen. Die Vorschau zeigt alle Dateien innerhalb des ausgewählten Ordners, sowie die Dateien in den Unterordnern.
- **Löschen:** Löscht die ausgewählte Datei.
- **Kopieren:** Kopiert die ausgewählte Datei in die Zwischenablage.
- **Einsetzen:** Setzt den Inhalt der Zwischenablage in den ausgewählten Ordner, wenn in der Zwischenablage ein Bild oder der Pfadname einer Datei liegt. Bei einem Bild erstellt 4D eine Bilddatei vom entsprechenden Typ. Auf dem Bildschirm erscheint ein Dialogfenster, um der angelegten Datei einen Namen zu geben.

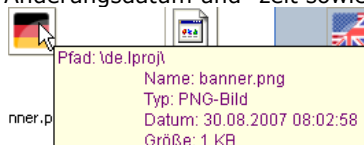
Anzeige als Thumbnail / Liste

Diese Befehle erscheinen abwechselnd. Sie bestimmen den aktuellen Anzeigemodus des Bereichs.

- Bei **Anzeige als Thumbnail** erscheinen die Dateien als Thumbnails. Die Bilder erscheinen als Miniaturansicht, wenn 4D das Bildformat erkennt, andere Bildtypen erscheinen mit dem Icon des Systems.



Klicken Sie mit der rechten Maustaste in die Vorschau, öffnet sich ein Kontextmenü mit Befehlen zum Bearbeiten. Ziehen Sie die Maus über ein Thumbnail, erhalten Sie weitere Informationen wie Pfadname (relativ), Name, Typ, Änderungsdatum und -zeit sowie Größe.



- Bei **Anzeige als Liste** erscheinen die Dateien der gewählten Ordner als Tabelle. Jede Tabelle zeigt Name, Änderungsdatum und -zeit, Größe und Typ der einzelnen Dateien:

Name	Änderung	Größe	Typ
4DIconv11-4c.jpg	18.01.2008 08:56:14	46 KB	JPEG-Bild
4D_Logo-4c.gif	14.01.2008 13:05:03	2 KB	GIF-Bild
Ajax_Icon.png	19.01.2007 09:34:25	4 KB	PNG-Bild
Ajax_IconOriginal.tif	06.11.2006 09:47:42	1007 KB	TIF-Datei
HängeOrchideen.jpg	13.12.2006 08:27:40	284 KB	JPEG-Bild
Schildkröten.jpg	13.12.2006 08:30:53	277 KB	JPEG-Bild
Wolken80.JPG	12.08.2007 21:23:00	244 KB	JPEG-Bild
banner.png	30.08.2007 08:02:58	1 KB	PNG-Bild
de.xlf	08.10.2007 12:58:04	6 KB	XLF-Datei
banner.png	30.08.2007 08:02:58	1 KB	PNG-Bild
en.xlf	08.10.2007 12:57:59	6 KB	XLF-Datei

11 Dateien

Sie können in die Titelleiste einer Spalte klicken, um die Tabelle zu sortieren.

Verwaltung per Drag-and-Drop

Im Ressourcen Explorer können Sie Elemente auch per Drag-and-Drop im Ordner **Resources** und seinen Unterordnern hinzufügen:

- Sie können die Ordner innerhalb der Liste per Drag-and-Drop versetzen, um die Hierarchie im Ordner **Resources** zu verändern.
- Sie können auch Dateien vom Desktop des Betriebssystems per Drag-and-Drop hinzufügen. 4D erstellt im Ordner **Resources** automatisch eine Kopie der übertragenen Datei. Sie können die Dateien in die Liste der Ordner oder direkt in die Vorschau ziehen.
- Sie können Ordner per Drag-and-Drop in den Ordner **Resources** ziehen. Sie können vom Desktop des Betriebssystems oder vom Ordner **Resources** einer anderen 4D Anwendung stammen. Ordner aus einer anderen 4D Anwendung werden verschoben, Ordner vom Desktop werden, analog zu Dateien, kopiert. Sie können einen Ordner in die Liste der Ordner oder direkt in die Vorschau ziehen. In der Vorschau wird der Ordner in der hier angezeigten Ebene abgelegt.
- Sie können Bilder aus dem Ordner **Resources** per Drag-and-Drop in den Formulareditor oder Methodeneditor ziehen. Das Bild wird dann als Referenz eingefügt.

Dynamische Synchronisation im Client-Server Betrieb

Der Ressourcen Explorer vereinfacht die Team-Entwicklung in der Client/Server Umgebung. Hauptziel ist dabei, den Inhalt des Ordners **Resources** in Echtzeit auf allen angeschlossenen Rechnern zu synchronisieren.

Es gibt folgende Schritte:

1. Bei Änderung auf einem Client Rechner wird der Ordner **Resources** auf dem Server aktualisiert
2. Alle angeschlossenen Clients erhalten eine Meldung
3. Die angeschlossenen Clients werden unmittelbar aktualisiert

Hinweis: Für nicht angemeldete Clients wird der Ordner **Resources** automatisch bei der Anmeldung synchronisiert.

Update des Server

Jede Änderung, die über den Ressourcen Explorer auf lokaler Ebene ausgeführt wird, wird automatisch auf den Server übertragen.

Meldung für Client-Rechner

Die angeschlossenen Client-Rechner erhalten eine Meldung, dass der Inhalt des Ordners **Resources** geändert wurde:

- Automatisch vom Server, zwei Minuten nach der letzten Änderung durch einen Client. Die Zeitverzögerung sorgt dafür, dass beim Übertragen mehrerer Dateien nicht sofort eine Meldung erscheint.
- Manuell nach einer Änderung auf dem Client-Rechner. Rufen Sie dazu im Ressourcen Explorer dieses Clients im DropDown-Menü am unteren Rand den Befehl **Clients benachrichtigen** auf (siehe **Ressourcen Explorer verwenden**).
- Oder per Programmierung über den 4D Befehl **NOTIFY RESOURCES FOLDER MODIFICATION**. Er ist hilfreich, wenn der Inhalt des Ordners **Resources** auf dem Server Rechner über eine Serverprozedur geändert wird.

Auf den Clients hängt die Meldung bei Änderungen von der jeweiligen Einstellung ab (siehe nächster Abschnitt).

Update der Client Rechner

Wurde gemeldet, dass sich der Ordner **Resources** auf dem Server geändert hat, kann jeder Client-Rechner selbst synchronisieren. Das kann automatisch oder manuell sein, je nach den Voreinstellungen, die global für die Anwendung oder spezifisch für den Client-Rechner definiert wurden.

Sie können die Konfiguration global über die Datenbank-Eigenschaften oder über den 4D Befehl **SET DATABASE PARAMETER** individuell pro Client-Rechner und pro Arbeitssitzung setzen.





Den Update-Modus legen Sie in den Datenbank-Eigenschaften auf der **Seite Client-Server/Netzwerk** mit der Option *'Resources'-Ordner während einer Session aktualisieren* fest.

Es gibt drei Möglichkeiten:

- **Nie:** Der lokale Ordner **Resources** wird während der Arbeitssitzung nicht aktualisiert. Die vom Server gesendete Meldung wird ignoriert. Der lokale Ordner **Resources** lässt sich bei Bedarf manuell über den Menübefehl **Lokale Ressourcen aktualisieren** aktualisieren.
- **Immer:** Der lokale Ordner **Resources** wird während der Arbeitssitzung automatisch synchronisiert, wenn der Server eine Meldung sendet.
- **Fragen:** Sendet der Server eine Meldung, erscheint auf den Client-Rechnern ein Dialogfenster, das die Änderung meldet. Der Benutzer kann dann die Synchronisation des lokalen Ordners **Resources** bestätigen oder ablehnen.

Hinweis: Wird die Konfiguration in den Einstellungen auf dem Server festgelegt, gilt sie für alle Client-Rechner. Wird sie auf einem Client-Rechner definiert, gilt sie nur für diesen Rechner.

Datensätze verwalten

-  Zwischen Tabellen und Formularen wechseln
-  Datensätze anzeigen und auswählen
-  Datensätze bearbeiten
-  Datensätze drucken

Zwischen Tabellen und Formularen wechseln

In einer 4D-Datenbank speichern Sie Ihre Daten in Tabellen. Jede Tabelle enthält bestimmte Informationen. Eine Adressdatenbank kann beispielsweise aus einer Tabelle mit Informationen zu Personen sowie einer weiteren Tabelle mit Angaben zu Firmen bestehen.

Sie verwenden Formulare, um Daten einzugeben und zu verwalten. Jede Tabelle Ihrer Datenbank hat ein aktuelles Eingabe- und Ausgabeformular. Das Eingabeformular zeigt jeweils einen Datensatz an. Im Eingabeformular geben Sie die Informationen eines einzelnen Datensatzes ein, sehen sich die Angaben an oder verändern sie. Das Ausgabeformular zeigt mehrere Datensätze in Form einer Liste an. Im Ausgabeformular blättern Sie durch die Datensätze, wählen Datensätze aus und sortieren oder drucken diese. Sie können Datensätze auch direkt im Ausgabeformular eingeben und ändern.

Sie können zwischen unterschiedlichen Tabellen hin- und herspringen und im Anzeigefenster der Datensätze zwischen Eingabe- und Ausgabeformular wechseln.

In der Anwendungsumgebung erfolgt dieser Wechsel in der Regel in der angepassten Oberfläche. Der Wechsel zwischen verschiedenen Tabellen und Formularen wird über 4D Befehle der Programmiersprache ausgeführt.

In der Designumgebung gibt es dafür spezifische Befehle. Sie können jederzeit zwischen Tabellen und Formularen wechseln.

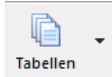
Jede Tabelle hat ein aktuelles Eingabe- und Ausgabeformular. Diese Standardformulare sind im Explorer für jede Tabelle der Datenbank festgelegt und werden systematisch verwendet, außer sie rufen während einer Arbeitssitzung andere Formulare auf.

Um die Tabellen zu wählen, deren Daten Sie anzeigen wollen, klicken Sie in der 4D Werkzeugleiste auf die Schaltfläche **Tabellen**. Um die gewünschten Tabellen und Formulare anzuzeigen, müssen Sie das Fenster **Liste der Tabellen** verwenden. Hier können Sie rasch zwischen verschiedenen Tabellen und zwischen verschiedenen Formularen wechseln. Sie können auch eine neue Tabelle oder ein neues Formular wählen, wenn Sie bereits ein Eingabe- oder Ausgabeformular verwenden. Die Auswahl wird sofort umgesetzt.

Datensätze anzeigen

Um das Fenster zum Anzeigen der Datensätze in den Vordergrund zu setzen:

1. Klicken Sie in der 4D Werkzeugleiste auf die Schaltfläche **Tabellen**.



ODER

Wählen Sie im Menü **Datensätze>Aktuelle Tabelle anzeigen (Tabellennamen)**.

Die Daten der aktuellen Tabelle erscheinen dann im aktuellen Ausgabeformular der Tabelle.

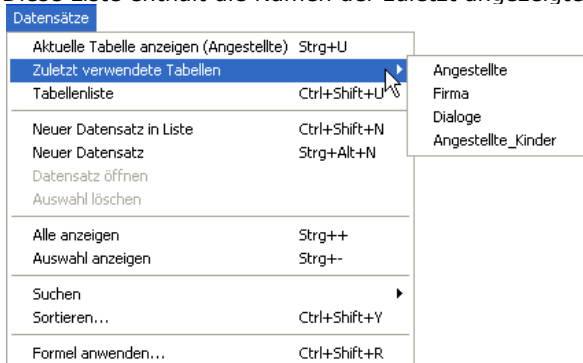
Um die Tabelle der angezeigten Datensätze zu wechseln:

1. Wählen Sie eine Tabelle aus der Liste der Tabellen.

ODER

Wählen Sie im Menü **Datensätze>Zuletzt verwendete Tabellen** eine Tabelle aus.

Diese Liste enthält die Namen der zuletzt angezeigten Tabellen in der Arbeitssitzung:



Die Daten der gewählten Tabelle erscheinen dann im aktuellen Ausgabeformular der Tabelle.

Fenster Liste der Tabellen



Um eine Tabelle oder ein Formular über das Fenster Liste der Tabellen zu wählen:

1. Wählen Sie im Menü **Datensätze>Tabellenliste**. Die Liste der Tabellen kommt in den Vordergrund.



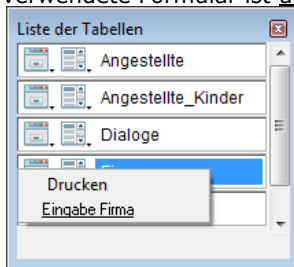
Bei Bedarf können Sie über den Rollbalken im Fenster scrollen oder das Fenster größer ziehen, um die Namen aller Tabellen anzuzeigen.

2. Klicken Sie auf einen Tabellennamen im Fenster.
4D zeigt die aktuelle Auswahl dieser Tabelle im Ausgabeformular an.

Vor jedem Tabellennamen im Fenster gibt es zwei Icons für Eingabe- und Ausgabeformulare. Der erste Icon  zeigt die Eingabeformulare, der zweite Icon  die Ausgabeformulare.

Um die aktuellen Formulare zu ändern:

1. Klicken Sie mit gedrückter Maustaste auf das entsprechende Icon in der Liste.
Ein DropDown-Menü erscheint, das alle Eingabe- bzw. Ausgabeformulare der gewählten Tabelle auflistet. Das aktuell verwendete Formular ist unterstrichen.



2. Wählen Sie das Formular, das Sie als das standardmäßige Eingabe- bzw. Ausgabeformular verwenden wollen.
Diese Einstellungen bleiben erhalten, bis Sie neue definieren oder die Datenbank verlassen.

Hinweis: Standardmäßige Eingabe- und Ausgabeformulare definieren Sie auf der [Seite Formulare](#) des Explorer. Weitere Informationen dazu finden Sie im Abschnitt [Eingabe- und Ausgabeformulare festlegen](#).

Aktuelle Auswahl

Bei der Verwaltung von Daten markieren Sie eine Reihe von Datensätzen, mit denen Sie arbeiten möchten. Diese Reihe von Datensätzen wird als aktuelle Auswahl bezeichnet. Die aktuelle Auswahl kann keinen, einen, ein paar oder alle Datensätze einer Tabelle umfassen. Jede Tabelle hat eine eigene aktuelle Auswahl von Datensätzen.

Die aktuelle Auswahl ist ein wichtiges Konzept in 4D. Die meisten Operationen zur Datenverwaltung werden in der aktuellen Auswahl ausgeführt. Diese Operationen sind:

- Datensätze sortieren
- Einzelne Datensätze anzeigen und ändern
- Mehrere Datensätze aktualisieren
- Einen Bericht drucken
- Etiketten erstellen
- Daten in Diagrammen darstellen
- Datensätze exportieren

Mit anderen Worten, das Erstellen einer aktuellen Auswahl ist der erste Schritt vieler weiterer Operationen zur Datenverwaltung.

Die aktuelle Auswahl von Datensätzen entspricht immer der zuletzt ausgeführten Datensatzauswahl. Sie haben zum Beispiel eine Firmendatenbank mit einer Tabelle [Angestellte] zum Verwalten der Personaldaten. Nehmen wir an, Sie wollen darin die Datensätze aller Ingenieure der Firma herausfinden.

Vor Ihrer Suche mag die aktuelle Auswahl alle Mitarbeiter der Firma umfassen - Vertrieb, Buchhaltung, Ingenieure, usw.. Nach Ihrer Suche enthält die aktuelle Auswahl nur noch die Datensätze aller Ingenieure. Würden Sie eine Liste von Datensätzen ausdrucken, enthält der Ausdruck nur die Datensätze der aktuellen Auswahl - in diesem Fall alle Ingenieure der Firma. Würden Sie die Löhne der Mitarbeiter grafisch darstellen, würde das Diagramm nur die Gehälter aller Ingenieure der Firma berücksichtigen.

Die aktuelle Auswahl bleibt bestehen, bis eine Operation ausgeführt wird, die diese ersetzt. Die aktuelle Auswahl von Datensätzen ändert sich bei folgenden Operationen:

- Alle Datensätze auswählen
- Eine Untergruppe von Datensätzen auswählen
- Nach Datensätzen suchen

Die Titelleiste des Ausgabeformulars gibt an, wieviel Datensätze in der aktuellen Tabelle liegen und wie viele davon in der aktuellen Auswahl sind

4D Server: In remote 4D erscheint nur die Anzahl der Datensätze in der aktuellen Auswahl.

Die Steuertafel mancher Eingabeformulare zeigt unter den Steuerschaltflächen die Anzahl der gewählten Datensätze und die Gesamtanzahl in der aktuellen Auswahl. Die Anzahl der ausgewählten Datensätze entspricht ihrer Position in der aktuellen Auswahl.

Jede Tabelle in einer Datenbank hat ihre eigene aktuelle Auswahl. In einer relationalen Datenbank kann der Wechsel der aktuellen Auswahl einer Tabelle ebenfalls die aktuelle Auswahl in verknüpften Tabellen ändern. Beispielsweise ändert in einer Personaldatenbank mit den verknüpften Tabellen [Angestellte] und [Abteilungen] das Öffnen eines Eingabelayouts in der Tabelle [Abteilungen] auch die aktuelle Auswahl der Tabelle [Angestellte]. Das bedeutet, dass die Mitarbeiter dieser gesuchten Abteilung auch die neue aktuelle Auswahl der Tabelle [Angestellte] werden. Weitere Informationen dazu finden Sie im Abschnitt **Verknüpfungsarten**.

Wenn Sie verschiedene Prozesse auslösen, um Datenbankaufgaben zu erledigen, können Sie gleichzeitig mit mehreren aktuellen Auswahlen arbeiten. Jeder Prozess verhält sich wie eine eigene 4D Umgebung, in der eigenständige Aufgaben ausgeführt werden. Es kann nützlich sein, pro Tabelle mehrere aktuelle Auswahlen zu haben, um zwei oder mehrere Datensatzgruppen miteinander zu vergleichen. Sie können beispielsweise die monatlichen Umsätze mehrerer Verkaufsbezirke miteinander vergleichen.

Weitere Informationen zu Prozessen finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Einführung in Prozesse**.

Alle Datensätze anzeigen

Wenn Sie ein Ausgabeformular verwenden, können Sie die aktuelle Auswahl widerrufen und wieder alle Datensätze der aktuellen Tabelle aufrufen.

In der Anwendungsumgebung können Sie diese Operation über die Standardaktion "Alles auswählen" oder den 4D Befehl **ALL RECORDS** ausführen.

In der Designumgebung können Sie im Menü **Datensätze>Alle anzeigen** wählen. Alle Datensätze der aktuellen Tabelle werden wieder in einer Liste gezeigt.

Hinweis: Der Menübefehl **Alle anzeigen** ist inaktiv, wenn Sie ein Eingabeformular verwenden.

Eine Unterauswahl von Datensätzen festlegen

Sie können eine neue aktuelle Auswahl erzeugen, indem Sie direkt im Ausgabeformular Datensätze markieren und diese als neue aktuelle Auswahl festlegen. Dies ist dann eine Unterauswahl.

In der Anwendungsumgebung verwalten Sie vom Benutzer markierte Datensätze über Befehle zum Verwalten von **Mengen** und Befehle der Programmiersprache im Kapitel **Auswahl**.

In der Designumgebung wählen Sie im **Menü Datensätze>Auswahl anzeigen**. Wählen Sie diesen Befehl, beschränkt 4D die aktuelle Auswahl auf die Datensätze, die im Fenster manuell gewählt (markiert) wurden.

Die verschiedenen Möglichkeiten zum Auswählen von Datensätzen werden im Abschnitt "Datensätze markieren" unter **Ausgabeformulare** beschrieben.

Dieser Abschnitt behandelt die wichtigsten Operationen beim Arbeiten mit Datensätzen in der Datenbank. Das sind folgende Operationen:

- Datensätze hinzufügen
- Datensätze ändern
- Datensätze löschen

Haben Sie in der Designumgebung eine Datenbank erstellt, können Sie mit den Datensätzen arbeiten, also Daten eingeben, ändern, löschen, etc. Diese Aktionen werden in der Regel in der Anwendungsumgebung über eine eigene Oberfläche ausgeführt.

4D bietet auch die Möglichkeit, über die Designumgebung Datensätze einzugeben, zu ändern oder zu löschen. Sie können also bestimmte Grundaufgaben ausführen und die Daten in Ihrer Anwendung testen.

Oft müssen Sie bereits eingegebene Daten wieder ändern. Wollen Sie einen oder mehrere Datensätze ändern, können Sie zur Anzeige auf dem Bildschirm eine der Auswahlmethoden verwenden. Weitere Informationen dazu finden Sie im Kapitel **Datensätze suchen**. Sie können dann die Änderungen im Eingabeformular oder Ausgabeformular ausführen.

Manchmal müssen Sie exakt die gleichen Änderungen an mehreren Datensätzen einer Tabelle vornehmen. Dies wird globale Aktualisierung genannt. In 4D können Sie die aktuelle Auswahl einer Tabelle automatisch aktualisieren und müssen nicht jeden Datensatz einzeln ändern.

Es kann ebenso vorkommen, dass Sie einen oder mehrere Datensätze löschen möchten, z.B. weil die Daten veraltet sind. Wird ein Datensatz weiterhin benötigt, der nicht mehr korrekte Werte enthält, so empfiehlt es sich, nur die Inhalte zu ändern, anstatt den ganzen Datensatz zu löschen. Den aktuellen Datensatz löschen Sie im Eingabeformular oder einer Teilauswahl an Datensätzen in der aktuellen Auswahl des Ausgabeformulars.

Neue Datensätze hinzufügen

Sie fügen neue Datensätze entweder mit dem Eingabeformular oder dem Ausgabeformular hinzu.

In der Anwendungsumgebung können Datensätze auch über die Standardaktion „Neuer Datensatz“ oder die 4D Befehle **ADD RECORD** und **CREATE RECORD** hinzufügen.

In der Designumgebung bietet 4D verschiedene Funktionalitäten zum Hinzufügen.

- Um einen neuen Datensatz über das Eingabeformular einzugeben:
 1. Wählen Sie in der Designumgebung im Menü **Datensätze > Neuer Datensatz**.
Sie können den Befehl **Neuer Datensatz** sowohl in einem Eingabeformular als auch in einem Ausgabeformular verwenden.
4D zeigt ein leeres Eingabeformular und setzt die Einfügemarke in den ersten editierbaren Bereich des Formulars.
 2. Geben Sie Ihre Daten ein und drücken die **Tabulatortaste** oder die **Zeilenschaltung**, um zum nächsten Bereich zu springen
Wiederholen Sie diesen Vorgang in jedem editierbaren Bereich, bis alle Daten für diesen Datensatz eingegeben sind.
Sind alle Daten eingegeben, können Sie den Datensatz bestätigen.
 3. Drücken Sie die Eingabetaste auf dem Zahlenblock Ihrer Tastatur oder klicken Sie auf die Schaltfläche **Bestätigen** des Eingabeformulars, um den Datensatz zu bestätigen.
Hinweis: Zugewiesene Tastaturkürzel können Sie in den Datenbank-Eigenschaften auf der **Seite Oberfläche** ändern.
Wenn Sie den Datensatz bestätigen, wird er in der Datenbank hinzugefügt. Auf dem Bildschirm erscheint ein neues leeres Eingabeformular.
 4. Wiederholen Sie diesen Vorgang, bis alle Datensätze eingegeben sind.
ODER
Um die Dateneingabe abzuschließen, klicken Sie entweder auf die Schaltfläche **Abbrechen** des Formulars oder drücken Sie die **Esc-Taste**, wenn der nächste leere Datensatz angezeigt wird.
Hinweis: Zugewiesene Tastaturkürzel können Sie in den Datenbank-Eigenschaften auf der **Seite Oberfläche** ändern.
Mit jeder dieser Aktionen wechseln Sie in das Ausgabeformular.

Sie können Datensätze auch direkt im Ausgabeformular eingeben, Hier können Sie jedoch keine Daten in Variablen, Datenfelder aus anderen Tabellen oder Unterformularen eingeben.

- Um einen neuen Datensatz mit dem Ausgabeformular einzugeben:
 1. Wählen Sie in der Designumgebung im Menü **Datensätze > Neuer Datensatz in Liste**.
Im ersten Datenfeld unter dem zuletzt angezeigten Datensatz im Ausgabeformular erscheint eine Einfügemarke.
 2. Tippen Sie den Wert in das Datenfeld und klicken auf die **Tabulatortaste** oder die **Zeilenschaltung**, um jeweils zum nächsten Datenfeld dieses Datensatzes zu springen.
Hinweis: Mit der Kombination **Umschalttaste + Tabulatortaste** bzw. **Umschalttaste + Zeilenschaltung** können Sie sich in umgekehrter Richtung in den Datenfeldern bewegen.
Alle Kontrollen zur Datenprüfung, die den Datenfeldern des Ausgabeformulars zugewiesen wurden, werden automatisch ausgeführt. Beispielsweise wird bei einem Datenfeld mit Auswahlliste die Liste in dem Moment angezeigt, in dem Sie die Einfügemarke mit der **Tabulatortaste** oder mit Mausclick in das Feld setzen.

3. Drücken Sie die **Eingabetaste** auf dem Zahlenblock, um den neuen Datensatz zu speichern und einen neuen leeren Datensatz zu erzeugen
ODER
Klicken Sie in einen anderen Datensatz.
4D bestätigt die hinzugefügten Eingaben in den Datensatz.

Datensätze ändern

Sie ändern Datensätze, um Informationen zu aktualisieren oder wenn Sie feststellen, dass Ihre ursprünglich eingegebenen Informationen falsch sind. Sie müssen die betreffenden Datensätze jedoch erst markieren, um sie als aktuelle Auswahl festzulegen. Sie können die Datensätze entweder suchen oder im Ausgabeformular markieren.

Sie können Datensätze sowohl im Eingabeformular als auch im Ausgabeformular ändern. Das Ausgabeformular ist praktischer, wenn Sie mehr als einen Datensatz ändern wollen, da dieses Formular mehrere Datensätze zur gleichen Zeit anzeigt. Beachten Sie jedoch, dass ein Ausgabeformular nicht unbedingt alle Datenfelder und Eingabekontrollen des Eingabeformulars enthält.

Wird gerade ein Datensatz in einem anderen Prozess geändert, ist der Datensatz gesperrt, d.h. er wird angezeigt, lässt sich jedoch nicht ändern. Öffnen Sie einen gesperrten Datensatz, können Sie die Einträge der Datenfelder ansehen, jedoch nicht verändern.

In der Anwendungsumgebung werden Datensätze über die Standardaktion "Unterdatensatz bearbeiten" (Datensatz im Listenformular ändern) oder über den 4D Befehl **MODIFY RECORD** bearbeitet.

In der Designumgebung bietet 4D verschiedene Möglichkeiten.

- Um Datensätze im Eingabeformular zu ändern:
 1. Markieren Sie in der Designumgebung einen Datensatz im Ausgabeformular und wählen im Menü **Datensätze > Datensatz öffnen**
ODER
Doppelklicken Sie auf den Datensatz im Ausgabeformular.
4D zeigt den Datensatz im Standard Eingabeformular an.
 2. Wählen Sie verschiedene Datenfelder aus und bearbeiten, ersetzen oder löschen Sie Werte.
 3. Klicken Sie auf die Schaltfläche Bestätigen oder drücken Sie die Eingabetaste des Zahlenblocks, um den geänderten Datensatz zu bestätigen und zum Ausgabeformular zurückzukehren
ODER
Klicken Sie am linken Rand auf eine Steuerschaltfläche (Erster, vorheriger, nächster, erster, letzter Datensatz), um den Datensatz zu bestätigen und in der aktuellen Auswahl zum nächsten zu springen.
Durch Anklicken einer Steuerschaltfläche bewegen Sie sich in den Datensätzen der aktuellen Auswahl.
Sie können Ihre Änderungen jederzeit unterbrechen und in das Ausgabeformular zurückkehren. Klicken Sie dazu entweder auf die Schaltfläche **Abbrechen** oder drücken die **Esc-Taste**.

Sie können die angezeigten Datenfelder im Ausgabeformular direkt ändern. Beachten Sie, dass Sie in diesem Fall keine Daten in Variablen, Feldern aus anderen Tabellen oder Unterformularen eingeben können.

In der Anwendungsumgebung können Sie die Möglichkeit, Datensätze im Listenformular zu ändern, steuern.

- Um Datensätze im Ausgabeformular zu ändern:
 1. Wählen Sie einen Datensatz aus und markieren das entsprechende Datenfeld.
Das Feld im Ausgabeformular wird eingebbar.
Hinweis: In der Designumgebung können Sie jederzeit im Menü **Datensätze** den Befehl **Datensatz ändern** aufrufen, um den Datensatz im Eingabeformular zu ändern
 2. Tippen Sie den neuen Text ein und drücken die **Tabulatortaste** oder die **Zeilenschaltung**.
4D speichert Ihre Änderungen und wählt das nächste Datenfeld aus.
 3. Ändern Sie weitere Datenfelder, falls erforderlich.
 4. Doppelklicken Sie in einen anderen Datensatz des Ausgabeformulars, um ihn zu ändern.

Globale Aktualisierungen

Wollen Sie eine bestimmte Änderung bei einer Gruppe von Datensätzen ausführen, können Sie eine globale Aktualisierung ausführen. Mit einer globalen Aktualisierung können Sie Änderungen an einer Gruppe von Datensätzen automatisieren, die andernfalls zeitaufwendig und ermüdend wären. In folgenden Fällen ist eine globale Aktualisierung sinnvoll:

- Alle Preise einer Inventartabelle um einen bestimmten Prozentsatz verändern
- Ein numerisches oder alphanumerisches Datenfeld formatieren

Eine globale Aktualisierung erfolgt durch Zuweisung einer Formel in der aktuellen Datensatzauswahl, d.h. über die Formel werden alle Datensätze der aktuellen Auswahl geändert.

Hierzu einige Beispiele:

- Die folgende Formel multipliziert das Datenfeld mit 1,05, etwa bei einer tariflichen Gehaltserhöhung:

```
[Angestellte]Gehalt:=[Angestellte]Gehalt*1.05
```

- Die folgende Formel verwendet eine eingebaute Funktion, um den Inhalt des Datenfeldes *LKZ* automatisch in Großbuchstaben einzugeben. Damit stellen Sie sicher, dass Länderkürzel wie „D“ oder „NL“ auf Etiketten und in Berichten einheitlich erscheinen:

```
[Kunden]LKZ:=Uppercase([Kunden]LKZ)
```

- Diese Formel verwendet eine benutzerdefinierte Funktion zum Formatieren des Datenfeldes *Nachname*. Es schreibt den ersten Buchstaben groß und alle übrigen klein.

[Angestellte]Nachname:=**Capitalize**([Angestellte]Nachname)

Eine der großen Stärken von 4D ist, dass bei globalen Aktualisierungen auch benutzerdefinierte Funktionen ausführbar sind. Formeln können auch Befehle der 4D Programmiersprache sowie Projektmethoden enthalten. Aus Sicherheitsgründen kann der Entwickler den Zugriff auf Projektmethoden in Formeln einschränken (siehe [Seite Sicherheit](#)).

Eine Formel muss immer eine logische Zeile lang sein, d.h. Sie können nicht die Zeilenschaltung drücken und eine zweite Zeile eingeben. Dagegen können Methoden, die für den Formeleditor benutzbar sind, auch mehrere Zeilen lang sein.

Für eine globale Aktualisierung können Sie in eine Formel eingeben, die dann auf jeden Datensatz der aktuellen Auswahl angewandt wird. Dazu wählen Sie im Menü **Datensätze > Formel anwenden**. Sie können auch eine zuvor auf der Festplatte gesicherte Formel laden (Endung .4fr). Weitere Informationen dazu finden Sie im Kapitel .

In der Anwendungsumgebung können Sie eine Formel zur Aktualisierung direkt über den 4D Befehl **EXECUTE FORMULA** oder den Formeleditor über den 4D Befehl **EDIT FORMULA** anzeigen.

Datensätze löschen

Sie können einen Datensatz löschen, der veraltet ist oder nicht mehr gebraucht wird. Wird der Datensatz weiterhin benötigt, aber seine Werte sind falsch, sollten Sie ihn lieber ändern als löschen.

Sie können einen Datensatz auf zwei Arten löschen:

- Einzelnen im Eingabeformular
- Eine ausgewählte Gruppe von Datensätzen im Ausgabeformular

In der Anwendungsumgebung werden Datensätze über die Standardaktionen „Datensatz löschen“ oder „Unterdatsatz löschen“ gelöscht oder über den 4D Befehl **DELETE RECORD** bzw. **DELETE SELECTION**.

In der Designumgebung können Sie auch die Löschtaste oder im Menü **Bearbeiten** den Befehl **Löschen** verwenden.

Warnung: Das Löschen von Datensätzen ist endgültig und lässt sich nur durch Wiederherstellen der Anwendung via Backup wieder rückgängig machen. Deshalb zeigt 4D beim Löschen von Datensätzen ein Dialogfenster, um die Operation zu bestätigen.

Das Löschen von Datensätzen im Eingabeformular hat den Vorteil, dass Sie zuvor den gesamten Inhalt des Datensatzes überprüfen können.

- Um Datensätze im Eingabeformular zu löschen:
 1. Öffnen Sie den gewünschten Datensatz.
 2. Klicken Sie auf die Schaltfläche **Löschen**, um den Datensatz zu löschen.
Diese Schaltfläche kann je nach Layout des Eingabeformulars unterschiedlich aussehen. Standardmäßig erscheint sie als Icon, der einen Papierkorb darstellt und dem die Aktion "Datensatz löschen" zugeordnet ist. Die Schaltfläche Löschen kann im Formular auch nicht vorhanden sein.
4D zeigt ein Dialogfenster, um die Operation zu bestätigen. Sie können diese nicht widerrufen, nachdem der Datensatz aus der Datenbank entfernt wurde.
 3. Klicken Sie auf die Schaltfläche **Ja**, um die Operation auszuführen
4D löscht den aktuellen Datensatz aus der Datenbank und kehrt zum Ausgabeformular zurück.

Das Löschen im Ausgabeformular hat den Vorteil, dass Sie mehrere Datensätze auf einmal aus der Anwendung entfernen können. Die entsprechenden Datensätze müssen markiert sein.

- Um Datensätze im Ausgabeformular zu löschen:
 1. Markieren Sie den oder die Datensätze, die gelöscht werden sollen.
 2. Wählen Sie in der Designumgebung im Menü **Bearbeiten** den Befehl **Löschen** oder drücken Sie die **Löschen- bzw. Rückschritt-Taste**.
4D zeigt ein Dialogfenster, um die Operation zu bestätigen. Sie können das Löschen nicht widerrufen.
 3. Klicken Sie auf die Schaltfläche **OK**, um den Befehl auszuführen.
4D entfernt die markierten Datensätze aus der Datenbank.
Hinweis: Wollen Sie alle Datensätze einer Tabelle löschen, wählen Sie die Menüs **Datensätze > Alle anzeigen** und **Bearbeiten > Alles auswählen** und dann im Menü **Bearbeiten > Löschen** (oder die Taste **Löschen**).

Gesperrte Datensätze löschen

Sie können keine gesperrten Datensätze löschen. Datensätze sind gesperrt, wenn sie von einem anderen Prozess verwendet werden. Öffnet beispielsweise ein Prozess einen Datensatz, um ihn zu ändern, so sperrt 4D diesen Datensatz für alle anderen Prozesse.

Hinweis zu 4D Server: Datensätze können auch gesperrt sein, weil andere Anwender diese gerade bearbeiten.

Bevor Sie Datensätze löschen, erstellen Sie eine Auswahl, die gelöscht werden soll. Enthält Ihre Auswahl einen oder mehrere gesperrte Datensätze, wird das Löschen bis auf die gesperrten Datensätze ausgeführt. Die gesperrten Datensätze verbleiben in der aktuellen Auswahl. Sie müssen warten, bis die Sperre aufgehoben ist (z.B. nicht mehr bearbeitet werden), um sie zu löschen. Solche Situationen können Sie mit Befehlen der 4D Programmiersprache verwalten, siehe Kapitel [Datensatz sperren](#).

In einem anderen Prozess gelöschte Datensätze

Die aktuelle Auswahl kann sich ändern durch Datensätze, die in anderen Prozessen gelöscht wurden. So kann es vorkommen, dass Sie

während einer Arbeitssitzung einen weiteren Prozess starten, der verschiedene Datensätze aus einer Tabelle löscht. Die in

diesem Prozess gelöschten Datensätze werden dauerhaft aus der Tabelle entfernt. Dennoch erscheinen die Datensätze noch in der aktuellen Auswahl.

Hierzu ein Beispiel: Stellen Sie sich eine Tabelle mit fünfzig Datensätzen vor, alle in der aktuellen Auswahl. Zu diesem Zeitpunkt gibt die Titelleiste des Ausgabeformulars an, dass „50 von 50“ Datensätzen in der aktuellen Auswahl ausgewählt sind. Wird davon ein Datensatz in einem anderen Prozess gelöscht, ändert sich die Anzeige der Titelleiste auf „50 von 49“. Es befinden sich also mehr Datensätze in der aktuellen Auswahl als in der eigentlichen Tabelle. Die Titelleiste wird angeglichen, wenn Sie eine andere aktuelle Auswahl festlegen.

Versuchen Sie den gelöschten Datensatz zu ändern oder zu löschen, erscheint ein Dialogfenster mit der Meldung, dass der Datensatz gelöscht wurde.

Hinweis zu 4D Server: Datensätze, die von anderen Anwendern gelöscht wurden, haben den gleichen Effekt auf die aktuelle Auswahl. Die Datensätze werden aus der Tabelle, aber nicht aus der aktuellen Auswahl gelöscht. So kann die aktuelle Auswahl mehr Datensätze enthalten, als tatsächlich in der Tabelle existieren.

Datensätze drucken

Sie können die Datensätze Ihrer Tabellen jederzeit drucken. Dazu verwenden Sie Formulare in der Anwendung.

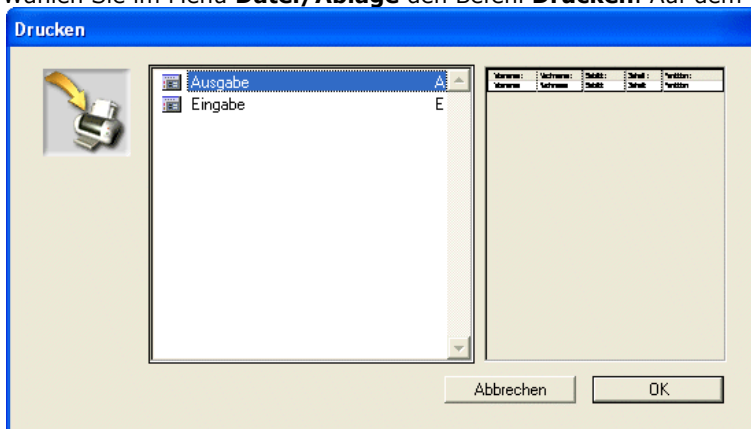
In der Anwendungsumgebung verwenden Sie zum Drucken in der Regel Schaltflächen oder eigene Menüs. Die Datensätze werden in dafür vorgesehenen Formularen gedruckt. Über die Befehle der 4D Programmiersprache im Kapitel **Drucken** können Sie eigene Druckprozesse einrichten.

In der Designumgebung ermöglicht 4D, einen oder mehrere Datensätze zu drucken oder ein bestimmtes Formular zum Drucken festzulegen. Sie können eine Liste von Datensätzen (Bericht) oder Datensätze einzeln drucken.

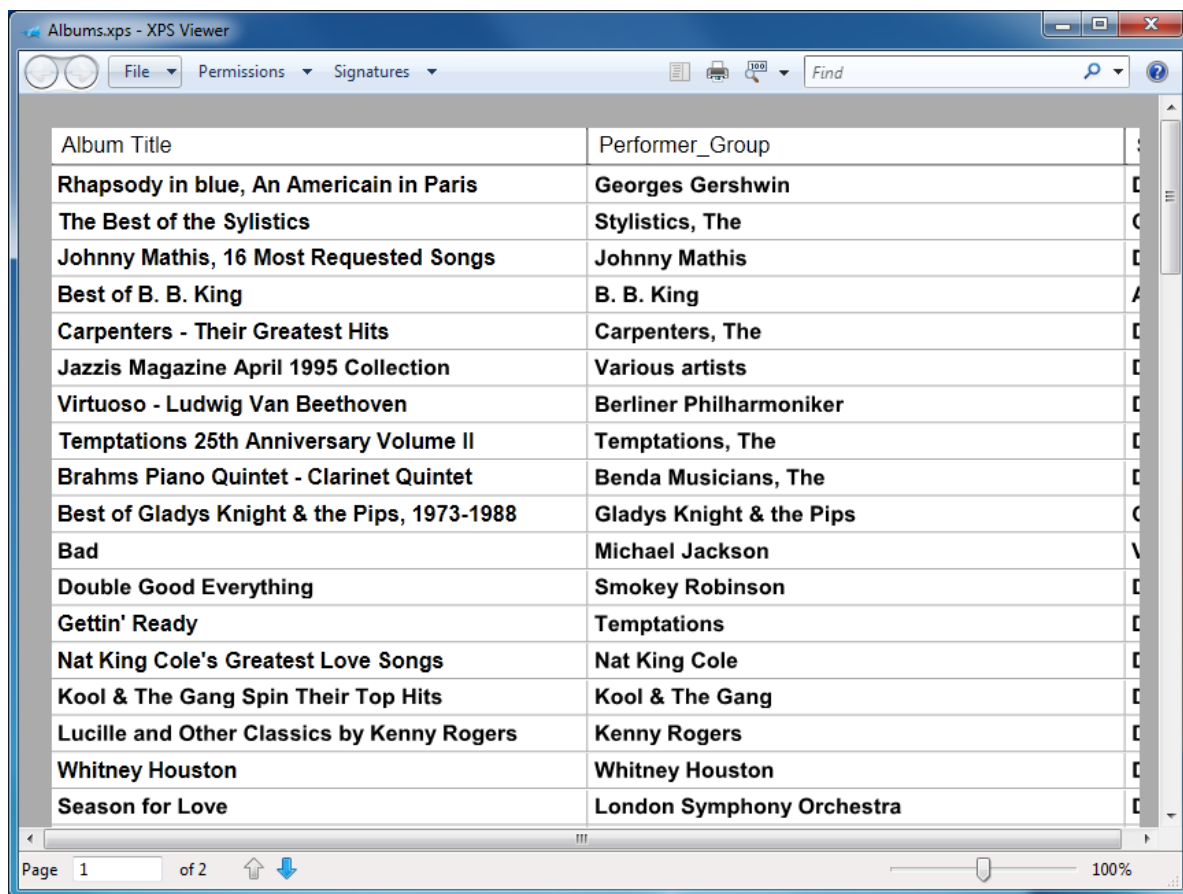
4D druckt die Datensätze der aktuellen Auswahl. Legen Sie die aktuelle Auswahl mit der Untermenge von Datensätzen fest, die Sie drucken wollen. Wenn Sie wollen, dass der Bericht die Datensätze in einer bestimmten Reihenfolge auflistet, müssen Sie die aktuelle Auswahl vor dem Drucken sortieren.

Um Datensätze zu drucken:

1. Wählen Sie im Anzeigefenster der Datensätze die entsprechenden Datensätze aus.
2. (Optional) Sortieren Sie die Datensätze.
Dieser Schritt ist zwingend, wenn Sie Datensätze in einem Bericht mit Umbrüchen drucken. In diesem Fall müssen Sie die Datensätze mindestens eine Ebene mehr sortieren als Umbrüchebenen im Bericht vorhanden sind. Weitere Informationen dazu finden Sie im Abschnitt **Formulare für gedruckte Berichte**.
3. Wählen Sie im Menü **Datei/Ablage** den Befehl **Drucken**. Auf dem Bildschirm erscheint folgendes Dialogfenster:



4. Wählen Sie das entsprechende Formular. Klicken Sie auf ein Formular, erscheint eine Vorschau im rechten Bereich
5. Klicken Sie auf die Schaltfläche **OK**.
4D zeigt das Dialogfenster für die Seiteneinstellung des ausgewählten Druckers. Wählen Sie die gewünschten Einstellungen und klicken auf die Schaltfläche **OK**. 4D zeigt den Druckdialog für den Drucker, den Sie im Druckmanager ausgewählt haben. Wollen Sie den Bericht erst auf dem Bildschirm ansehen, klicken Sie auf das Kontrollkästchen **Druckvorschau**.
Der Bericht wird dann Seite für Seite auf dem Bildschirm abgebildet. Die folgende Abbildung zeigt die Vorschau eines Berichts auf dem Bildschirm:



Hinweis: Auf Mac OS verwendet 4D das PDF Format, das nativ im System zum Erstellen von Vorschauen enthalten ist. Unter Windows erzeugt 4D ab Version 13 eine Druckvorschau als XPS Dokument. Weitere Informationen dazu finden Sie im nächsten Abschnitt.

Wird eine Seite des Berichts in der Vorschau angezeigt, können Sie folgendes ausführen:

- Wollen Sie den Bericht vergrößern, klicken Sie auf die Schaltfläche für Zoomen. Um den Zoom-Modus zu verlassen, klicken Sie erneut auf diese Schaltfläche.
- Beim Zoomen können Sie durch Bewegen des Mauszeigers in Handform den angezeigten Ausschnitt der Seitenansicht verschieben.
- Wollen Sie zur nächsten oder vorherigen Seite des Berichts gehen, klicken Sie auf die Schaltflächen für **Nächste Seite** oder **Vorige Seite**. Sie erscheinen in Grauschrift, wenn der Bericht nur 1 Seite hat.
- Wollen Sie den Druckvorgang abbrechen, klicken Sie auf die Schaltfläche **Stopp**.
- Wollen Sie die Seite der Vorschau drucken, klicken Sie auf die Schaltfläche **Drucken**.

Ist das Kontrollkästchen **Druckvorschau** nicht markiert, wird der Bericht direkt zum Drucker geschickt.

XPS Druckvorschau unter Windows

4D erstellt unter Windows ab Version 13 Druckvorschauen als XPS Dokumente. Das Format XPS (XML Paper Specification) ist eine offene Spezifikation für eine Sprache zur Seitenbeschreibung, die zuerst von Microsoft entwickelt und dann von Ecma International standardisiert wurde. Heute ist sie in Windows Anwendungen weit verbreitet und bietet eine akkurate Bildschirmarstellung gedruckter Dokumente, die von der Hardware-Konfiguration unabhängig ist.

Unterstützung durch das Betriebssystem

Um eine Druckvorschau als XPS Dokument zu erzeugen, ruft 4D einen XPS Treiber auf. Dieser Treiber ist ab Windows Vista integriert. Unter Windows XP2 oder Windows Server 2003 ist er verfügbar, wenn das Framework .NET installiert wird.

Die Darstellung erfolgt in einem XPS Viewer, der je nach Windows Version nativ oder via Internet Explorer verfügbar ist.

Nachfolgende Tabelle zeigt die Verfügbarkeit des XPS Treibers und Viewer, je nach der Windows Version:

Windows Version	XPS Treiber	XPS Viewer
Windows XP (SP2) oder Windows Server 2003	Nein	Nein
Windows XP (SP2) oder Windows Server 2003 + .NET 3.0 Framework	Ja	Ja mit IE
Windows Vista	Ja	Ja mit IE(*)
Windows 7 und höher	Ja	Ja(reader.exe)

(*) Der Viewer ist auch separat verfügbar.

Funktionsweise in 4D

In 4D wird die XPS Vorschau verwendet für:

- Alle 4D Formulare und Objekte, im Design- und im Anwendungsmodus
- Im Etiketten- und im Schnellberichteditor

Wird eine Druckvorschau erzeugt, d.h. im Druckdialog ist die Option "Auf Bildschirm" markiert, leitet 4D das Drucken in eine temporäre XPS Datei um und startet die Anwendung zusammen mit .XPS Dateien.

Hinweis: Ist diese Anwendung nicht der XPS Viewer (das ist z.B. mit Google Chrome der Fall), erzwingt 4D den Wechsel zum Internet Explorer.

Temporäre XPS Dateien werden im lokalen Benutzerordner gesichert: ...:\Users\
{UserName}\AppData\Local\Temp\4D\PrintPreview\FolderNumber\NumUUID\






Um eine Überladung dieses Ordners zu verhindern, werden nur die letzten 100 XPS Dokumente beibehalten. Hat der Benutzer nicht die passenden Zugriffsrechte, wird der Fehler -9799 zurückgegeben. Beachten Sie, dass Windows den "Temp" Ordner auch beim Starten leeren kann.

Der Viewer öffnet das Dokument automatisch, erlaubt aber nicht, direkt einen Namen zu vergeben; aus diesem Grund wird im nummerierten Ordner der UUID Unterordner zur Aufnahme des XPS Dokuments angelegt. Das Dokument wird dauerhaft benannt, wenn das Drucken beendet ist.

Standardmäßig wird der Name des Fensters verwendet, in dem das Drucken gestartet wurde. Sie können diesen Namen ändern, wenn Sie vor dem Drucken den Befehl **SET WINDOW TITLE** oder **SET PRINT OPTION(Spooler document name option; "xx")** aufrufen.

Zwei neue Befehle verbessern die Steuerung der Vorschau: **Get print preview** und **Is in print preview**.

Datensätze suchen

-  Suchen in 4D
-  Sucheditor
-  Suche nach Formel
-  Nach Beispiel suchen
-  Suchen und ändern

Suchen ist eine der häufigsten Datenbankaufgaben. Dies ist auch die einfachste Methode, um Daten auszuwählen, mit denen Sie arbeiten wollen.

Mit Suchen finden Sie eine Gruppe von Datensätzen in der Datenbank, die in einem oder mehreren Datenfeldern die gleichen Inhalte besitzen. Eine Suche führen Sie mit einem Suchkriterium aus. Ein Suchkriterium besteht aus mehreren Anweisungen, die 4D mitteilen, welche Datensätze in die aktuelle Auswahl aufgenommen werden sollen. Beispiel "Firmenname ist gleich 4D."

in Suchkriterium besteht stets aus drei Teilen: Feldname, Vergleichsoperator und Wert. Der Feldname gehört zur aktuellen Tabelle oder einer verknüpften Tabelle. Der Vergleichsoperator teilt 4D mit, wie es die Inhalte eines Datenfeldes mit dem angegebenen Wert vergleichen soll, z.B. ist gleich, größer als oder kleiner als. Der Wert legt die Zahl, den Text oder andere Werte fest, nach denen im angegebenen Datenfeld eines jeden Datensatzes gesucht werden soll.

Nehmen wir an, Sie möchten alle Datensätze sehen, die Mitarbeiter mit Jahresgehältern über 30.000 € enthalten. Ihr Suchkriterium würde „Gehalt größer als 30000“ lauten. „Gehalt“ ist das Datenfeld, „größer als“ ist der Vergleichsoperator und „30000“ ist der Wert.

Wenn Sie eine Datenbank durchsuchen, vergleicht 4D die Inhalte des im Suchkriterium angegebenen Datenfeldes mit dem von Ihnen bestimmten Wert. Die aktuelle Auswahl enthält anschließend alle Datensätze, die Ihren Suchkriterien entsprechen haben. Die neue aktuelle Auswahl kann aus keinem oder einem Datensatz bestehen, aber auch aus einer größeren Gruppe oder allen Datensätzen der Tabelle.

Sie können eine Suche sowohl in einem Eingabeformular als auch in einem Ausgabeformular ausführen. Führen Sie eine Suche in einem Eingabeformular aus, so wird der erste darin gefundene Datensatz der neuen aktuellen Auswahl angezeigt. Sie können diesen Datensatz anzeigen, ändern oder drucken. Besteht die aktuelle Auswahl aus mehreren Datensätzen, können Sie in ihr mit den Steuerungsschaltflächen blättern (Voriger und nächster, erster und letzter Datensatz). Ändern Sie einen Datensatz, bevor Sie eine Steuerungsschaltfläche anklicken, speichert 4D die Änderungen auf Ihre Festplatte.

In relationalen Datenbanken können Sie auch Datenfelder anderer Tabellen suchen, vorausgesetzt, dass eine Verknüpfung zwischen den Tabellen eingerichtet wurde.

Führen Sie einen Suchlauf in einem Ausgabeformular aus, wird die neue aktuelle Auswahl auch darin angezeigt. Sie können die aktuelle Auswahl im Menü **Datensätze** mit dem Befehl **Alle anzeigen** auf die gesamte Anzahl der Datensätze in der aktuellen Tabelle zurücksetzen.

Hinweis: Wird ein Datenfeld der Datenbankstruktur in der aktuellen Datenbank nicht verwendet, kann der Datenbankentwickler dieses durch Zuweisen des Attributs Unsichtbar ausblenden. Nur sichtbare Tabellen und Formulare erscheinen in den Sucheditoren. Weitere Informationen dazu finden Sie in den Abschnitten **Tabelleneigenschaften** und **Feldeigenschaften**.

Sucharten

4D bietet einige leistungsfähige Editoren zum Durchsuchen einer Datenbank. Sie können mit jedem dieser Editoren eine Suchbedingung aufstellen. Es spielt auch keine Rolle, ob Sie in einem Eingabeformular oder einem Ausgabeformular arbeiten. Datensätze, die die Suchbedingung erfüllen, werden zur neuen aktuellen Auswahl.

In der Designumgebung enthalten der Menübefehl **Suchen** im Menü **Datensätze** sowie die Schaltfläche **Suchen** in der Werkzeugleiste vier Einträge. Jeder zeigt ein anderes Dialogfenster. Sie unterscheiden sich in der Art der Suchläufe, die sie ausführen und in der Art, wie die aktuelle Auswahl angezeigt wird.

In der Anwendungsumgebung sind diese Dialogfenster über Befehle der Programmiersprache im Kapitel **Suchen** verfügbar.

Jeder Menübefehl bietet eine andere Suchmethode:

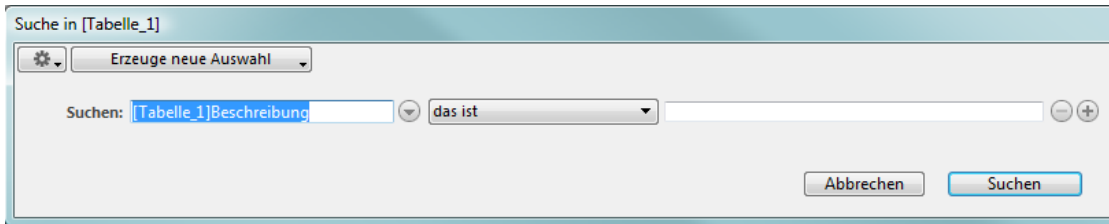
- **Suchen** (4D Befehle **QUERY** und **QUERY SELECTION**): öffnet den Sucheditor von 4D. Das ist ein vielseitiges Suchdialogfenster, das einfache wie verknüpfte Suchläufe durchführen kann. Sie können verknüpfte Suchbedingungen mit logischen Operatoren definieren, Suchbedingungen abspeichern oder die Suche auf die aktuelle Auswahl beschränken. Dieser Editor enthält einen fortgeschrittenen Modus, wo Sie einen Suchlauf anhand einer Formel einrichten (4D Befehle **QUERY BY FORMULA** und **QUERY SELECTION BY FORMULA**). Sie können z.B. die letzten drei Stellen einer sechsstelligen Versicherungsnummer suchen. Eine gültige Formel gibt als Ergebnis einen booleschen Ausdruck zurück (WAHR oder FALSCH).
- **Nach Beispiel suchen** (4D Befehl **QUERY BY EXAMPLE**): verwendet das aktuelle Eingabeformular als Suchmaske. Sie geben Ihre gesuchten Werte in die Eingabebereiche der gewünschten Datenfelder ein. Sie können verknüpfte Suchläufe definieren, indem Sie Werte in mehr als einen Bereich eingeben. Die Ergebnisse Ihrer Suche werden im aktuellen Ausgabeformular angezeigt.
- **Suchen und Ändern**: ist identisch zur Option „Nach Beispiel suchen“, mit dem Unterschied, dass der erste Datensatz der Auswahl für die Suche geladen wird und so gleich geändert werden kann. Sie können Änderungen machen, durch die gefundenen Datensätze navigieren und diese der Reihe nach ändern.

Sucheditor

Der Sucheditor ist ein vielseitiger Editor zum Erzeugen von einfachen oder verknüpften Suchbedingungen. Sie können verknüpfte Suchläufe erstellen, die mit den Operatoren Und, Oder bzw. Außer verbunden sind. Sie können z.B. nach Mitarbeitern suchen, die über 60 Jahre alt sind oder ein monatliches Einkommen über 4.000 € haben.

Sie können die aktuelle Auswahl an Datensätzen durchsuchen. Die anderen Suchmethoden durchsuchen stets die gesamte Tabelle. Sie können Suchbedingungen abspeichern und sie bei Bedarf wieder laden, um den Suchlauf zu wiederholen. Der Sucheditor merkt sich die zuletzt verwendeten Suchbedingungen. So können Sie diese entweder ändern oder löschen und neu eingeben.

Sie können in Datenfeldern der aktuellen Tabelle und der verknüpften Tabellen suchen. Sie können auch komplexe Suchläufe mit Formeln ausführen (siehe [Suche nach Formel](#)).

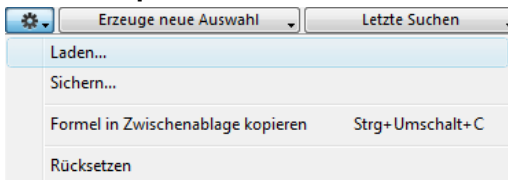


Menüs des Editors

Das Editorfenster enthält eine Menüleiste, einen Bereich zum Erstellen der Suche und das DropDown-Menü **Letzte Suchen** (wenn während der Sitzung bereits eine Suche durchgeführt wurde).

Menü Optionen

Das Menü **Optionen** enthält Befehle zum Verwalten des Suchcodes:



- **Laden...** und **Sichern...**: Diese Menübefehle verwalten das Laden und Sichern der Suchdateien auf der Festplatte. Führen Sie dieselbe Suche öfters durch, macht es Sinn, diese abzuspeichern. Sie müssen Ihre Suche nur einmal einrichten und können sie dann später wieder in den Sucheditor laden. Klicken Sie auf **Suchen**, um sie erneut auszuführen.

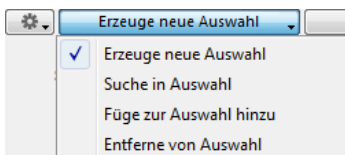
Um eine Suche auf der Festplatte zu sichern, legen Sie im Sucheditor die Suchkriterien fest und klicken auf die Schaltfläche **Sichern...**. 4D zeigt den Standard Sichern-Dialog an, wo Sie den Dateinamen und den Speicherort auf der Festplatte festlegen. Die Endung für 4D Suchen ist ".4df". Alle Parameter werden gesichert, d.h. Suchzeile(n), Suchaktion und Suche nach Formel.

Um eine gesicherte Suche zu laden, klicken Sie im Sucheditor einfach auf **Laden...** und wählen die Suchdatei (Endung ".4df"). 4D lädt Ihre Suche in den Sucheditor. Sie überschreibt alle zuvor im Sucheditor angezeigten Suchen.

- **Formel in Zwischenablage kopieren** setzt den Code der Formel aus dem Bearbeitungsbereich in die Zwischenablage.
- **Rücksetzen** löscht alle Zeilen der Suche im Editor. Die Operation lässt sich nicht rückgängig machen.

Menü Auswahlaktion

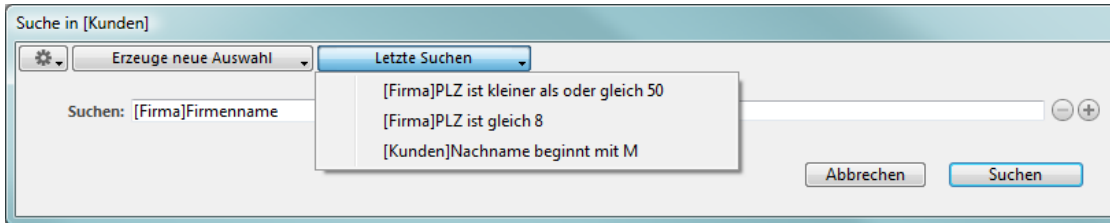
Dieses Menü definiert die auszuführende Suchaktion, sie basiert auf der aktuellen Auswahl der vorhandenen Datensätze.



- **Erzeuge neue Auswahl** (Standardaktion): 4D sucht in allen Datensätzen der Tabellen und ersetzt die ursprüngliche aktuelle Auswahl, um die gefundenen Datensätze anzuzeigen.
- **Suche in Auswahl**: 4D sucht nur in den Datensätzen der ursprünglichen aktuellen Auswahl und ersetzt dann diese Auswahl mit den gefundenen Datensätzen.
- **Füge zur Auswahl hinzu**: 4D sucht in allen Datensätzen der Tabellen und fügt die gefundenen Datensätze zur ursprünglichen aktuellen Auswahl hinzu. Gefundene Datensätze, die bereits Teil der aktuellen Auswahl sind, werden angezeigt, aber nicht dupliziert.
- **Entferne von Auswahl**: 4D sucht in allen Datensätzen der Tabellen und entfernt alle gefundenen Datensätze aus der ursprünglichen aktuellen Auswahl.

Menü Letzte Suchen

Dieses Menü erscheint, wenn bereits eine Suche durchgeführt wurde. Es zeigt die aktuellsten Suchläufe während der Sitzung, so dass Benutzer ihre gängigsten Suchen leicht wiederholen können. Die 10 letzten Suchen werden beibehalten.



Beachten Sie, dass Auswahlaktionen, die den Suchen zugeordnet sind, wie **Suche in Auswahl**, **Füge zu Auswahl hinzu**, etc. nicht gesichert werden.

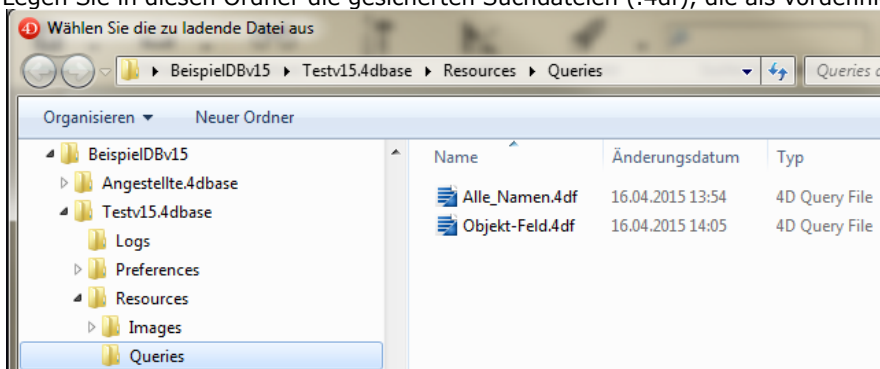
Wird eine Suche aus diesem Menü ausgewählt, erscheint die definierte Suche im Suchbereich. Sie können sie dann direkt ausführen oder bei Bedarf verändern.

Vordefinierte Suchläufe

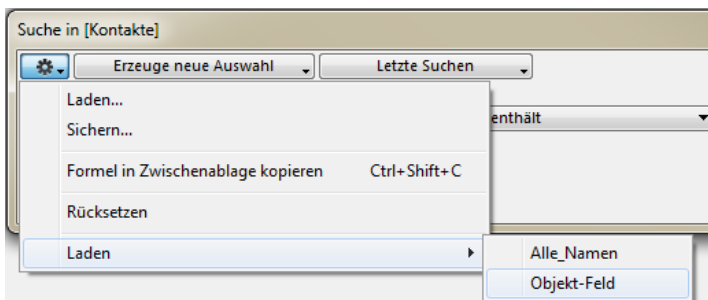
Der Sucheditor unterstützt **vordefinierte Suchen**. Eine vordefinierte Suche enthält, wie eine standardmäßig gesicherte Suche, die vollständige Definition der Suche mit allen Suchkriterien und kann jederzeit in den Sucheditor geladen werden. Vordefinierte Suchen lassen sich in Anwendungen im Einsatz einbinden und werden in einem DropDown-Menü des Sucheditors aufgelistet.

Um eine vordefinierte Suche einzurichten:

1. Erstellen Sie im Ordner **Resources** Ihrer Anwendung einen Unterordner "Queries".
2. Legen Sie in diesen Ordner die gesicherten Suchdateien (.4df), die als vordefinierte Suchen dienen sollen:



Wird im Ordner **Resources->Queries** mindestens eine .4df Suchdatei zur aktuellen Tabelle gefunden, wird im Sucheditor am Ende des Menüs ein neuer Eintrag **Laden >** angefügt. Das DropDown-Menü zeigt die Einträge der vordefinierten Suchen:



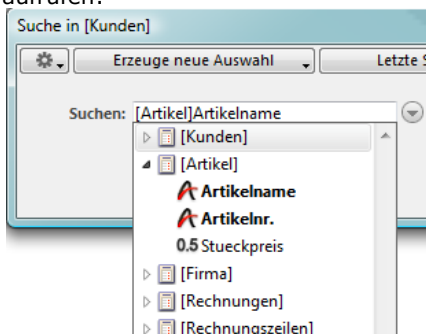
Klicken Sie auf einen Eintrag im DropDown-Menü, wird die entsprechende Suche in den Sucheditor geladen.

Zur Erinnerung: Im DropDown-Menü erscheinen nur die Suchdateien zur aktuellen Tabelle.

Suche erstellen

Um eine Standardsuche einzurichten, erstellen Sie eine Suchzeile vom Typ "Feld Operator Wert":

1. Um das Feld zu definieren, können Sie über den Pfeil rechts neben dem Eingabebereich die hierarchische Liste aufrufen:



Sie können auch in den Eingabebereich klicken und den/die ersten Buchstaben eines Tabellennamens eintippen (ohne "[") : Über das automatische Vervollständigen erscheinen dann passende Vorschläge:

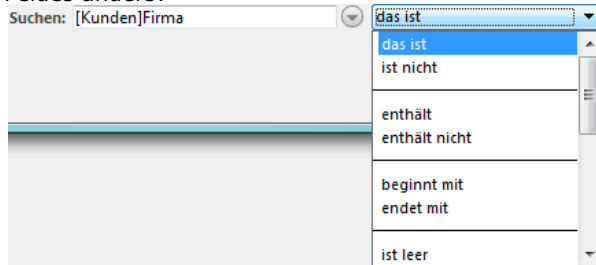
Suchen: [Kunden]

Haben Sie die Tabelle gewählt, drücken Sie die rechte Pfeiltaste →, um den Vorschlag zu bestätigen und auf die Liste der Felder zuzugreifen. Sie können wieder den/die ersten Buchstaben des Feldnamens eintippen oder über die Pfeiltasten ↑↓ durch die Felder der Tabelle scrollen:

Suchen: [Kunden]Nachname

Diese Liste zeigt alle Tabellen und Felder der Anwendung. Wurde über die Befehle **SET TABLE TITLES** und **SET FIELD TITLES** eine virtuelle Struktur erstellt, wird sie berücksichtigt.

2. Wählen Sie einen Vergleichsoperator aus dem mittleren Menü. Die Liste der Operatoren ist je nach Art des gewählten Feldes anders:



Der Sucheditor bietet zusätzlich zu den standardmäßigen Vergleichsoperatoren erweiterte Operatoren und vordefinierte Wertetypen, damit Sie rasch alle gängigen Suchen durchführen können (siehe **Vergleichsoperatoren**).

3. Typen Sie den Wert, nach dem Sie suchen wollen.
In einem Feld vom Typ Text oder Alpha können Sie für eine Suche "Beginnt mit" am Ende des Wertes das Joker-Zeichen (@) setzen.
Ist dem gewählten Feld eine Auswahlliste zugeordnet, zeigt 4D die Liste, wo Sie einen Wert auswählen können. Ist das gewählte Feld vom Typ Boolean, zeigt 4D einen Satz Optionsfelder.
4. Um eine Suche mit mehreren Suchkriterien durchzuführen, klicken Sie auf das Icon (+), um weitere Suchzeilen hinzuzufügen. 4D dupliziert den Inhalt der Zeile so oft wie erforderlich.
Für eine Suche nach Formel klicken Sie unter Windows mit gedrückter **Alt-Taste**, auf Mac OS mit gedrückter **Wahltaste** am Ende der Zeile auf das Icon (+). Weitere Informationen dazu finden Sie im Abschnitt **Suche nach Formel**.
5. Setzen Sie den passenden Verbindungsoperator (**Und, Oder, Außer**).
4D setzt standardmäßig den Operator **Und** neben die hinzugefügte Zeile.
6. Wiederholen Sie Schritt 1 bis 3, um neue Suchkriterien zu definieren.
Bei einer zusammengesetzten Suche bewertet 4D die einzelnen Suchen in der Reihenfolge, wie sie im Sucheditor erscheinen, also z.B. von oben nach unten. Die einzelnen Operatoren sind gleichwertig, d.h. **Und** ist nicht vorrangig vor **Oder**. Beachten Sie, dass sich die Reihenfolge beim Zusammensetzen mehrerer einfacher Suchen auf das Suchergebnis auswirkt.
Beim Erstellen einer zusammengesetzten Suche können Sie bestehende Teile der Suche verändern. Klicken Sie dazu in die entsprechende Suchzeile und wählen ein anderes Feld, einen anderen Operator oder tippen Sie einen neuen Wert ein.
Sie können eine einfache Suche entfernen. Klicken Sie dazu am Ende der Zeile auf das Icon (-). Beachten Sie, dass Sie das Löschen von Zeilen nicht rückgängig machen können.
7. Wählen Sie das Ziel der Suche über das Menü Auswahlaktionen: **Erzeuge neue Auswahl, Suche in Auswahl, Füge zur Auswahl hinzu, Entferne von Auswahl** (siehe vorigen Abschnitt).
8. (Optional) Um die Suche auf der Festplatte zu sichern, wählen Sie im Menü **Optionen** den Eintrag **Sichern...**
9. Klicken Sie auf die Schaltfläche **Suchen**, um die Suche zu starten.

Vergleichsoperatoren

Mit einem Suchkriterium teilen Sie 4D mit, wie es den angegebenen Wert mit dem Inhalt der Datenbank vergleichen soll. Beispielsweise verwendet die Suchbedingung „Nachname ist gleich Schmidt“ den Vergleichsoperator „ist gleich“. Es teilt 4D mit, dass es die Werte des Datenfeldes Name mit der Zeichenkette „Schmidt“ vergleichen soll.

Vergleiche, die alphanumerische Werte betreffen, ignorieren die Groß- und Kleinschreibung. Ein Suchlauf nach dem Nachnamen „Schmidt“ wird auch Datensätze finden, die „schmidt“, „SCHMIDT“ oder „schMidT“ enthalten.

Suchläufe mit den Operatoren *Enthält* und *Enthält nicht* sind stets sequentiell.

Der Operator *beinhaltet Schlüsselwort* ist nur für Datenfelder vom Typ Alpha oder Text verfügbar.

Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Vergleichsoperatoren**.

Je nach Feldtyp sind bestimmte Operatoren verfügbar:

Operator	Alp/Txt	Datum	Zeit	Bool	Num	Bild	Beschreibung
is leer	x					x	Feld enthält keine Daten
ist nicht leer	x					x	Feld enthält Daten.
ist gleich					x		Standardmäßige Vergleichsoperatoren für Zahlen
ist größer als oder gleich					x		
ist größer als					x		
ist kleiner als oder gleich					x		
ist kleiner als					x		
ist unterschiedlich von					x		
ist falsch				x			
ist wahr				x			
das ist	x	x	x				Feld enthält genau den eingegebenen Wert
ist nicht	x	x	x				Feld ist unterschiedlich zum eingegebenen Wert
beginnt mit	x	x	x				eldwert ist größer als oder gleich dem eingegebenen Wert (*).
ist nach	x	x	x				Feldwert ist größer als der eingegebene Wert (*).
ist bis zu	x	x	x				Feldwert ist kleiner als oder gleich dem eingegebenen Wert (*).
ist vor	x	x	x				Feldwert ist kleiner als der eingegebene Wert (*).
ist zwischen		x					Das erste Datum muss vor dem zweiten liegen. Die Suche beinhaltet die Felder dazwischen und die eingegebenen Datumswerte (inklusive)
ist vorher oder danach		x					Das erste Datum muss vor dem zweiten liegen. Die Suche beinhaltet nicht die Felder mit den eingegebenen Datumswerten (exklusive).
ist heute		x					Aktuelles Datum wird angezeigt
ist gestern		x					Datum des vorigen Tages wird angezeigt.
ist in aktueller/aktuellem		x					Mögliche Werte: - Woche (So-Sa) - Woche (Mo-So) - Woche (Mo-Fr) - Monat - Quartal - Jahr. Diese Werte werden ausgehend vom aktuellen Datum berechnet.
ist in den letzten		x					Mögliche Werte: - Stunden - Minuten - Sekunden. Diese Werte werden ausgehend von der aktuellen Zeit berechnet.
ist in den nächsten		x					
ist in den letzten			x				Mögliche Werte: - Stunden - Minuten - Sekunden. Diese Werte werden ausgehend von der aktuellen Zeit berechnet.
ist in den nächsten			x				
ist zwischen	x		x		x		Feldwert (inklusive) liegt zwischen den eingegebenen Werten (*).
ist zwischen (exklusive)	x		x		x		Feldwert (exklusive) liegt zwischen den eingegebenen Werten (*).
dauert exakt			x				Mögliche Werte: - Stunden - Minuten - Sekunden.
dauert nicht			x				
dauert mindestens			x				
dauert über			x				
dauert maximal			x				
ist kürzer als			x				
startet mit	x						Standardmäßige Vergleichsoperatoren für Text
endet mit	x						
enthält	x						
enthält nicht	x						
enthält Wort/Wörter	x					x	Suche nach Schlüsselwort/wörtern. Sie können wählen zwischen der Option "alle Wörter" (Feld muss alle eingegebenen Wörter enthalten) oder "einige Wörter" (Feld muss mindestens eins der eingegebenen Wörter enthalten).
enthält nicht Wort/Wörter	x					x	
belegt maximal						x	Suche basiert auf Bildgröße (es gibt verschiedene Einheiten: Bytes, KB, MB, GB)
belegt mindestens						x	

(*) Bei Zeichenketten richten sich Suchen nach dem Alphabet (es gilt a < b).

Beispiel: Eine Suche vom Typ *Name ist nach "don"* findet Donna, Don Juan, Schmid, usw., aber nicht Anton oder Dominik.

Jokerzeichen (@)

Um Suchläufe einfacher zu gestalten, besitzt 4D ein Jokerzeichen (@), das ein oder mehrere Zeichen bei einem Suchlauf innerhalb von alphanumerischen Datenfeldern oder einem Textfeld ersetzen kann. Suchen Sie z.B. nach allen Personen mit Namen „Belmondo“ in einem Datenfeld, können Sie den Suchwert unterschiedlich angeben:

Suchen nach: Findet

Bel@	Alle Werte, die mit „Bel“ beginnen
@do	Alle Werte, die mit „do“ enden
Bel@do	Alle Werte, die mit „Bel“ beginnen und mit „do“ enden
@elm@	Alle Werte, die „elm“ beinhalten


Hinweis: Sie können den Joker mit dem Suchkriterium *beinhaltet Schlüsselwort* kombinieren, z.B. *„Hinweise beinhaltet Schlüsselwort 'anti@'“*.

Einfache und zusammengesetzte Suchläufe

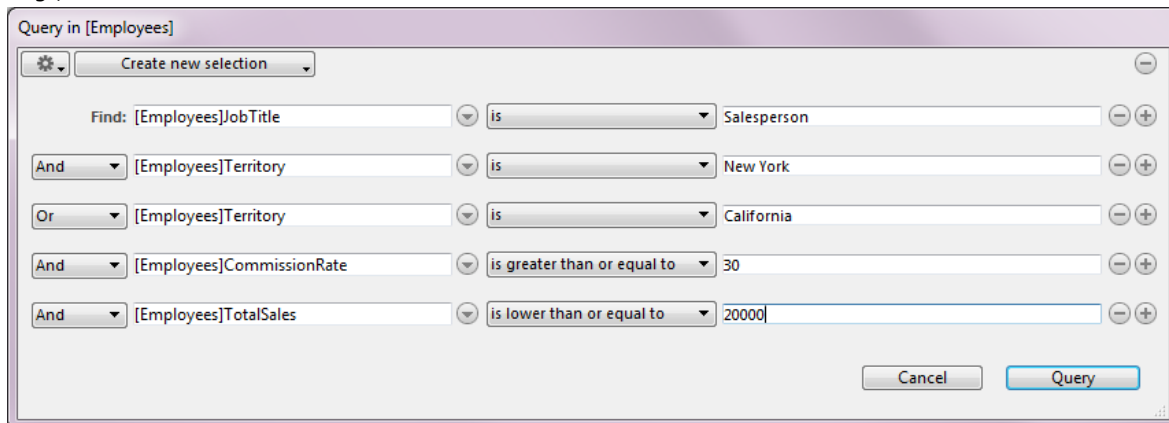
Sie können in einem oder mehreren Datenfeldern suchen. Eine Suche in einem einzigen Datenfeld heißt einfacher Suchlauf. So ist beispielsweise das Suchkriterium „Nachname ist gleich Schmidt“ ein einfacher Suchlauf. Bei einfachen Suchläufen untersucht 4D nur die Inhalte eines Datenfeldes in der Anwendung.

Ein Suchlauf in zwei oder mehr Datenfeldern heißt zusammengesetzter Suchlauf bezeichnet. Bei diesem Suchlauf verbinden Sie einzelne Suchkriterien mit einem logischen Operator. Der logische Operator teilt 4D mit, wie es die Ergebnisse der einzelnen Suchläufe miteinander verbinden soll. Es gibt drei logische Operatoren:

- **Und:** Dieser Operator findet alle Datensätze, die zwei Bedingungen gleichzeitig erfüllen. Beispielsweise bildet das Suchkriterium „Suche alle Mitarbeiter, die in der Entwicklungsabteilung arbeiten und die über 3.000 € verdienen“ eine gemeinsame Teilmenge aus den beiden Suchläufen nach Mitarbeitern der Entwicklungsabteilung UND nach Gehältern von über 3.000 € pro Monat.
- **Oder:** Dieser Operator findet alle Datensätze, auf die eine von zwei Suchbedingungen zutrifft. Beispielsweise findet die Suchbedingung „Suche alle Mitarbeiter der Entwicklungsabteilung ODER alle mit einem Gehalt von über 3.000 €“ sowohl alle Mitarbeiter der Entwicklungsabteilung, als auch die Besserverdienenden mit Gehältern über 3.000 € anderer Abteilungen.
- **Außer:** Der Operator entspricht „nicht“. So findet die Suchbedingung „Suche alle Ingenieure, jedoch NICHT diejenigen, die über 3.000 € verdienen“ nur die Ingenieure, die weniger als 3.000 € monatlich verdienen.

Um eine Suche mit mehreren Suchkriterien durchzuführen, können Sie so oft wie erforderlich auf das Icon  klicken.

Mit den logischen Operatoren können Sie komplexe Suchbedingungen erstellen, wie z.B. „Finde alle Vertriebsmitarbeiter in New York ODER Kalifornien, die über 30 % Provision haben UND ein Verkaufsvolumen unter 20.000“. Das folgende Bild zeigt, wie dieser Suchlauf im Sucheditor definiert wird:



Wenn diese Suche ausgeführt wird, findet 4D alle Mitarbeiter des Vertriebs in New York und Kalifornien, die eine hohe Provision bei niedrigem Verkaufsvolumen erhalten. Bei jeder Suchmethode finden Sie weitere Anwendungsbeispiele für Vergleichs- und logische Operatoren.

Suche nach Formel

Sie können eine **Suche nach Formel** verwenden, um Datensätze über eine Berechnung oder Datenbearbeitung zu finden. Ihre Formel muss als Suchergebnis für jeden Datensatz WAHR oder FALSCH ergeben. Sie können jede Funktion der Programmiersprache sowie vom Entwickler speziell dafür vorgesehene Projektmethoden verwenden.

Suche nach Formel eignet sich zur Eingabe von Suchbedingungen für folgende Operationen:

- Operationen oder Auswertungen in alphanumerischen Zeichenketten
- Datumsberechnungen
- Arithmetische Berechnungen

Es folgen ein paar Anwendungsbeispiele:

- Die folgende Formel sucht nach Datensätzen, in denen die letzten sieben Zeichen der Telefonnummer dem Wert „2524444“ haben.

```
Substring([Angestellte]Telefonnummer;4;7)="2524444"
```

- Die folgende Formel findet Personen, die zum aktuellen Datum geboren wurden, unabhängig vom Jahr:

```
(Day of(Current date)=Day of([Angestellte]Geburtstag)) & (Month of(Current date)=Month of([Angestellte]Geburtstag))
```

- Die folgende Formel teilt den jährlichen Umsatz durch die Lebenshaltungskosten und findet Datensätze, deren berechneter Wert größer als 1.000 ist:

```
([Statistik]Jahresumsatz/[Statistik]Lebenshaltungskosten)>1000
```

Dagegen ist die Formel:


```
[Statistik]Jahresumsatz/[Statistik]Lebenshaltungskosten
```

nicht richtig, da sie einen numerischen Wert ausgeben würde und nicht WAHR oder FALSCH.

Sie können nur Formeln eingeben, die eine logische Zeile lang sind. Das bedeutet, Sie können nicht die Zeilenschaltung betätigen und eine zweite Zeile schreiben. Der Editierbereich bricht eine Anweisung in die nächste Zeile um, wenn er zu lang ist. Wenn Sie eine Formel benötigen, die länger als eine Zeile ist, müssen Sie die Formel als Projektmethode schreiben und diese anschließend in der Formel aufrufen.

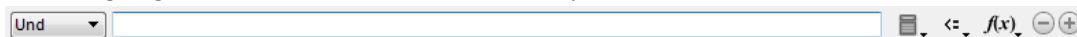
Eine Suche nach Formel ausführen


Suchläufe nach Formel werden im standardmäßigen Sucheditor angelegt. Es ist ein Suchmodus für Fortgeschrittene.


Für eine Suche nach Formel öffnen Sie den **Sucheditor** und klicken dann unter Windows mit gedrückter **Alt-Taste**, auf Mac OS mit gedrückter **Wahltaste** am Ende der Zeile auf das Icon .

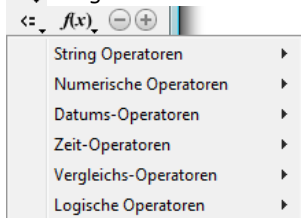
Hinweis: Bei einer einzeiligen Suche nach Formel löschen Sie die erste Zeile, die standardmäßig hinzugefügt wird.


Die hinzugefügte Zeile enthält dann zusätzliche DropDown-Menüs:

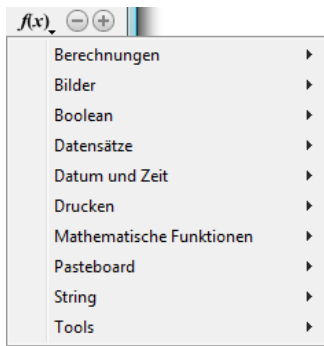


-  zeigt eine hierarchische Liste der Tabellen und Felder der Datenbank, deren Typ mit einer Suche nach Formel kompatibel ist.

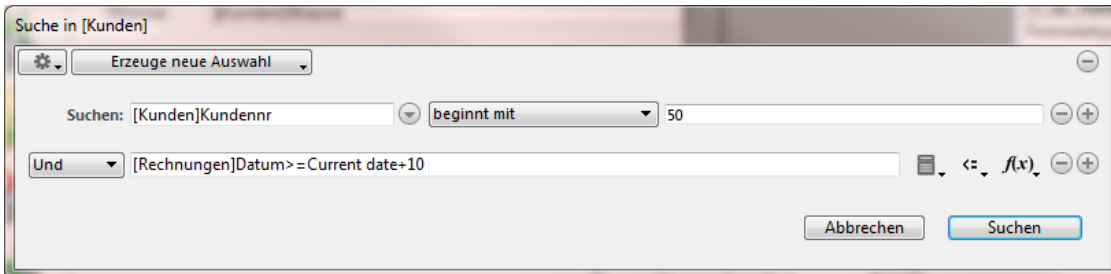
-  zeigt eine hierarchische Liste der Operatoren, die für eine Suche nach Formel verwendbar sind.



-  zeigt eine Liste der 4D Funktionen, die standardmäßig für eine Suche nach Formel verfügbar sind. Über den Befehl **SET ALLOWED METHODS** können Sie Projektmethoden in diese Liste einfügen.



Über dieses Feature lassen sich Kriterien für Suche nach Formel mit den standardmäßigen Suchkriterien kombinieren:



Da sie in dieses Dialogfenster integriert sind, haben Suchen nach Formel dieselbe Funktionalität wie Standardsuchen:

- Über das Menü **Erzeuge neue Auswahl** lässt sich die Behandlung des Suchergebnisses wählen (**Menü Auswahlaktion**),
- Die Suche wird im Menü **Letzte Suchen** hinzugefügt (**Menü Auswahlaktion**),
- Sichern und Laden über eine Suchdatei (siehe **Menü Optionen**).

Nach Beispiel suchen

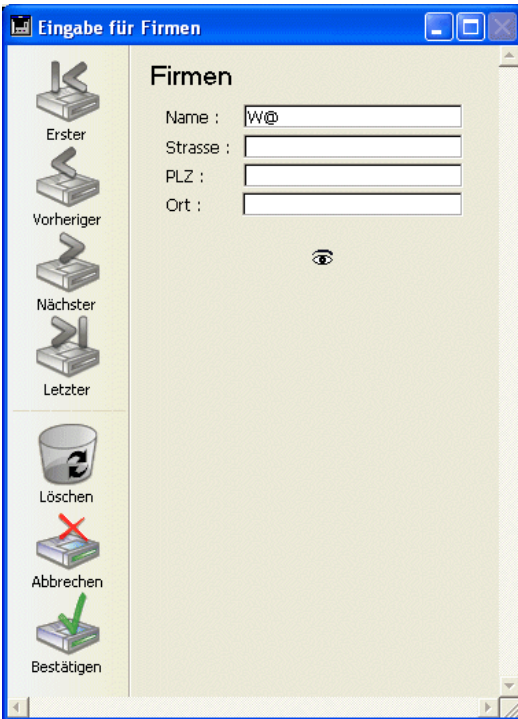
Die Option Nach Beispiel suchen eignet sich besonders, um viele Suchläufe gleichzeitig auszuführen. In dieser Suchmethode verwenden Sie das aktuelle Eingabeformular, um Suchwerte einzugeben. Sie können jedoch nur in den Datenfeldern der aktuellen Tabelle suchen. Nach Beispiel suchen führt sowohl indizierte als auch sequentielle Suchläufe aus.

Das gezeigte Fenster ist nur ein Beispiel. Ihr aktuelles Eingabeformular wird stets als Fenster Suche nach Beispiel verwendet. Sie können seine Darstellung durch Ändern des Eingabeformulars steuern.

Sie können in diesem Dialogfenster Vergleichsoperatoren verwenden. Um beispielsweise den Operator „ist gleich“ zu gebrauchen, geben Sie den gesuchten Wert in das jeweilige Datenfeld ein. Benötigen Sie einen anderen Vergleichsoperator, setzen Sie vor den Wert das entsprechende Zeichen:

Vergleichsoperator	Zeichen	Beispiel
ist nicht gleich	#	#Marketing
ist größer als	>	>30000
ist größer als oder gleich	>=	>=30000
ist kleiner als	<	<30000
ist kleiner als oder gleich	<=	<=30000

Sie können einen Suchlauf „Beginnt mit“ aufrufen, indem Sie an den gesuchten Wert ein Jokerzeichen (@) anhängen. Die folgende Abbildung zeigt das Fenster vor einem Suchlauf nach Mitarbeiternamen, die mit „W“ beginnen:



Sie können auch eine zusammengesetzte Suche definieren. Geben Sie dazu einen Wert in ein anderes Feld ein. Beide Werte werden mit dem Operator Und verbunden, z.B. Name beginnt mit W und Ort ist München.

Gehen Sie folgendermaßen vor:

1. Wählen Sie in der Designumgebung im Menü **Datensätze>Suchen** und dann im DropDown-Menü den Eintrag **Nach Beispiel suchen**
ODER
Klicken Sie in der Werkzeugleiste auf die Schaltfläche **Suchen** und wählen im DropDown-Menü den Eintrag **Nach Beispiel suchen**.
4D zeigt das Eingabeformular der aktuellen Tabelle als ein Fenster Suche nach Beispiel. Nur die Schaltflächen **Bestätigen** und **Abbrechen** sind aktiv.
2. Geben Sie einen zu suchenden Wert in das entsprechende Datenfeld ein.
Suchen Sie beispielsweise nach Personen mit dem Nachnamen Wagner, geben Sie in das Datenfeld Name „Wagner“ ein.
Um einen Vergleichsoperator zu verwenden, setzen Sie vor den Wert ein Operatorzeichen (siehe o.a. Tabelle). Für eine Suche „Beginnt mit“ setzen Sie das Symbol „@“ an das Ende der Zeichenkette.
Hinweis: Mit diesem Editor können Sie keine Volltextsuche ausführen.
3. Für einen zusammengesetzten Suchlauf geben Sie Werte in weitere Datenfelder ein.
4D verwendet den logischen Operator Und, wenn Sie Werte in mehr als ein Datenfeld eingeben.
4. Klicken Sie auf die Schaltfläche **Bestätigen** oder drücken die **Eingabetaste** auf dem Zahlenblock Ihrer Tastatur, um mit der Suche zu beginnen. Um die Suche abzubrechen, klicken Sie auf die Schaltfläche **Abbrechen** oder drücken die **Escape** oder **Esc** Taste.
Die Datensätze, auf welche die Suchkriterien zutreffen, erscheinen im aktuellen Ausgabeformular.

Diese Option eignet sich zum schnellen Finden und Ändern von Datensätzen. Sie verwendet wie „Nach Beispiel suchen“ das aktuelle Eingabeformular als Suchfenster. Sie legen die Suche genauso wie bei „Nach Beispiel suchen“ fest.

Wird ein Datensatz oder eine Auswahl von Datensätzen gefunden, zeigt 4D den ersten im Eingabeformular an. Wenn Sie den Datensatz bestätigen, kehren Sie in das Ausgabeformular zurück.

- Mit Suchen und Ändern werden gefundene Datensätze im aktuellen Ausgabeformular angezeigt, d.h. die Datensätze innerhalb der neuen Auswahl erscheinen als Liste.
- Mit Suchen und Ändern wird der aktuelle Datensatz der neuen Auswahl auch im aktuellen Eingabeformular angezeigt. So können Sie direkt die gewünschten Werte ändern. Enthält die neue Auswahl mehrere Datensätze, können Sie im Eingabeformular auf die Schaltfläche zum Navigieren klicken, um durch jeden Datensatz zu scrollen und die Werte der Reihe nach zu ändern.

Gehen Sie folgendermaßen vor:

1. Wählen Sie in der Designumgebung im Menü **Datensätze>Suchen** und dann im DropDown-Menü den Eintrag **Suchen und ändern**
ODER
Klicken Sie in der Werkzeugleiste auf die Schaltfläche **Suchen** und wählen im DropDown-Menü den Eintrag **Suchen und ändern**. 4D zeigt das aktuelle Eingabeformular als ein Suchfenster an. Nur die Schaltflächen **Bestätigen** und **Abbrechen** sind aktiv.
2. Legen Sie Ihre Suche an wie für eine Suche nach Beispiel.
Weitere Informationen dazu finden Sie im Abschnitt **Nach Beispiel suchen**.
3. Klicken Sie auf die Schaltfläche **Bestätigen** oder drücken die **Eingabetaste** auf dem Zahlenblock Ihrer Tastatur, um die Suche zu starten.
4D führt die Suche aus, stellt die neue aktuelle Auswahl zusammen und zeigt den ersten Datensatz im Eingabeformular an, der den Suchkriterien entspricht.
4. Klicken Sie auf die Steuerschaltflächen, um zu anderen Datensätzen der aktuellen Auswahl zu springen.

Datensätze sortieren

-  Funktionsweise
-  Sortiereditor

Die Funktion Sortieren ordnet die Datensätze nach Werten in der Tabelle neu. In der Regel sortieren Sie Ihre Datensätze:

- Um Datensätze auf dem Bildschirm in einer bestimmten Reihenfolge anzuzeigen
- Vor dem Drucken von Berichten oder Etiketten
- Vor dem Darstellen der Daten in Diagrammen

Wenn Sie in eine neue Datenbank Daten eingeben, speichert 4D die Datensätze in der Reihenfolge ab, in der sie eingegeben oder importiert werden. Zeigen Sie Datensätze in einem Ausgabeformular an oder drucken sie aus, erscheinen sie in dieser Reihenfolge. Oft ist aber eine andere Reihenfolge sinnvoll. Zum Beispiel wollen Sie eine Liste von Namen in einem Bericht alphabetisch anzeigen - Sortieren Sie nach dem Datenfeld Nachname, werden die Datensätze alphabetisch nach dem Nachnamen geordnet.

Sie können in einem Eingabe- oder Ausgabeformular sortieren. Im Eingabeformular wird der erste Datensatz der neuen Sortierfolge angezeigt. Im Ausgabeformular erscheinen alle Datensätze in der entsprechenden Sortierfolge.

4D führt indiziertes Sortieren sehr schnell durch. Wenn Sie nur nach einem Datenfeld sortieren und dieses Datenfeld indiziert ist, verwendet 4D den Index.

Sie können für eine Sortierung den Sortiereditor verwenden. Hier können Sie die Sortierkriterien manuell einstellen.

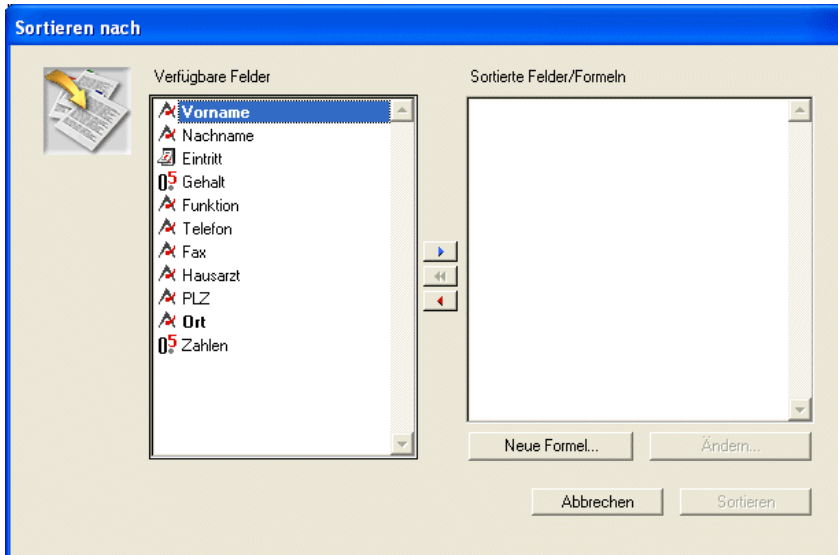
In der Anwendungsumgebung können Sie eine Sortierung direkt mit den 4D Befehlen **ORDER BY** oder **ORDER BY FORMULA**. Sie können über diese Befehle aber auch den Sortiereditor anzeigen.

In der Designumgebung wählen Sie im Menü **Datensätze** den Befehl **Sortieren**.

Sortieren sortiert die aktuelle Auswahl der Datensätze um. Diese Sortierung ist temporär und gilt nur für die aktuelle Auswahl; sie beeinflusst nicht die Reihenfolge, in der Datensätze in der Anwendung gespeichert sind. Die Sortierung wird normalerweise unmittelbar nach einer Suche oder vor dem Drucken eines Berichts oder von Etiketten verwendet.

Beschreibung

Wählen Sie in der Designumgebung im Menü **Datensätze** den Befehl **Sortieren** oder klicken Sie in der Werkzeugleiste auf die Schaltfläche **Suchen** und wählen im DropDown-Menü den Eintrag **Sortieren**.



Der Sortiereditor enthält folgende Bereiche und Befehle:

- **Liste der Datenfelder:** Dieser Bereich zeigt die Datenfeldnamen in der aktuellen Tabelle. Indizierte Datenfelder sind fett dargestellt. Sie können nach Datenfeldern aus Untertabellen und verknüpften Tabellen sortieren (vorausgesetzt, die Verknüpfung ist automatisch). Wollen Sie ein Datenfeld aus einer verknüpften Tabelle oder Untertabelle verwenden, klicken Sie dazu auf das Pluszeichen, bzw. den Pfeil vor dem verknüpften Feld. Die Datenfelder der verknüpften Tabelle werden angezeigt.
Hinweis: Tabellen und Felder mit der Eigenschaft Unsichtbar erscheinen nicht in der Liste. Weitere Informationen dazu finden Sie in den Abschnitten **Tabelleneigenschaften** und **Feldeigenschaften**.
- **Sortierbereich:** Dieser Bereich zeigt die Sortierdatenfelder oder Sortierformeln und die Sortierrichtung an. Mit den Pfeilen am rechten Rand legen Sie fest, ob Sie in aufsteigender oder absteigender Reihenfolge sortieren
- **Schaltfläche Neue Formel:** Damit können Sie eine Formel als Sortierkriterium schreiben. Sie verwenden eine Formel, wenn Sie nach etwas sortieren wollen, das kein Datenfeld ist, wie ein berechneter Wert oder der Teil eines Datenfeldes. Zum Beispiel können Sie nach einer berechneten Fehlerrate für Artikel oder nach den letzten vier Stellen einer Artikelnummer sortieren.
- **Schaltfläche Ändern:** Damit erscheint das ausgewählte Sortierkriterium im Formeleditor. Ist das ausgewählte Kriterium eine Formel, wird die Formel zur Bearbeitung angezeigt. Ist das Kriterium ein Datenfeld, erscheint der Datenfeldname im Bearbeitungsfenster des Formeleditors.
- **Schaltflächenreihe:** Damit können Sie Felder in der Sortierliste hinzufügen oder entfernen. Sie können die Felder aber auch per Drag-and-Drop bewegen.
- **Schaltfläche Abbrechen:** Damit brechen Sie das Sortieren ab und kehren zum gerade verwendeten Formular zurück.
- **Schaltfläche Sortieren:** Damit führen Sie den Sortiervorgang aus. 4D führt die Sortierung aus und zeigt die Liste der sortierten Datensätze. Starten Sie die Sortierung über ein Eingabeformular, erscheint der erste Datensatz der aktuell sortierten Auswahl in diesem Formular.

Hinweis: **Schnellberichte** können Datensätze sortieren, die in einem Schnellbericht erscheinen.

Sortierebenen

Sie können Datensätze nach bis zu 30 verschiedenen Datenfeldern oder Formeln sortieren. Jedes Sortierfeld oder -formel ist eine Sortierebene. Ein aufsteigendes Sortieren in zwei Ebenen nach den Datenfeldern Nachname und Vorname kann z.B. folgende Liste ergeben:

Abele, Andrea
Abele, Anton
Abele, Arthur
[...]
Zeiler, Ellen
Zenker, Bernd

Beim Sortieren von Feldinhalten berücksichtigt 4D nicht Groß- und Kleinschreibung ("Schmid" = "schmid") oder diakritische Zeichen ("Aá" = "Aa"). Verwenden Sie dagegen eine internationale Version von 4D und berücksichtigt Ihr Betriebssystem diakritische Zeichen ("Áá" ; "Aa"), werden beim Sortieren auch diakritische Zeichen berücksichtigt.


Sortierlauf ändern

Sie können Felder oder Formeln im Sortierbereich jederzeit umstellen oder ganz entfernen.


Um Sortierebenen umzustellen, verwenden Sie Drag-and-Drop

1. Markieren Sie im Sortierbereich mit gedrückter Maustaste die entsprechende Ebene und ziehen Sie diese an die gewünschte Stelle. Sie wird direkt darüber eingefügt.

Um eine Sortierebene zu entfernen:

1. Markieren Sie im Sortierbereich die entsprechende Sortierebene
 2. Drücken Sie die **Rückschrittaste**
ODER
Klicken Sie auf die Schaltfläche **Löschen** .
- Die markierte Ebene wird aus dem Bereich entfernt.

Um alle Sortierebenen zu entfernen:

1. Klicken Sie auf die Schaltfläche **Alles Entfernen** .
- Alle Ebenen werden aus dem Bereich entfernt.

Aufsteigende und absteigende Reihenfolge

Sie können für jedes Sortierfeld bzw. jede Formel eine aufsteigende oder absteigende Reihenfolge festlegen.

Aufsteigendes Sortieren heißt von A bis Z oder Klein zu Groß ▲, absteigendes Sortieren heißt von Z bis A, Groß zu Klein, Spät zu Früh ▼.

Wenn Sie nach mehr als einer Ebene sortieren, können Sie aufsteigende und absteigende Sortierrichtung beliebig mischen. Mehrstufiges Sortieren kann auch Datenfelder und Formeln sowie aufsteigende und absteigende Sortierfolge miteinander mischen.

Um die Sortierreihenfolge einer Ebene zu ändern:

1. Klicken Sie im Sortierbereich auf den Sortierpfeil am rechten Rand

Nach einer Formel sortieren

Sie können auch nach einer Formel sortieren. Zum Beispiel sortiert die folgende Formel nach dem Monat innerhalb des Datenfeldes Geburtsdatum.



```
Month of([Kinder]Geburtsdatum)
```

Um eine Sortierformel zu erstellen:

1. Klicken Sie im Sortiereditor auf die Schaltfläche **Neue Formel**.
Auf dem Bildschirm erscheint der .
Hier können Sie eine Formel anlegen, die Werte ausgibt, nach denen sortiert werden soll. Die Formel kann Werte beliebiger Datentypen zurückgeben. Weitere Informationen dazu finden Sie im Abschnitt .
2. Klicken Sie auf die Schaltfläche **OK**, wenn Ihre Formel fertig ist.
4D zeigt die Formel im Sortierbereich an. Die Formel erscheint mit einem Sortierrichtungspfeil im Sortierbereich.

Um eine Formel zu ändern oder eine Formel in einen bereits im Sortierbereich vorhandenen Datenfeld anzulegen, Wählen Sie im Sortierbereich die entsprechende Formel bzw. das Datenfeld aus und klicken am unteren Rand auf die Schaltfläche **Ändern** oder doppelklicken Sie im Sortierbereich auf die Formel bzw. das Datenfeld.

Formeleditor

-  Formeleditor einsetzen
-  Beschreibung des Formeleditors

Formeleditor einsetzen

Sie können den Formeleditor in der 4D Datenbank in verschiedenen Kontexten verwenden:

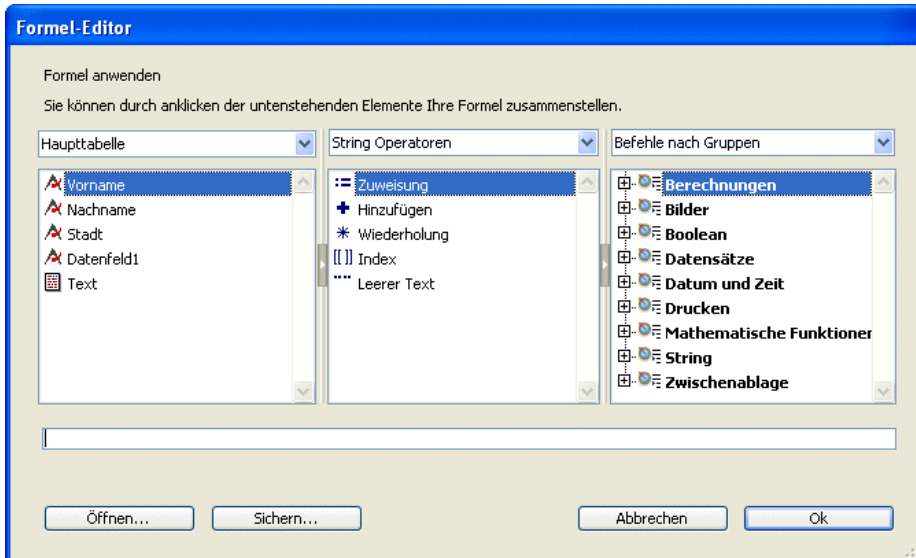
- Zur globalen Aktualisierung auf eine Datensatzauswahl (siehe "Globale Updates" unter **Datensätze bearbeiten**).
- Zum Ausführen von Sortierläufen nach Formel (siehe "Nach einer Formel sortieren" in **Sortiereditor**),
- Zum Berechnen einer Spalte im Schnellbericht (siehe **Formeln hinzufügen**),

In der Anwendungsumgebung können Sie eine Formel zur Aktualisierung direkt über den 4D Befehl **EXECUTE FORMULA** ausführen oder den Formeleditor über den 4D Befehl **EDIT FORMULA** anzeigen.

Hinweis: Sie können eine Formel auch beim Ausführen einer Suche in den Datensätzen anwenden. Dazu verwenden Sie nicht den Formeleditor, sondern die erweiterte Funktion im Sucheditor (siehe **Suche nach Formel**).

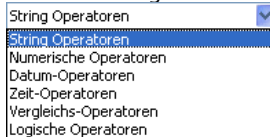
Beschreibung des Formeleditors

Der Formeleditor vereinfacht das Schreiben von Formeln. Sie können beispielsweise auf Feldnamen, Operatoren und Routinen, sowie Projektmethoden klicken, um sie in Ihre Formel einzufügen. Klicken Sie auf ein Element, erscheint es automatisch im Eingabebereich, wo Sie es mit den Standardaktionen Ausschneiden/Kopieren/Einsetzen verändern können. Sie können Elemente auch direkt eingeben oder per Drag&Drop aus einer der Listen in den Eingabebereich setzen.



Der Formeleditor ist in folgende Bereiche gegliedert:

- **Liste der Tabellen und Felder:** Dieser Bereich zeigt die Felder der Tabelle. Im darüberliegenden DropDown-Menü wählen Sie die Tabelle der Felder aus. Sie können Felder der aktuellen Tabelle, von verknüpften Tabellen oder aus allen Tabellen verwenden.
Hinweis: Tabellen und Felder mit der Eigenschaft *Unsichtbar* erscheinen nicht in der Liste. Weitere Informationen dazu finden Sie in den Abschnitten **Tabelleneigenschaften** und **Feldeigenschaften**.
- **Liste der Operatoren:** Hier wählen Sie die Operatoren für die Formel. Sie sind nach Themen geordnet, welche Sie im darüberliegenden Menü aufrufen können:



Jedes Thema enthält alle für den Datentyp bzw. die Operation verfügbaren Operatoren. So ist z.B. der Operator `:=` für alle Datentypen verfügbar. Die ausführliche Beschreibung der einzelnen Operatoren finden Sie im nächsten Abschnitt.

- **Liste der Routinen:** Dieser Bereich enthält die für Formeln verfügbaren Befehle, sowie Projektmethoden, die der Entwickler für Benutzer freigegeben hat. Im darüberliegenden Menü definieren Sie, ob die Befehle nach Gruppen oder alphabetisch geordnet werden. Die ausführliche Beschreibung der Befehle finden Sie im Handbuch *4D Programmiersprache*.
In der Regel müssen Projektmethoden, die in Formeln verwendbar sind, zuvor mit dem 4D Befehl **SET ALLOWED METHODS** deklariert sein. Designer und Administrator haben dagegen standardmäßig vollen Zugriff auf die 4D Befehle und Benutzermethoden im Formeleditor. Die Zugriffskontrolle lässt sich auch für alle Benutzer deaktivieren. Diese Option wird auf der **Seite Sicherheit** der Datenbank-Eigenschaften eingerichtet.

Operatoren für Formeln

Nachfolgend sind die Operatoren aufgelistet, die im Formeleditor verfügbar sind. Eine ausführliche Beschreibung dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Operatoren**.

- **String Operatoren**

A und B sind Zeichenketten; N ist die Anzahl.

Operator	Einsatz	Beschreibung
<code>:=</code> Zuweisung	A:=B	Weist A den Wert B zu
<code>+</code> Verbindung	A+B	Gibt AB zurück
<code>*</code> Wiederholung	A*N	Wiederholt N-mal den Wert A
<code>[[]]</code> Indizes	<code>[[A]]N</code>	Gibt das N-te Zeichen von A zurück
<code>""</code> Leerer String	<code>""</code>	Setzt ein Paar Anführungszeichen

- **Numerische Operatoren**

X und Y sind Zahlen.

Operator	Einsatz	Beschreibung
:= Zuweisung	X:=Y	Weist X den Wert Y zu
+ Addition	X+Y	Gibt X plus Y zurück
- Subtraktion	X-Y	Gibt X minus Y zurück
* Multiplikation	X*Y	Gibt X multipliziert mit Y zurück
/ Division	X/Y	Gibt X dividiert durch Y zurück
\ Ganzzahlige Division	X\Y	Gibt die ganzzahlige Division von X durch Y zurück (X und Y müssen Ganzzahlen sein)
% Modulo	X%Y	Dividiert X durch Y und gibt den Restwert zurück
^ Exponent	X^Y	Gibt Y als Hochzahl von X an

Hinweis: Der Operator Modulo % gibt signifikante Werte mit Zahlen aus dem Bereich Lange Ganzzahl zurück (von -2^{31} bis $+2^{31}$ minus 1). Verwenden Sie bei Restwertberechnungen mit Zahlen außerhalb dieses Bereichs die 4D Funktion **Mod**.

• Datumsoperatoren

D1 und D2 sind Datumsangaben. N ist die Anzahl.

Operator	Einsatz	Beschreibung
:= Zuweisung	D1:=D2	Weist D1 den Wert D2 zu
+ Addition	D1+N	Gibt D1 plus N Tage zurück
- Differenz	D1-D2 oder D1-N	Gibt Anzahl der Tage zwischen D1 und D2 zurück Gibt D1 minus N Tage zurück
!//! Leeres Datum	!00/00/00!	Fügt ein leeres Datum ein

• Zeitoperatoren

H1 und H2 sind Zeitangaben; N ist eine Zahl.

Operator	Einsatz	Beschreibung
:= Zuweisung	H1:=H2	Weist H1 den Wert H2 zu
+ Addition	H1+H2 oder H1+N	Gibt H1 plus H2 zurück Gibt H1 plus N Sekunden zurück, ausgedrückt in Sekunden ab Mitternacht
- Subtraktion	H1-H2 oder H1-N	Gibt H1 minus H2 zurück Gibt H1 minus N Sekunden zurück, ausgedrückt in Sekunden ab Mitternacht
* Multiplikation	H1*N	Gibt H1 multipliziert mit N zurück, ausgedrückt in Sekunden ab Mitternacht
/ Division	H1/N	Gibt H1 dividiert durch N zurück, ausgedrückt in Sekunden ab Mitternacht
\ Integer Division	H1\N	Gibt die ganzzahlige Division von H1 durch N zurück, ausgedrückt in Sekunden ab Mitternacht
% Modulo	H1%N	Dividiert H1 durch N und gibt den Restwert zurück
?::? Leere Zeit	? 00:00:00?	Fügt eine leere Zeit ein

• Vergleichsoperatoren

Z1 und Z2 können vom Typ String, Zahl, Datum oder Zeit sein.












Operator	Einsatz	Beschreibung
= gleich	Z1=Z2	Gibt wahr zurück, wenn Z1 gleich Z2 ist
# ungleich	Z1#Z2	Gibt wahr zurück, wenn Z1 ungleich Z2 ist
> Größer als	Z1>Z2	Gibt wahr zurück, wenn Z1 größer als Z2 ist
>= Größer als oder gleich	Z1>=Z2	Gibt wahr zurück, wenn Z1 größer oder gleich Z2 ist
< Kleiner als	Z1<Z2	Gibt wahr zurück, wenn Z1 kleiner als Z2 ist
<= Kleiner als oder gleich	Z1<=Z2	Gibt wahr zurück, wenn Z1 kleiner als oder gleich Z2 ist

• Logische Operatoren

B1 und B2 müssen Boolean sein (Ausdruck ist WAHR oder FALSCH)

Operator	Einsatz	Beschreibung
& UND	B1 & B2	Gibt wahr zurück, wenn B1 und B2 wahr sind
ODER	B1 B2	Gibt wahr zurück, wenn B1 oder B2 wahr ist

Schnellberichte

-  Einleitung
-  Schnellberichte verwalten
-  Schnellberichteditor
-  Grafische Attribute eines Berichts
-  Datensätze sortieren
-  Berechnungen hinzufügen
-  Anzeigeformate definieren
-  Zeilen und Spalten aus- und einblenden
-  Kopf- oder Fußzeilen hinzufügen
-  Schnellbericht ausführen
-  4D Code generieren

Einleitung

Eine der wichtigsten Aufgaben in der Datenverwaltung ist das Erstellen von Berichten. Mit dem Schnellberichteditor erstellen Sie ad-hoc Berichte in der Design- und Anwendungsumgebung. Die andere Möglichkeit ist der **Formulareditor**. Damit gestalten Sie Berichte in der Designumgebung (Sie können diese anschließend in der Anwendungsumgebung verändern). Verwenden Sie ein Ausgabeformular für Berichte, die ein komplexes Layout erfordern, Programmierung oder Bilder enthalten. Weitere Informationen dazu finden Sie im Kapitel **Ausgabeformulare und Berichte**.

Mit dem Schnellberichteditor können Sie:

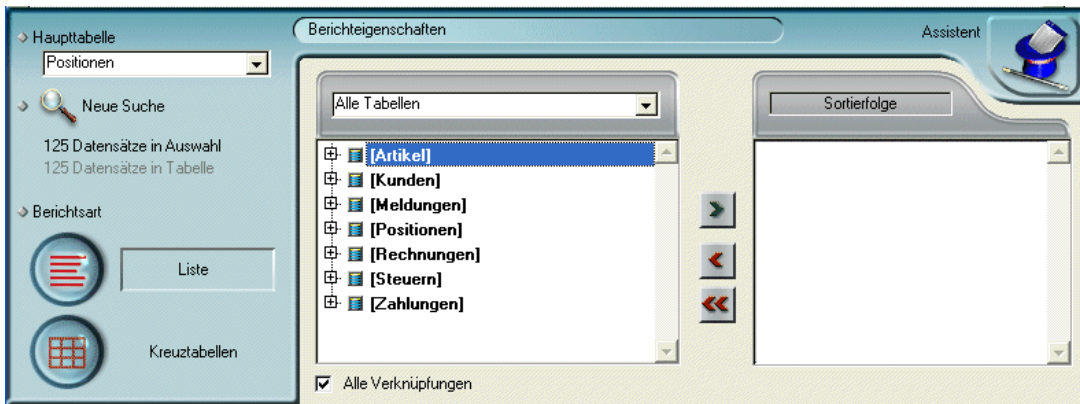
- Datensätze als Liste erstellen
- Umbrüche einfügen
- Kreuztabellen-Berichte anlegen
- Berechnungen in Summenzeilen durchführen
- Den Bericht mit Schriftarten und -stilen gestalten
- Für jede Zelle Ränder und Hintergrundfarbe definieren
- Gestaltete Berichte auf der Festplatte sichern und erneut aufrufen
- Zwischen verschiedenen Ausgabearten wählen, z.B. HTML oder Textdatei, 4D View Bereich, Drucken oder auf Festplatte sichern

Der Schnellberichteditor lässt sich auch per Programmierung über die Befehle des Kapitels **Schnellbericht** steuern.

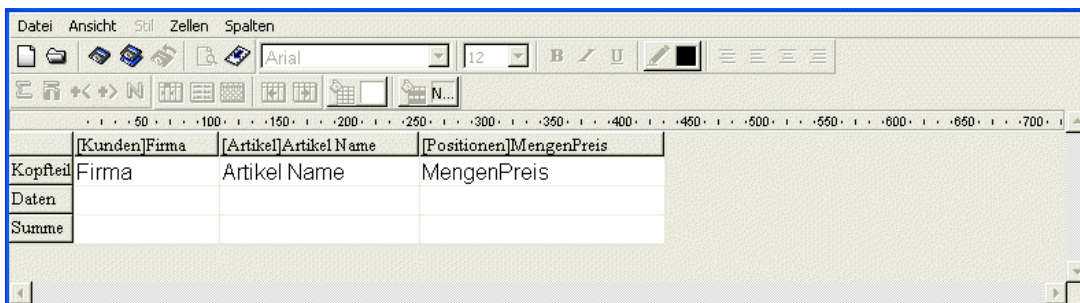
Schnellberichteditor

Der Schnellberichteditor ist in zwei Hauptbereiche gegliedert:

Datenteil des Schnellberichts: Hier definieren Sie die Berichtart und wählen die gewünschten Daten aus. Über das Icon in der rechten oberen Ecke können Sie auch den Berichtsassistenten aufrufen:

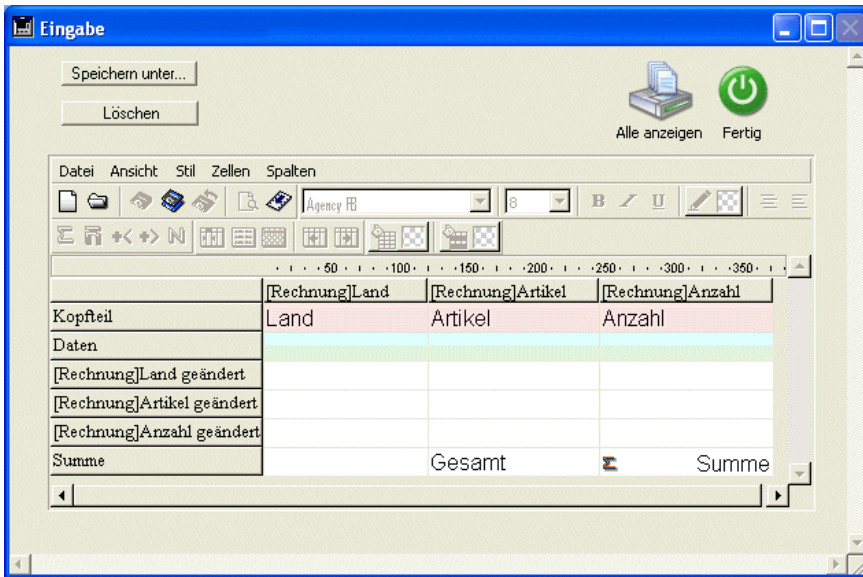


Bereich mit dem **aktuellen Bericht**:



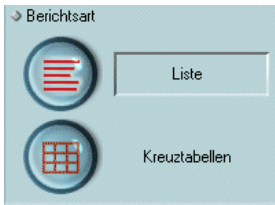
Er zeigt die aktuelle Anordnung von Spalten und Zeilen, die Sortierreihenfolge, etc., wenn ein Schnellbericht in ein Formular eingefügt wird. Er hat eigene Werkzeug- und Menüleisten sowie Kontextmenüs.

Dieser Bereich könnte folgendermaßen aussehen:



Modus Liste und Kreuztabelle

Der Schnellberichteditor bietet zwei Modi für Berichte: Liste und Kreuztabellen.



• Liste

Dies ist der Standardmodus. Er zeigt Datensätze als Liste mit Umbrüchen an der Stelle, wo Summen berechnet wurden. Hierzu ein Beispiel:

Vorname	Nachname	Abteilung	Gehalt
Franz	Becker	Technik	4.000,00 €
Peter	Eigenrauch		5.000,00 €
Hubert	Hagen		4.000,00 €
Alexander	Maisberger		6.500,00 €
Tom	Müller		4.500,00 €
Martin	Redlin		6.000,00 €
Franz	Westermeier		5.500,00 €
		Gehalt für Abteilung Technik	35.500,00 € 7
Birgitte	Haendl	Vertrieb	7.500,00 €
Henry	Hansen		5.000,00 €
Bernhard	Herlitz		6.900,00 €
Josef	Huber		8.000,00 €
Johanna	Kisch		6.700,00 €
Sabine	Oswald		4.500,00 €
Barbara	Schmitt		4.800,00 €
		Gehalt für Abteilung Vertrieb	43.400,00 € 7
		Gehalt für alle Abteilungen	78.900,00 €

• Kreuztabelle

Dieser Modus stellt Ihren Bericht als zweidimensionale Tabelle dar. Er eignet sich zum Anzeigen von Daten aus einer Datenquelle mit Kategorien, die aus zwei anderen Datenquellen stammen.

Beispiel: In einer Kreuztabelle können Sie anzeigen, wieviel von jeder Produktart pro Quartal verkauft wurden. Hierzu ein Beispiel:

	Remote control	Power module	AV Preamplifier	Zwischensumme
Q3	71,00	64,00	65,00	200,00 6,45
Q1	46,00	51,00	21,00	118,00 5,61
Q2	102,00	40,00	17,00	159,00 6,36
Q4	82,00	29,00	17,00	128,00 5,56
Gesamtsumme	301,00	184,00	120,00	605,00
	6,02	6,57	5,45	6,05
	1,00	1,00	1,00	1,00

Beim Erstellen eines eigenen Schnellberichts können Sie mit dem Assistenten arbeiten, der Sie Schritt für Schritt leitet. Um den Assistenten zu öffnen, klicken Sie im Schnellberichteditor rechts oben auf die Schaltfläche:



Folgen Sie den Anweisungen auf dem Bildschirm und klicken Sie von einem Schritt zum nächsten. Im rechten Bereich erscheinen Erläuterungen zu den einzelnen Schritten.

Haben Sie den Bericht fertiggestellt, können Sie über den Assistenten auch den entsprechenden 4D Code generieren. Weitere Informationen dazu finden Sie im Abschnitt **4D Code generieren**.

Wollen Sie den Berichtsassistenten wieder verlassen, klicken Sie rechts oben auf die Schaltfläche **Manueller Modus**:



Schnellbericht erstellen

Mit dem Schnellberichteditor können Sie Berichte in der Design- und Anwendungsumgebung erstellen. Sie können ein Schnellberichtsbereich auch über einen Plug-In Bereich vom Typ Bericht in ein Formular integrieren (siehe **Plug-In Bereich**).

Um einen Schnellbericht anzulegen:

1. Wählen Sie im Menü **Bericht** den Befehl **Schnellbericht** oder klicken Sie in der Werkzeugleiste auf das Icon **Tools>Schnellbericht**.
4D zeigt den Schnellberichteditor an. Erscheint ein bereits angelegter Bericht, wählen Sie im Menü **Datei/Ablage** den Befehl **Neu** oder klicken Sie in der oberen Werkzeugleiste auf das Icon **Neu**, um einen neuen Bericht zu erstellen.
2. Klicken Sie im unteren linken Bereich des Schnellberichteditors auf die gewünschte Berichtart (**Liste** oder **Kreuztabelle**).
Standardmäßig ist der Typ Liste aktiv. Der Bereich für den Schnellbericht wird an den gewählten Typ angepasst.
3. Starten Sie mit dem Anlegen Ihres Berichts.

Schnellberichtvorlage laden und sichern

Sie können eine Berichtvorlage als Datei sichern und dann erneut über den Schnellberichteditor öffnen. Die Vorlage enthält alle Einstellungen für den Bericht ohne die Daten. Sie können verschiedene Vorlagen sichern und dann je nach Bedarf aufrufen.

Ist der Berichteditor geöffnet, können Sie eine gespeicherte Vorlage laden und zum Drucken eines neuen Berichts verwenden. Sie können dieselbe Vorlage auch mehrmals verwenden, um unterschiedliche Auswahlen von Datensätzen zu drucken.

Hinweis: Der zuletzt angelegte Bericht erscheint beim nächsten Öffnen des Schnellberichteditors, jedoch nur während derselben Arbeitssitzung.

Um eine Berichtvorlage zu sichern:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Speichern**
ODER
Klicken Sie in der Werkzeugleiste auf die Schaltfläche **Sichern als**.
4D zeigt einen Sichern-Dialog, wo Sie der Vorlage einen Namen geben können. Dateien für Schnellberichte haben die Endung ".4QR".
2. Geben Sie einen Namen für die Vorlage ein und klicken auf die Schaltfläche **OK**.
4D sichert den Bericht als Datei, die Sie mit dem Schnellberichteditor öffnen können. Sie können die Datei auf Ihrer Festplatte sichern. Verändern Sie einen Bericht und sichern ihn erneut, wird der alte Bericht überschrieben. Der Standard Sichern-Dialog erscheint nicht.

Um eine Berichtvorlage zu laden:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Öffnen**
ODER
Klicken Sie in der Werkzeugleiste auf die Schaltfläche **Öffnen**.
4D zeigt den Öffnen-Dialog mit der Liste der verfügbaren Berichtvorlagen.
2. Doppelklicken Sie auf einen Dateinamen oder wählen Sie einen Dateinamen in der Liste und klicken auf die Schaltfläche **OK**.
4D ersetzt die aktuelle Vorlage durch die neu geöffnete Vorlage.

Beschreibung des Editors

Der Schnellberichteditor enthält folgende Elemente:

- **Haupttabelle:** Hier legen Sie die Haupttabelle fest, die als Grundlage für den Bericht dient. In der Liste Felder erscheinen dann die dazugehörigen Felder sowie die verknüpften Felder zu dieser Tabelle.
- **Liste Felder:** Hier wählen Sie die Felder für den Bericht durch Doppelklick oder per Drag-and-Drop aus. Sie zeigt entweder die Felder der Haupttabelle, die hierarchische Liste der Tabellen und die verknüpften Felder oder die Liste aller Tabellen und Felder der Datenbank. Die Art der Anzeige wählen Sie in der darüberliegenden DropDown-Liste aus.

Indizierte Felder erscheinen in Fettschrift. Sie können auch die Felder von verknüpften Tabellen anzeigen und auswählen.

Hinweise:

- Im Schnellberichteditor erscheinen nur Tabellen und Felder, die nicht ausgeblendet sind. Weitere Informationen dazu finden Sie in den Abschnitten **Tabelleneigenschaften** und **Feldeigenschaften**.
- Der Schnellberichteditor unterstützt keine Datenfelder vom Typ Objekt
- **Alle Verknüpfungen:** Mit dieser Option definieren Sie, wie der Schnellberichteditor die Verknüpfungen zwischen den verschiedenen Tabellen der Datenbank verwendet. Prinzipiell kann der Editor nur automatische Verknüpfungen verwenden. Die Option ist standardmäßig inaktiv und der Editor kann nur automatische Verknüpfungen einsetzen.

Ist die Option markiert, betrachtet der Schnellberichteditor auch die manuellen Verknüpfungen als automatische Verknüpfungen und ermöglicht so den Zugriff auf alle verknüpften Daten in der Datenbank.

Hinweise für Entwickler:

- Die Option **Alle Verknüpfungen** setzt alle Verknüpfungen des aktuellen Prozesses auf automatisch. Das bleibt auch nach Schließen des Schnellberichteditors erhalten. Wollen Sie sicherstellen, dass alle Verknüpfungen nach Benutzen des Schnellberichteditors manuell sind, verwenden Sie die Anweisung:

SET AUTOMATIC RELATIONS(False;False)

- Wird der Editor über den 4D Befehl **QR REPORT** aufgerufen, erscheint diese Option nicht, damit der Entwickler den Status der Verknüpfungen direkt steuern kann.
- **Spaltentrenner:** Diese Linien grenzen die Spalten im Bericht voneinander ab. Sie können diese manuell bewegen, um die Spalten kleiner oder größer zu ziehen. Die manuelle Anpassung hebt die Option **Automatische Breite** auf, falls sie für die entsprechende Spalte gewählt wurde.
- **Zellen:** Das ist die Schnittstelle zwischen einer Zeile und einer Spalte.
- **Rollbalken:** Sie können in Bereichen mit Rollbalken scrollen, um Bereiche anzuzeigen, die außerhalb des Bereichs für den Schnellbericht liegen.
- **Sortierliste/Anzeige für Sortierreihenfolge:**
 - Im Modus **Liste** erscheint die dem Bericht zugewiesene Sortierreihenfolge sowie die Richtung aufsteigend oder absteigend. Jedes eingefügte Feld erzeugt im Bereich des Schnellberichts eine Zeile für Zwischensumme.
 - Im Modus **Kreuztabelle** erscheint die Sortierrichtung für jede der Datenquellen.
- **Oberer Bereich:** Hier legen Sie Ihren Bericht an. Sie können Datenfelder per Drag-and-Drop, per Doppelklick oder über das Kontextmenü einfügen; die Breite der hinzugefügten Spalten anpassen, Umbrüche oder Formeln löschen, Farben und Zellenränder definieren, etc.

Modus Liste:

		Datenquelle für Spalte		
Titelzeile	Kopfzeile	[Kunden]Firma	[Artikel]Artikel Name	[Positionen]MengePreis
Zeile Datenteil	Data	Firma	Artikel Name	MengenPreis
Zeilen Zwischensumme	[Artikel]Artikel Name gesamt			
Zeile Gesamtsumme	Gesamt			

- **Titelzeile:** Diese Zeile zeigt standardmäßig die Namen der Felder bzw. Formeln, die in den Bericht eingefügt wurden. Sie wird auf jeder Seite wiederholt. Sie können den Standardeintrag verändern.
- **Zeile Datenteil:** Das ist der Datenteil des Berichts. Die Zeile wird für jeden Datensatz wiederholt. Sie können je nach Art der Daten auch bestimmte Anzeigeformate zuweisen.
- **Zeilen Zwischensumme:** Diese Zeilen werden für jede Sortierebene angelegt. Sie enthalten Zwischensummen sowie zugeordneten Text.
- **Datenquellen für Spalten:** Das sind die Bezeichnungen der aktuellen Referenzen auf die Datenquelle.

Modus Kreuztabelle:

		Mittlere Zelle	Titel letzte Spalte
Datenquellen	[Kunden]Firma	[Artikel]Artikel Name	Summe
Titel letzte Spalte	Gesamt Summe	Summe	Summe
Summenzellen			

- **Datenquellen:** Diese beiden Zellen enthalten Felder für die zwei Kategorien des Array. Die mittlere Zelle kann sowohl Daten aus Feldern als auch Berechnungen aufnehmen.
- **Summenzellen:** Diese Zellen dienen zur Aufnahme von Berechnungen in einer Spalte.
- **Titelzellen:** Diese Zellen enthalten die Bezeichnungen der letzten Spalte oder Zeile.
- **Kontextmenüs:** Der Schnellberichteditor enthält Kontextmenüs, über die Sie rasch auf bestimmte Zeilen, Spalten sowie Operationen in Zellen zugreifen können. Sie können bestimmte Operationen direkt über ein

Kontextmenü des Berichteditors ausführen und müssen nicht erst ein Menü auswählen oder mit den Eigenschaften für Zellen bzw. Spalten arbeiten.

Nachfolgende Darstellung zeigt ein fertiges Formular für einen Schnellbericht und das Ergebnis des gedruckten Berichts:

Bericht als Liste:

Firma	Artikel Name	MengenPreis
WILLCOX Distribution	Hafer	1560
WILLCOX Distribution	Hopfen	20
WILLCOX Distribution	Hopfen	1680
Gesamtsumme		3260

Bericht als Kreuztabelle:

Rechnungsquartal	Artikel	Zwischensumme
Q1	Remote control	20,00
Q1	Power module	11,00
Q1	AV Preamp/iter	11,00
Q2	Remote control	17,00
Q2	Power module	23,00
Q2	AV Preamp/iter	17,00
Gesamtsumme	Remote control	37,00
Gesamtsumme	Power module	34,00
Gesamtsumme	AV Preamp/iter	28,00
Gesamtsumme	Zwischensumme	99,00

Text hinzufügen oder ändern

Sie können in der Berichtvorlage Bezeichnungen hinzufügen oder ändern. Haben Sie z.B. eine Berechnung definiert, können Sie in anderen Zellen mit der Zeile Zwischensumme bzw. Gesamtsumme erklärenden Text hinzufügen. Es gibt folgende Möglichkeiten:

- Sie bearbeiten den Text, welchen 4D automatisch in der Titelzeile des Berichts vorgibt
- Sie fügen Text in leere Zellen der Zeile für Umbruch oder Summe ein
- Sie fügen den Wert eines Feldes Zwischensumme in die Zeilen für die Zwischensumme ein
- Sie definieren Schriftart, -größe, -stil oder Ausrichtung für beliebigen Text im Bericht

Um Text in einer Zelle hinzuzufügen, doppelklicken Sie in der Vorlage des Schnellberichts auf eine leere Zelle. In der Zelle erscheint ein Einfügekpunkt. Geben Sie eine Bezeichnung für die Berechnung ein, wählen Sie eine Zelle in derselben Zeile wie die Symbole für Berechnung. In einer Zelle mit Berechnung können Sie keinen Text eingeben.

Um Text in einer Zelle zu ändern, Doppelklicken Sie in die Zelle, damit dort ein Einfügekpunkt erscheint. Bewegen Sie diesen durch den Text, den Sie ändern wollen. 4D hebt diesen Teil hervor.

Spalten hinzufügen

Im Modus Listerstellen Spalten, indem Sie Feldnamen aus der Feldliste im unteren Bereich per Drag-and-Drop in den Formularbereich bewegen. Um ein Feld aus einer verknüpften Tabelle hinzuzufügen, klicken Sie vor dem Feldnamen auf das Erweiterungskästchen, unter Windows ein Pluszeichen, auf Mac OS ein Pfeil. Dazu müssen Sie zuvor im darüberliegenden DropDown-Menü **Verknüpfte Tabellen** oder **Alle Tabellen** ausgewählt haben. Sie können Felder aus verknüpften Tabellen hinzufügen, wenn die Verknüpfung automatisch ist. Wollen Sie auch manuelle Verknüpfungen verwenden, markieren Sie die Option Alle Verknüpfungen. Dann setzt der Editor alle Verknüpfungen auf automatisch. Sie können auch eine leere Spalte in einen Schnellbericht setzen und dieser dann ein Feld oder eine Formel zuweisen.

Hinweis: Im Modus Kreuztabellen können Sie eine Spalte weder bearbeiten noch hinzufügen, da der Bericht bereits alle benötigten Spalten enthält.

Um ein Feld hinzuzufügen:

1. Setzen Sie im Bereich den Feldnamen rechts neben die letzte vorhandene Spalte und lassen Sie die Maustaste los ODER Doppelklicken Sie in der Liste der Felder auf den gewünschten Feldnamen.

Hinweis: Setzen Sie ein Feld auf eine bereits vorhandene Spalte, wird sie ersetzt.

4D legt eine Spalte an und setzt den Feldnamen in die Kopfzeile und die erste Zelle der Spalte.

4D druckt die Feldnamen standardmäßig auf jeder Seite des Berichts als Spaltentitel.

Um ein Feld über Menübefehl hinzuzufügen:

1. Wählen Sie im Berichteditor im Menü **Spalten** den Befehl **Einfügen**.
ODER
Klicken Sie bei gedrückter rechter Maustaste in einen unbenutzten Bereich des Schnellberichts. Es erscheint die Schaltfläche **Neu**.
2. Klicken Sie auf die Schaltfläche, öffnet sich der Formulareditor. Wählen Sie eine Formel oder ein Feld und klicken Sie auf die Schaltfläche **OK**, wird die Spalte angelegt.

Spalten löschen

Haben Sie Spalten für Ihren Schnellbericht im Modus Liste angelegt, können Sie diese auch an eine andere Stelle legen oder wieder entfernen.

Um eine ausgewählte Spalte zu entfernen, Wählen Sie im Menü **Spalten>Löschen** oder klicken Sie mit gedrückter rechter Maustaste in den Kopfteil der gewünschten Spalte und wählen dann im Kontextmenü den Eintrag **Löschen**. 4D entfernt die gewählte Spalte aus der Berichtvorlage.

Spalten ersetzen

Sie können eine Spalte im Schnellbericht ersetzen, wenn Sie dort ein anderes Feld per Drag-and-Drop ablegen. Auf dieselbe Weise können Sie ein Feld mit einer Formel ersetzen und umgekehrt.

Um eine Spalte durch ein Feld zu ersetzen:

1. Ziehen Sie ein Feld aus der Feldliste im unteren Bereich in die vorhandene Spalte.
Die Spalte wird unmittelbar ersetzt. War das bisherige Feld als Sortierebene definiert, wird es auch in der Sortierliste ersetzt.

Um eine Spalte über den **Formeleditor** zu ersetzen:

1. Wählen Sie die betreffende Spalte und dann im Menü **Spalten>Bearbeiten**.
ODER
Klicken Sie mit gedrückter rechter Maustaste auf den Spaltentitel und dann im Kontextmenü auf den Eintrag **Bearbeiten**.
Auf dem Bildschirm erscheint der Formeleditor von 4D. Geben Sie einen Feldnamen oder eine Formel für die Spalte ein.
4D druckt das Formelergebnis für jeden Datensatz, der im Datenteil erscheint. Weitere Informationen dazu finden Sie im Abschnitt **Formeln hinzufügen**.

Spalten anpassen

Der Schnellberichteditor passt die Spaltenbreite automatisch an, d.h. die Option **Automatische Breite** ist standardmäßig aktiviert. Er richtet jede Spalte nach der max. Länge des Spaltentitels und der Daten innerhalb der Spalte aus. Die Anpassung erfolgt beim Drucken des Berichts. Die Berichte haben je nach Auswahl der Datensätze unterschiedlich breite Spalten.

Sie können die Option **Automatische Breite** im Menü **Spalten**, im Kontextmenü der Spalte oder über die Schaltfläche in der Werkzeugpalette festlegen bzw. ansehen.

Zum Betrachten der Breite einzelner Spalten schalten Sie um auf Bildschirmansicht. Weitere Informationen dazu finden Sie im Abschnitt **Schnellbericht ausführen**.

Haben Sie die Option **Automatische Breite** gewählt, richtet sich die Spaltenbreite nach der maximalen Breite der Daten in den ausgewählten Datensätzen. Wählen Sie andere Datensätze, kann das eine andere Spaltenbreite ergeben.

Sie können eine Spalte manuell anpassen, dann wird die Option **Automatische Breite** deaktiviert. Der Text in der Spalte wird dann in den angegebenen Bereich eingepasst.

Spalten verschieben

Beim Anlegen eines Berichts im Listenmodus können Sie Spalten auch an eine andere Stelle bewegen. Dazu wählen Sie die entsprechende Spalte aus und klicken in der unteren Werkzeugleiste auf das Icon mit Pfeil nach links oder nach rechts  oder wählen im Menü **Spalten>Nach links verschieben/ Nach rechts verschieben**.

4D bewegt die Spalte in die gewählte Richtung.

Grafische Attribute eines Berichts

Sie können das Erscheinungsbild eines Schnellberichts verändern. Mit dem Schnellberichteditor können Sie folgende Attribute definieren:

- Schriftart, -größe, -stil, Ausrichtung und Farbe
- Hintergrundfarbe der Zellen
- Zellenrahmen

4D bietet auch verschiedene Berichtsvorlagen, die Sie an Ihre Bedürfnisse anpassen können.

Hinweis: Die Farbangaben für die Darstellung des Berichts werden nur für die Ausgabeziele „Drucker“ und „HTML Datei“ berücksichtigt. Für andere Ausgabearten muss die Formatierung nach dem Erstellen des Berichts durchgeführt werden, und zwar mit dem jeweiligen Programm (4D View oder ein Texteditor). Weitere Informationen dazu finden Sie im Abschnitt [Schnellbericht ausführen](#).

Schrifteigenschaften zuweisen

Sie können beim Anlegen des Schnellberichts verschiedene Schriftarten, -größen, -stile sowie die Ausrichtung und Farben festlegen. Das ist möglich für Text, Daten und Summenberechnungen in Zeilen, Spalten oder Zellen.

Weisen Sie dem Datenteil des Berichts eine Eigenschaft zu, sehen Sie das Ergebnis erst in der Vorschau oder beim Drucken.

Sie können Schrifteigenschaften entweder über die Menübefehle oder Kontextmenüs des Schnellberichteditors definieren.

Um eine Schrifteigenschaft zu definieren:

1. Klicken Sie bei gedrückter rechter Maustaste auf die Bezeichnung der entsprechenden Zeile, Kopfzeile oder Zelle.
2. Wählen Sie im Kontextmenü in den Untermenüs die gewünschte Schriftart, -größe, -stil oder Ausrichtung:



ODER

1. Wählen Sie im Bericht die Spalte, Zeile oder Zelle, für welche die Eigenschaft gelten soll.
2. Wählen Sie in der oberen Werkzeugleiste in den entsprechenden DropDown-Menüs Schriftart und -größe aus und legen bei Bedarf Stil, Ausrichtung und Farbe fest:



Hinweis: Schriftstil und Ausrichtung können Sie auch im Menü **Stil** festlegen.

4D weist die Eigenschaften dem Text, den Daten oder Berechnungen zu, die im gewählten Bereich erscheinen.

Hintergrundfarbe für Zellen

Sie können für jede Zelle eine Hintergrundfarbe setzen. Sie können eine einzelne Farbe oder zwei wechselnde Farben setzen. Das macht die Tabellen leichter lesbar.

Das ist sowohl im Modus Liste als Kreuztabelle möglich. Hintergrundfarben werden mit den Ausgabezielen „Drucken“ und „HTML Datei“ verwendet.

Um eine Hintergrundfarbe zu setzen:

1. Wählen Sie eine Zelle, Spalte oder Zeile und dann die Hauptfarbe in der Farbpalette „Hintergrundfarbe“ der Werkzeugpalette:



Sie können für Zeilen, Zellen und Spalten eine beliebige Farbkombination wählen.

2. (Optional) Wollen Sie eine wechselnde Hintergrundfarbe verwenden, wählen Sie diese in der Farbpalette „Alt-Hintergrundfarbe“ aus:



Zellen, denen zwei wechselnde Hintergrundfarben zugewiesen wurden, zeigen beide Farben im Schnellberichteditor an.

Im folgenden sehen Sie eine Vorschau für einen Bericht im Listenmodus mit wechselnden Hintergrundfarben:

Land	Artikel	Anzahl
Germany	AV Preampfier	1
UK	AV Preampfier	1
France	Power module	1
France	Power module	1
Italy	Remote control	1
UK	Remote control	1
Germany	AV Preampfier	2
Italy	AV Preampfier	2
UK	AV Preampfier	2
UK	AV Preampfier	2
UK	Power module	2
Germany	Remote control	2
Germany	Remote control	2
Italy	Remote control	2
Italy	Remote control	2
Italy	Remote control	2
Italy	Remote control	2
Italy	Remote control	2
Italy	Remote control	2
UK	Remote control	2
USA	Remote control	2
USA	Remote control	2

Hintergrundfarben werden auch in Berichten im HTML-Format berücksichtigt:

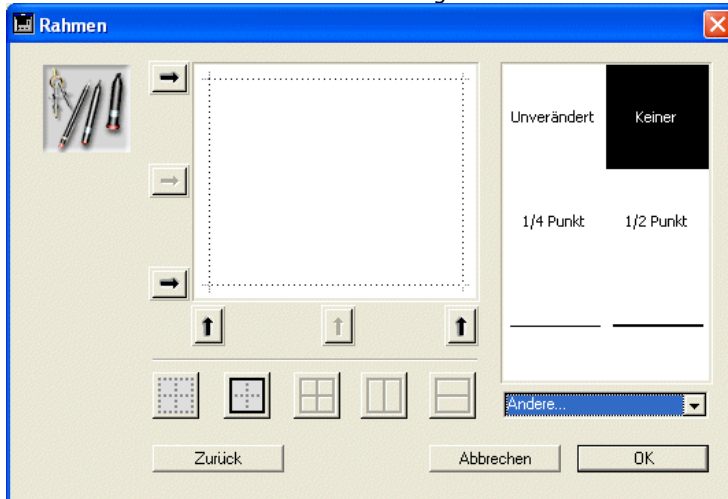
	[Rechnung]Land	[Rechnung]Artikel	[Rechnung]Anzahl
Kopfteil	Land	Artikel	Anzahl
Daten			
[Rechnung]Land geändert			

Land	Artikel	Anzahl
UK	AV Preampfier	1
France	Power module	1
UK	Remote control	1
Italy	AV Preampfier	2
UK	AV Preampfier	2
Germany	Remote control	2
Italy	Remote control	2
Italy	Remote control	2
Italy	Remote control	2

Sie können Ränder für Zellen sowohl im Modus Liste als auch im Modus Kreuztabelle setzen.
Um Rahmen für eine Zelle, Spalte oder Zeile zu setzen:

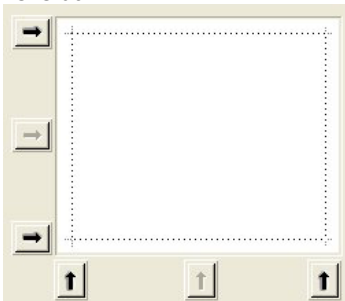
1. Markieren Sie eine Zelle, Spalte oder Zeile und wählen Sie im Menü **Spalten** oder im Kontextmenü den Eintrag **Rahmen**.

Auf dem Bildschirm erscheint das Dialogfenster für Rahmen:



Die Darstellung des Dialogfensters für Rahmen variiert je nach Art der Auswahl in der Berichtvorlage. So verhält sich eine Mittelzelle in einem Kreuztabellen-Bericht anders als eine Datenzelle im Listenbericht: Die Mittelzelle wird horizontal und vertikal wiederholt, die Datenzelle dagegen nur vertikal. Andere Zellen, z.B. Titelzellen werden gar nicht wiederholt.

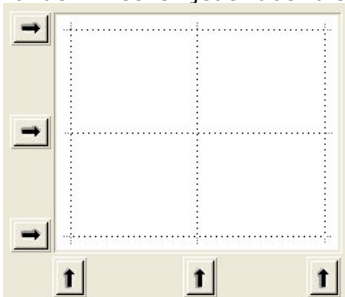
- Haben Sie eine Titelzelle gewählt, d.h. sie wird nicht wiederholt, stellen die äußeren Linien die Außenränder der Zelle dar:



- Haben Sie eine Datenzelle in einem Listenbericht gewählt, d.h. sie wird vertikal wiederholt, stellen die äußeren Linien die Außenränder der wiederholten Zellen dar, die innere horizontale Linie den Rand zwischen zwei aufeinanderfolgenden Zellen:



- Haben Sie eine Mittelzelle in einem Kreuztabellen-Bericht gewählt, d.h. sie wird horizontal und vertikal wiederholt, stellen die äußeren Linien die Außenränder der wiederholten Zellen dar, die inneren Linien die Ränder zwischen jeder über die Mittelzelle angelegten Zelle:



2. Wählen Sie Linienstärke und die gewünschte Farbe. Sie können für jeden Rand eine andere Farbe wählen.
3. Wählen Sie die Ränder über die Pfeile oder die darunterliegenden Symbole. Je nach Art der gewählten Zelle sind bestimmte Optionen inaktiv.
4. Wiederholen Sie die Schritte 2 und 3 für jeden Rand, den Sie einstellen wollen und klicken auf die Schaltfläche **OK**, um die Änderungen zu bestätigen.
Über die Schaltfläche **Zurück** können Sie zu den ursprünglichen Einstellungen zurückkehren.

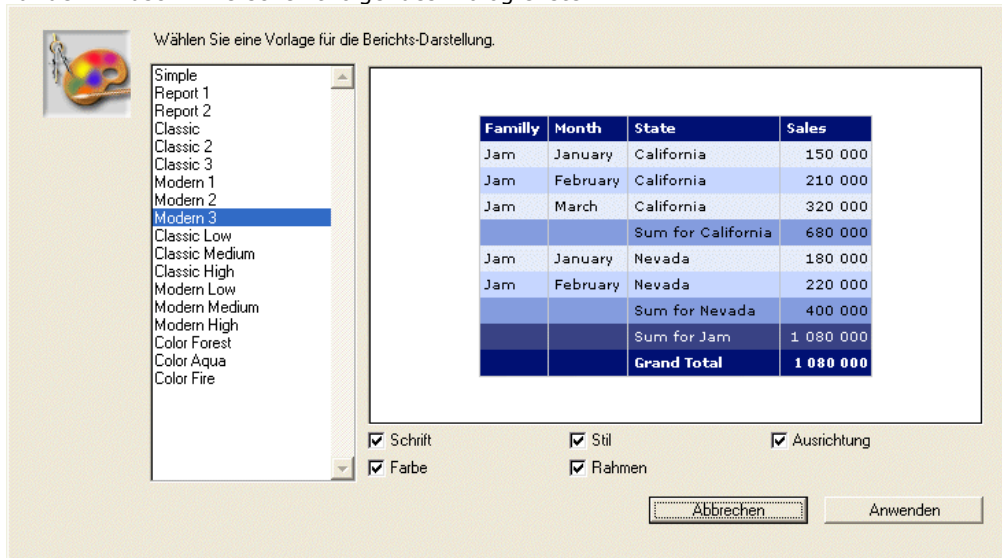
Der Schnellberichteditor bietet eine Reihe grafisch gestalteter Vorlagen für gedruckte oder HTML Berichte an. Sie können diese Vorlagen jederzeit verwenden und beliebig an Ihre eigenen Bedürfnisse anpassen.

Eine Berichtsvorlage kombiniert eine Reihe grafischer Elemente miteinander: Schriftart und -stil, Farben, Ränder und Ausrichtung. Sie können wählen, ob Sie alle Elemente oder nur einen Teil davon verwenden wollen.

Um eine Vorlage zu verwenden:

1. Wählen Sie im Menü **Stil>Darstellung**.

Auf dem Bildschirm erscheint folgendes Dialogfenster:



2. Wählen Sie eine Vorlage in der Liste aus. Im rechten Teil sehen Sie die jeweilige Vorschau. Wählen Sie eine Vorlage, die am ehesten Ihren Bedürfnissen entspricht. Sie können diese dann jederzeit noch weiter anpassen.

3. Wollen Sie nicht alle grafischen Elemente der Vorlage verwenden, können Sie einzelne Optionen unterhalb der Vorschau deaktivieren. Standardmäßig sind alle Optionen markiert. Deaktivieren Sie z.B. die Option "Ränder", werden die in der Vorlage definierten Ränder nicht für Ihren Bericht berücksichtigt.

Hinweis: Berichte im HTML Format berücksichtigen nur die Option "Farbe".

4. Klicken Sie auf die Schaltfläche **Anwenden**.

Die gewählten Merkmale werden sofort für Ihren Bericht angewendet. Sie ersetzen andere zuvor gesetzte grafische Attribute.

Beachten Sie, dass bestimmte Einstellungen erst nach Erstellen des Berichts zu sehen sind.

Datensätze sortieren

Mit dem Schnellberichteditor können Sie die Datensätze in Ihrem Bericht auch sortieren. Dafür gibt es zwei Gründe:

- Sie wollen Datensätze in einer bestimmten Reihenfolge betrachten
- Sie wollen Datensätze in Gruppen mit Zwischensummen erstellen, um bestimmte Berechnungen durchzuführen

Sortierreihenfolge im Modus Liste erstellen

Sie können jederzeit eine Sortierreihenfolge definieren; ziehen Sie einfach eine Spalte in den Bereich Sortierfolge.
Beispiel: Sie wollen die Datensätze der Vertriebsmitarbeiter nach Verkaufszonen sortieren. Hierzu ziehen Sie die Spalte Verkaufsgebiet in den Bereich Sortierfolge.


Sie können auch nach einer Formel sortieren. Dazu ziehen Sie die Spalte mit der Formel in den Bereich Sortierfolge.
Weitere Informationen dazu finden Sie im Abschnitt **Formeln hinzufügen**.

Sie können einen Bericht in mehreren Ebenen sortieren. Die Reihenfolge, in der die Datenfelder und Formeln in der Liste der Sortierfolge erscheinen, bestimmt die Sortierebene.

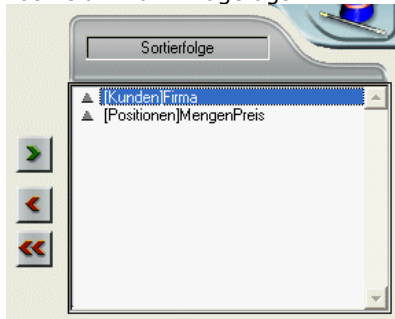
Um eine Sortierreihenfolge über die Liste der Felder festzulegen:

1. Ziehen Sie das entsprechende Feld aus der Liste in den danebenliegenden Bereich Sortierfolge.

ODER

Markieren Sie das Feld in der Liste und klicken auf den grünen Pfeil .

Das Feld wird hinzugefügt:

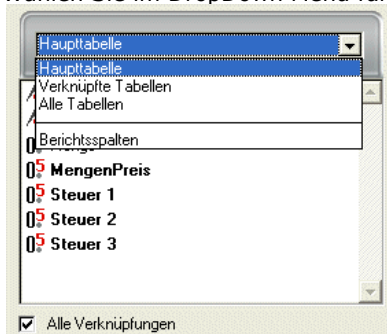


Standardmäßig wird in aufsteigender Reihenfolge sortiert, also von A nach Z. Über das Dreieck vor dem Namen können Sie in auf- oder absteigender Reihenfolge sortieren.

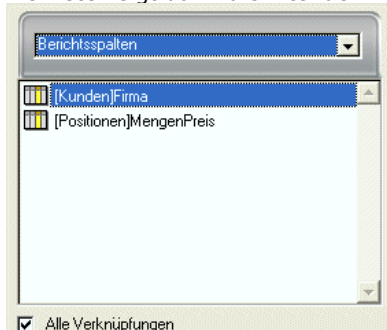
Hinweis: Ist das sortierte Feld noch nicht in eine Spalte im Bericht eingefügt, wird automatisch eine weitere Spalte angelegt.


Um die Sortierreihenfolge für eine Spalte (Feld oder Formel) festzulegen:

1. Wählen Sie im DropDown Menü für Tabellenwahl den Eintrag Berichtsspalten.



Die Liste zeigt dann die Titel der im Bericht vorhandenen Spalten:



- Ziehen Sie die entsprechende Spalte aus der Liste in den danebenliegenden Bereich Sortierfolge ODER Markieren Sie das Feld in der Liste und klicken auf den grünen Pfeil . Der Name der Spalte erscheint dann im Bereich Sortierfolge.

Sortierebene ändern

Um die Sortierebene zu ändern:

- Klicken Sie bei gedrückter rechter Maustaste auf das Feld in der Sortierliste, um das Kontextmenü anzuzeigen.
- Wählen Sie den Eintrag **Nach oben** oder **Nach unten**, um die Felder entsprechend zu bewegen.

Feld oder Formel in Sortierliste löschen

Sie können jederzeit Felder oder Formeln aus der Sortierliste entfernen.

Um ein Feld oder eine Formel zu entfernen:

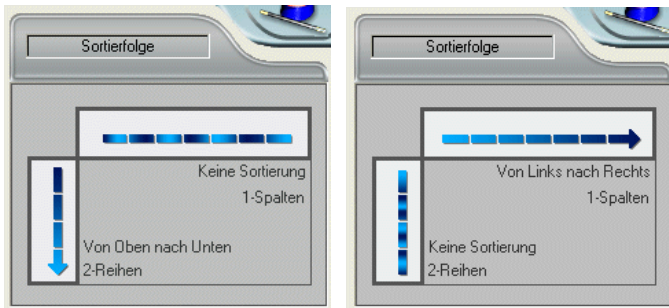
- Markieren Sie das betreffende Feld bzw. die Formel bei gedrückter rechter Maustaste, um das Kontextmenü zu öffnen.
- Wählen Sie den Eintrag **Löschen**.

Beachten Sie, dass die dazugehörige Spalte nicht aus dem Bericht entfernt wird, dagegen verschwindet die dazugehörige Zeile Zwischensumme.

Werte in Kreuztabellen sortieren

In einem Kreuztabellen-Bericht können Sie nur Werte aus den horizontalen und vertikalen Datenquellen sortieren. Sie werden in der Endtabelle als Kategorien verwendet.

Um in einem Kreuztabellen-Bericht nach Kategorien zu sortieren, klicken Sie im Bereich Sortierfolge auf die Sortierbalken. Die Pfeile geben die aktuell gewählte Sortierrichtung an:



Um die Sortierreihenfolge einer Datenquelle zu ändern bzw. zu löschen, klicken Sie erneut darauf. Die verschiedenen möglichen Berichte erscheinen der Reihe nach. Erscheinen keine Pfeile, wurde keine Sortierung ausgewählt. Die Werte erscheinen dann so wie in der Auswahl angegeben.

Ebenen für Zwischensummen

In einem Schnellbericht setzen Sie Umbrüchebenen, um Datensätze nach Werten in einem oder mehreren Sortierfeldern zu trennen. Für jede Umbrüchebene wird ein Umbruchbereich gedruckt. Sie können im Umbruchbereich Berechnungen drucken. Die Berechnungen – Summe, Durchschnitt, Min, Max und Anzahl – werden für jede Datensatzgruppe berechnet.

Umbrüchebenen werden durch Sortier- und Umbrüchebenen bestimmt. Sortieren Sie z.B. Datensätze nach Verkaufszonen und legen eine Umbruchzeile an, fügt 4D nach den Datensätzen für eine Zone einen Umbruch ein. Diese Zeilen werden automatisch eingefügt, wenn eine Sortierung definiert wurde.

Sie können in jeder Umbruchzeile eine Berechnung durchführen, z.B. Zwischensummen für den Produkteverkauf jedes Landes innerhalb einer Verkaufszone. Weitere Informationen dazu finden Sie im Abschnitt **Summenberechnungen hinzufügen**.

	[Kunden]Firma	[Artikel]Artikel Name	[Positionen]MengenPreis
Kopfteil	Firma	Artikel Name	MengenPreis
Daten			
[Artikel]Artikel Name geändert			
[Kunden]Firma geändert			
Summe			

Die Bezeichnung einer Zeile mit Zwischensumme gibt an, welche Änderung im Wert den Umbruch hervorruft.

Werte aus Umbruchfeldern für Bezeichnungen

Sie können die Darstellung und Verständlichkeit Ihrer Berichte verbessern, wenn Sie die Umbruchzeile mit dem Wert des Umbruchfeldes benennen.

Um den Wert eines Umbruchfeldes als Bezeichnung zu drucken, verwenden Sie das Nummernzeichen (#). Beispiel: Der Text "Gesamtlohn der Abteilung #" setzt den Namen der Abteilung, also den Wert des Feldes Abteilung an die Stelle des Platzhalters, wenn der Bericht gedruckt wird.

Der Platzhalter muss nicht in derselben Spalte wie das Umbruchfeld liegen. Er zeigt den Wert dieses Feldes in jeder beliebigen Zelle der Umbruchzeile an

Im folgenden sehen Sie ein Beispiel:

[Kunden]Firma	[Artikel]Artikel Name	[Positionen]MengenPreis
Kopfzeile	Firma	Artikel Name
Daten		
Summe	Gesamt #	Summe

Abstand Zwischensumme

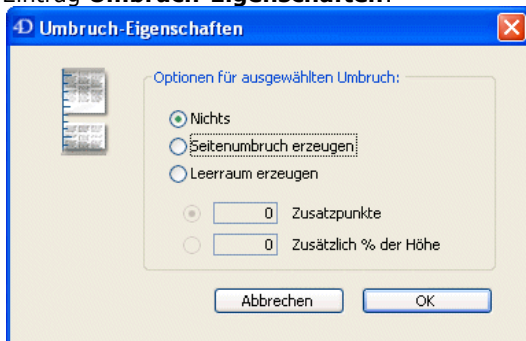
Sie können Zeilen mit Zwischensummen visuell gestalten, z.B. durch Seitenumbrüche oder Abstände. Sie können z.B. nach jeder Zwischensumme einen Seitenumbruch einfügen und so die verschiedenen Teile des Berichts voneinander trennen.

Um Abstand für eine Zeilen mit Zwischensumme zu setzen:

1. Wählen Sie die Zeile mit der Zwischensumme und dann im Menü **Datei/Ablage** des Editors den Befehl **Umbruch-Eigenschaften**.

ODER

Klicken Sie mit der rechten Maustaste auf die Titelzeile der Zwischensumme und wählen Sie im Kontextmenü den Eintrag **Umbruch-Eigenschaften**:



2. Wählen Sie eine der nachfolgenden Optionen aus:
 - **Nichts** (Standardeinstellung): Es wird keine spezifische Eigenschaft auf die Zeile mit Zwischensumme angewandt. Es gelten dieselben Einstellungen für Abstand wie für die anderen Zeilen im Bericht.
 - **Seitenumbruch erzeugen**: Nach jeder Zeile mit Zwischensumme wird ein Seitenumbruch erzeugt.
 - **Leerraum erzeugen**: Unter jeder Zeile mit Zwischensumme wird ein bestimmter Abstand eingefügt, den Sie über folgende Optionen definieren:
 - **Zusatzpunkte**: Damit richten Sie eine bestimmte Höhe für den zusätzlichen Abstand in Punkt ein.
 - **Zusätzlich % der Höhe**: Damit legen Sie den zusätzlichen Abstand als Prozentsatz der standardmäßigen Zeilenhöhe des Berichts fest. Wollen Sie z.B. einen Extraabstand von zwei Leerzeilen einfügen, übergeben Sie als Wert 200.

Berechnungen hinzufügen

Formeln hinzufügen

Sie können eine Formel im Schnellbericht hinzufügen. Sie können z.B. eine Formel hinzufügen, die aus dem Feld Jahreslohn den Monatslohn eines Angestellten berechnet

Um eine Formel hinzuzufügen:

1. Fügen Sie eine leere Spalte ein oder an
ODER
klicken Sie auf eine vorhandene Spalte und wählen im Menü **Spalten>Bearbeiten**.
ODER
Doppelklicken Sie auf einen vorhandenen Spaltentitel
ODER
Klicken Sie bei gedrückter rechter Maustaste auf einen Spaltentitel, um das Kontextmenü anzuzeigen und wählen den Eintrag **Bearbeiten**.
4D zeigt den **Formeleditor** an, um eine Formel anzulegen. Haben Sie eine vorhandene Spalte gewählt, ersetzt die Formel den bisherigen Inhalt der Spalte.
Hinweis: Stellen Sie sicher, dass die Formel nicht die aktuelle Auswahl verändert. Denn das verursacht Probleme bei Drucken des Berichts, da er auf der aktuellen Auswahl basiert.
2. Erstellen Sie selbst eine Formel
ODER
Klicken Sie auf die Schaltfläche **Öffnen**, um eine vorhandene Formel von der Festplatte zu laden.
Weitere Informationen zum Formeleditor finden Sie im Kapitel **Formeleditor**.
Um die Formel als Datei zu sichern, die Sie später für andere Spalten oder Berichte wiederverwenden können, klicken Sie auf die Schaltfläche **Sichern** und geben im Sichern-Dialog einen Namen ein.
3. Klicken Sie auf die Schaltfläche **OK**, wird die Formel der Spalte zugewiesen.
4D fügt in der Spalte eine neue Bezeichnung ein, die sie als Formel kennzeichnet. Sie können die Spalte umbenennen, also im Kopfteil der Spalte eine andere Bezeichnung eingeben. Die Formeln haben die Bezeichnung C1 bis Cn. Das sind die Variablenamen, die den aktuellen Wert der Spalte enthalten. Diese Variablen können Sie in anderen Formeln verwenden.

Summenberechnungen hinzufügen

Sie können in jeder Summen- bzw. Zwischensummenzeile eine Berechnung auf den Inhalt von Feldern und Formeln hinzufügen. In einem Kreuztabellen-Bericht lassen sich Berechnungen in der Mittelzeile einfügen.

Im Schnellberichteditor stehen in der Werkzeugleiste folgende Operatoren zur Verfügung:



Diese Icons entsprechen folgenden Berechnungsarten:

- **Sum** berechnet die Gesamtsumme der Werte im Bericht oder Umbruch.
- **Durchschnitt** berechnet den Durchschnitt der Werte im Bericht oder Umbruch.
- **Minimum** zeigt den niedrigsten Wert im Bericht oder Umbruch an.
- **Maximum** zeigt den höchsten Wert im Bericht oder Umbruch an.
- **Anzahl** berechnet die Anzahl der Datensätze im Bericht oder Umbruch.
- **Standardabweichung** zeigt die Quadratwurzel der Varianz des Berichts oder Umbruchs (Varianz ist ein abweichender Wert des Durchschnitts).










Diese Optionen erscheinen auch im Kontextmenü zu den Zellen der Zeilen Zwischensumme und Gesamtsumme bzw. in Kreuztabellen-Berichten in den Zellen Gesamtsumme und Mittelzellen.

• Listenbericht

Setzen Sie die Berechnung in der Zeile Gesamtsumme, wird die Berechnung für alle Datensätze im Bericht ausgeführt. Setzen Sie die Berechnung in der Zeile Zwischensumme, wird für die Datensätze zu jeder Umbruchzeile eine separate Berechnung ausgeführt.

• Kreuztabellenbericht

Berechnungen werden folgendermaßen ausgeführt:

	[Artikel]Artikel Name	Summe	
[Kunden]Firma	[Positionen]MengenPreis		Summe
	 Summe 1		Durchschnitt
Gesamt Summe	 Summe		Summe
	 Durchschnitt		Durchschnitt
	 3 Min	 4	Min

1 = Gilt für jede Zelle der Tabelle

2 = Gilt für Werte in jeder Zeile

3 = Gilt für Werte in jeder Spalte

4 = Gilt für Werte in der letzten Spalte und Zeile

Um eine Berechnung hinzuzufügen:

1. Wählen Sie die gewünschte Zelle.
Die Berechnungen Summe, Minimum, Maximum und Durchschnitt funktionieren nur in einem Feld vom Typ Zahl

oder einer Formel.

2. Klicken Sie in der Werkzeugleiste auf die entsprechenden Symbole

ODER

Wählen Sie im Kontextmenü die gewünschte Berechnung.

4D setzt das/die markierten Icons in die gewählte Zelle. Nachfolgender Bericht enthält ein Icon in der Zeile Gesamtsumme:

	[Kunden]Firma	[Artikel]Artikel Name	[Positionen]Mengen	Preis
Kopfzeile	Firma	Artikel Name	Mengen	Preis
Daten				
[Artikel]Artikel Name geändert				
[Kunden]Firma geändert				
Summe		Gesamtsumme		Summe

Platzhalter für Berechnung

Sie können Berechnungen mit folgenden Platzhaltern eingeben:

- ##S wird ersetzt durch die Summe in der Zeile Zwischensumme oder Gesamtsumme
- ##A wird ersetzt durch den **Durchschnitt**
- ##C wird ersetzt durch die **Anzahl**
- ##X wird ersetzt durch **Max**
- ##N wird ersetzt durch **Min**
- ##D wird ersetzt durch **Standardabweichung**
- ##xx, wobei xx eine Spaltennummer ist. Dieser Platzhalter wird ersetzt durch den Wert der Spalte mit der definierten Formatierung. Gibt es diese Spalte nicht, wird die Aktion nicht ausgeführt.

Diese Platzhalter sind hilfreich, wenn Sie in einer Zelle Bezeichnung und Werte mischen wollen.

Wiederholte Werte für Umbruchspalten

In einem Bericht mit Zwischensummen heißen die Spalten, welche Datensätze in Gruppen für Berechnungen zusammenfassen, Umbruchspalten. Im nachfolgend gezeigten Bericht ist das Feld Firma eine Umbruchspalte. Hier werden die Datensätze nach Firma gruppiert.

Beim Drucken dieses Berichts werden die Werte in der Umbruchspalte nur einmal pro Umbruch gedruckt, d.h. der Name der Firma wird pro Gruppe nur jeweils für den ersten Datensatz gedruckt und nicht für jeden Datensatz wiederholt.

Firma	Artikel Name	MengenPreis
OSTERMAN, Heinz	Weißkohl	600
	Weizen	1450
	Malz	6000
	Hafer	2000
	Weißkohl	1332
	Weizen	3219
Unterhoff Brauerei	Malz	13320
	Hafer	4440
	Weißkohl	12000
	Reis	20000
WILLCOX Distribution	Weißkohl	12000
	Malz	1440
WILLCOX Distribution	Roggen	654
	Hopfen	20

In manchen Fällen wollen Sie jedoch die Werte in Umbruchspalten wiederholen, so dass sie für jeden Datensatz angezeigt werden. Dazu wählen Sie die Spalteneigenschaft **Wiederholte Werte**.

Um wiederholte Werte anzuzeigen:

1. Klicken Sie in die Kopfzeile der gewünschten Spalte und dann in der Werkzeugleiste auf das Icon

ODER

Wählen Sie im Kontextmenü für diese Spalte den Eintrag **Wiederholte Werte**

ODER

Wählen Sie im Menü **Spalten>Wiederholte Werte**.

Im folgenden sehen Sie den o.a. Bericht mit Wiederholten Werten, d.h. der Name der Firma wird für jeden Datensatz angezeigt:

Firma	Artikel Name	MengenPreis
OSTERMAN, Heinz	Weißkohl	600
OSTERMAN, Heinz	Welzen	1450
OSTERMAN, Heinz	Malz	6000
OSTERMAN, Heinz	Hafer	2000
OSTERMAN, Heinz	Weißkohl	1332
OSTERMAN, Heinz	Welzen	3219
OSTERMAN, Heinz	Malz	13320
OSTERMAN, Heinz	Hafer	4440
OSTERMAN, Heinz	Weißkohl	12000
Unterhoff Brauerei	Reis	20000
Unterhoff Brauerei	Weißkohl	12000
Unterhoff Brauerei	Malz	1440
WILLCOX Distribution	Roggen	654
WILLCOX Distribution	Hopfen	20

Anzeigeformate definieren

Sie können Anzeigeformate definieren für Spalten vom Typ Zahl, Ganzzahl, lange Ganzzahl, alphanumerisch, Datum, Zeit und Bild.

Zeigen Sie z.B. in einer Spalte Löhne an, können Sie dafür ein Zahlenformat festlegen. Das folgende Format setzt ein sechsstelliges Format mit 1 Kommastelle + Euro-Zeichen: ###.##0,00 €.

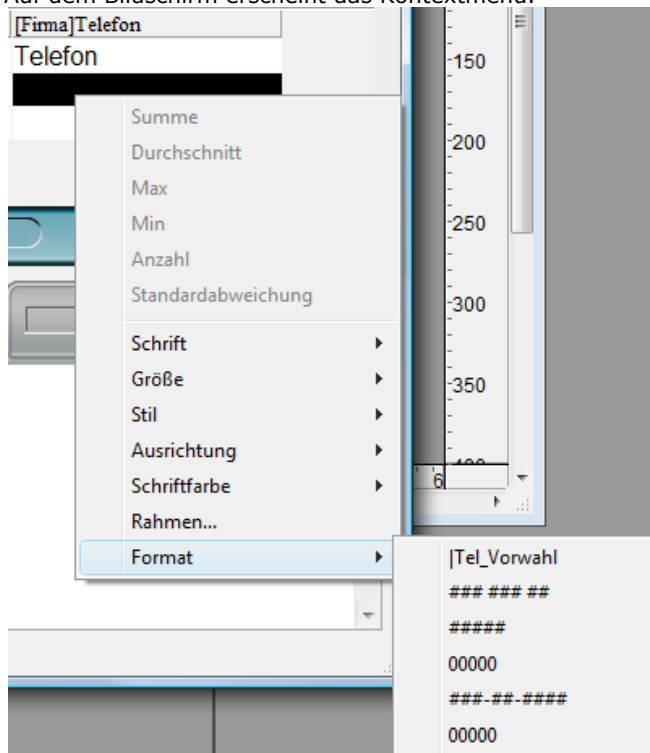
Es kann Beträge bis zu 999.999,99 € anzeigen.

Enthält Ihr Bericht alphanumerische Felder wie Telefon- oder Sozialversicherungsnummer, können Sie ein alphanumerisches Format festlegen, z.B. ###-##-####.

4D gibt verschiedene Formate vor. Sie können auch eigene Formate erstellen und dann für Schnellberichte verwenden. Weitere Informationen dazu finden Sie im Abschnitt **Anzeigeformate**.

Um ein Format für ein Datenfeld vom Typ Zahl, alphanumerisch, Datum oder Zeit einzugeben:

1. Klicken Sie mit der rechten Maustaste in eine Zelle der gewünschten Spalte.
Auf dem Bildschirm erscheint das Kontextmenü.



2. Wählen Sie im Untermenü **Format** das gewünschte **Anzeigeformat**.
Das Untermenü zeigt die Formate, die dem Datentyp der Spalte entsprechen. Das ist in unserem Beispiel das Datenfeld „MengenPreis“, welches vom Typ Zahl ist. Enthält die Spalte nicht formatierbare Daten, erscheint der Befehl **Format** nicht im Kontextmenü.

Haben Sie auch eine Berechnung für die Spalte zugewiesen, gilt das Format der Zelle automatisch auch dafür. Davon ausgenommen ist das Format „Anzahl“. Es wird immer als Ganzzahl angezeigt und akzeptiert keine anderen Formatierungssymbole, wie z.B. das Euro-Zeichen. Sie können auch eigene Formate eingeben. Doppelklicken Sie dazu in die Zelle und geben die Elemente des Formats über die Tastatur ein.

Sie können jeder Spalte im Bericht ein anderes Format zuweisen.

Zeilen und Spalten aus- und einblenden

Mit 4D können Sie im Schnellbericht im Listenmodus Spalten bzw. Zeilen ein- und ausblenden. Zeilen ausblenden ist hilfreich, wenn der Bericht nur Berechnungen anzeigen soll. In diesem Fall blenden Sie den Datenteil aus, es erscheinen nur die Berechnungen in den Zeilen Gesamtsumme und Zwischensumme. Sie können ebenso die Zeile Umbruch oder Gesamtsumme ausblenden. Sie können eine Spalte ausblenden, wenn sie nur zur Sortierung dient und im Bericht nicht erscheinen soll. Sie können eine Spalte/Zeile entweder über Menübefehl oder über das Kontextmenü des Schnellberichteditors ausblenden

Hinweis: Im Kreuztabellen-Bericht können Sie Zeilen oder Spalten nicht ausblenden.

Um eine Zeile/Spalte auszublenden:

1. Klicken Sie in die Titelzeile, um die entsprechende Zeile/Spalte zu wählen.
2. Klicken Sie mit der rechten Maustaste in die Titelzeile einer Zeile oder Spalte und dann im Kontextmenü auf den Eintrag **Ausblenden**.

ODER

Wählen Sie im Schnellberichteditor im Menü **Spalten>Ausblenden**.

ODER

Klicken Sie der Werkzeugleiste **Spalten** auf die Schaltfläche .

4D zeigt die Spalte/Zeile in grau an, d.h. sie erscheint nicht, wenn Sie den Schnellbericht auf dem Bildschirm anzeigen oder ausdrucken.

	[Kunden]Firma	[Artikel]Artikel Name	[Positionen]MengenPreis
Kopfteil	Firma	Artikel Name	MengenPreis
Daten		### ## #	### ##0,00 €
Summe			Σ Summe

	[Kunden]Firma	[Artikel]Artikel Name	[Positionen]MengenPreis
Kopfteil	Firma	Artikel Name	MengenPreis
Daten		### ## #	### ##0,00 €
Summe			Σ Summe

Zeile/Spalte einblenden

Ist eine Zeile/Spalte ausgeblendet, ist der Eintrag **Ausblenden** im Menü **Spalten** oder im Kontextmenü mit einem Häkchen gekennzeichnet. Außerdem ist die entsprechende Schaltfläche in der Werkzeugleiste „Spalten“ markiert.

Wählen Sie den Eintrag **Ausblenden** im Kontextmenü oder im Menü **Spalten** erneut oder klicken auf die entsprechende Schaltfläche in der Werkzeugleiste, wird die Zeile/Spalte wieder einblendet.

Kopf- oder Fußzeilen hinzufügen

Sie können im Schnellbericht auch Kopf- und Fußzeilen hinzufügen. Dazu verwenden Sie die Dialogfenster für Kopf- und Fußteil. Sie können folgendes definieren:

- Im Kopf- oder Fußteil Text oder ein Bild einfügen
- Eingaben im linken oder rechten Teil oder in der Mitte setzen
- Die Größe des Kopf- bzw. Fußteils verändern
- Für Kopf- bzw. Fußteil Schriftart, -größe und -stil angeben
- Code eingeben, der im Bericht Seitennummern oder Datum und Uhrzeit hinzufügt

Kopf- und Fußteile können Sie nur für Berichte anlegen, die ausgedruckt werden. Wurden sie einmal angelegt, bleiben sie im Bericht erhalten, auch wenn sich das Ausgabeziel ändert. Weitere Informationen zu den verschiedenen Ausgabearten finden Sie im Abschnitt **Schnellbericht ausführen**.

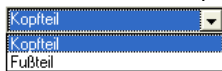
Um Kopf- und Fußteile hinzuzufügen:

1. Wählen Sie im Schnellberichteditor im Menü **Datei/Ablage>Kopf- und Fußteil**.

Auf dem Bildschirm erscheint das Dialogfenster für Kopf- und Fußteil.

Hinweis: Die Vorschau berücksichtigt die Konfiguration des Druckformats des Berichts. Sie können im gleichen Dialogfenster Kopf- und Fußteile angeben.

2. Wählen Sie im DropDown-Menü Kopfteil oder Fußteil:

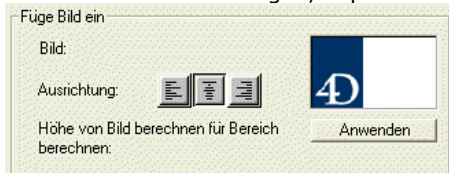


3. Geben Sie in „Höhe“ die gewünschte Höhe für den gewählten Teil ein.

Standardmäßig sind für Kopf- und Fußteil jeweils 25 Punkte angegeben. Sie können sowohl den Wert als auch die Einheit ändern.

Die eingegebene Höhe für Kopf- und Fußteil wird in der Seitenvorschau mit gestrichelten Linien angezeigt. So können Sie vorab sehen, wie Kopf- und Fußteil im gedruckten Bericht erscheinen.

4. Wollen Sie ein Bild einfügen, kopieren Sie es über die Standardbefehle Kopieren/Einsetzen in den Bildbereich:



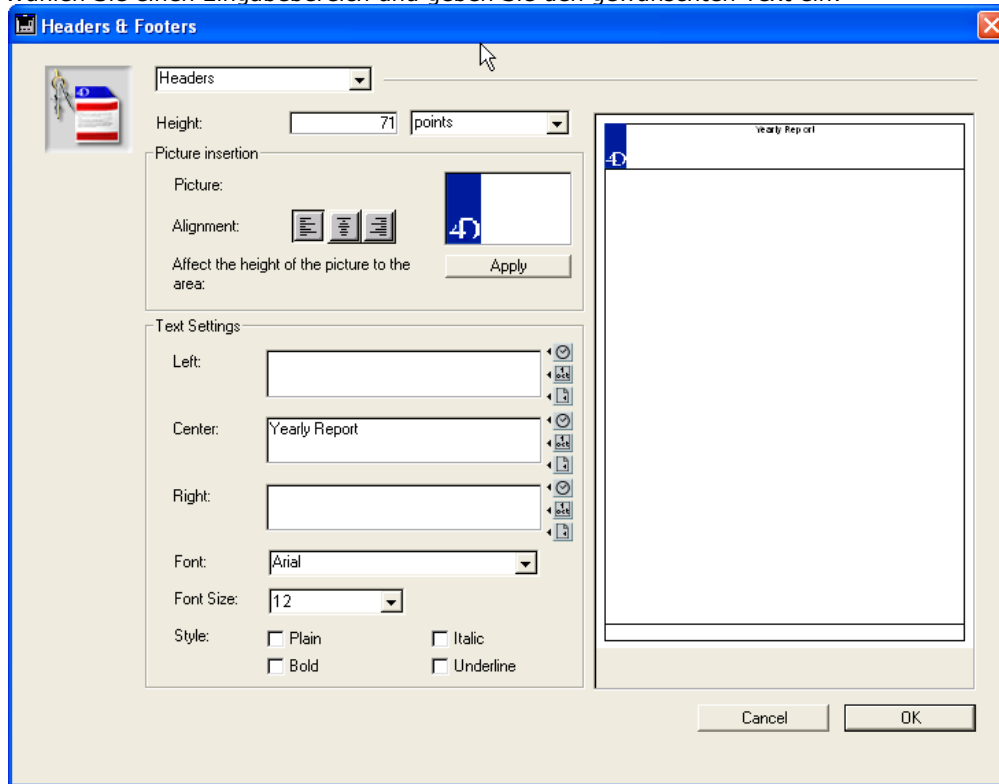
- Das Bild wird standardmäßig in der Mitte des Kopfteils bzw. Fußteils eingesetzt. Über die Icons für Ausrichtung können Sie eine andere Position definieren:



- Soll das Bild die Höhe des Kopf- bzw. Fußteils haben, klicken Sie auf die Schaltfläche **Anwenden**:



5. Wählen Sie einen Eingabebereich und geben Sie den gewünschten Text ein:



Über die Icons rechts neben dem Bereich können Sie zwischen drei Variablen wählen: aktuelle Seitennummer, aktuelle Ausführungszeit oder aktuelles Ausführungsdatum:



Sie können den Code für die Variablen auch direkt eingeben:

- #H für Zeit
- #D für Datum
- #P für Seitennummer

6. Weisen Sie die gewünschten Schrifteigenschaften zu und klicken auf die Schaltfläche **OK**, um die Eingaben zu bestätigen.

Schnellbericht ausführen

Haben Sie das Layout für Ihren Bericht fertiggestellt, können Sie den Schnellbericht „drucken“. Dafür gibt es verschiedene Möglichkeiten:

- Auf dem gewählten Standarddrucker
- In ein Textdokument auf der Festplatte
- In ein 4D View Dokument
- In ein HTML Dokument

Um ein Ausgabeziel zu definieren, Wählen Sie im Menü **Datei/Ablage** den Befehl **Ziel**.

Drucker

Diese Option verwendet den Drucker, den Sie in Ihren Druckereinstellungen gewählt haben. Vor dem Ausdruck können Sie eine Vorschau auf dem Bildschirm öffnen.

Um auf einem Drucker auszudrucken:

1. Wählen Sie im Menü **Datei/Ablage>Ziel** und dann im Untermenü den Eintrag **Drucker**
Diese Option ist standardmäßig ausgewählt.
2. Wählen Sie bei Bedarf im Menü **Datei/Ablage** den Befehl **Seite einrichten** und stellen Sie die Druckparameter ein.
Hinweis: Dieser Befehl ist nur verfügbar, wenn als Ziel **Drucker** ausgewählt wurde.
3. Wollen Sie erst eine Vorschau auf dem Bildschirm sehen, wählen Sie im Menü **Datei/Ablage>Vorschau**
Hinweis: Dieser Befehl ist nur verfügbar, wenn als Ziel **Drucker** ausgewählt wurde.
Auf dem Bildschirm erscheint die Vorschau des Berichts mit den aktuellen Daten der Anwendung.
4. Wählen Sie im Menü **Datei/Ablage>Erzeugen**.
5. Wählen Sie die geeigneten Einstellungen für Ihren Bericht und klicken auf die Schaltfläche **OK**.

Textdatei

Diese Option sendet Ihren Schnellbericht an die Festplatte. Sie können ihn mit anderen Anwendungen öffnen und verändern, z.B. Texteditoren und Tabellenkalkulation. Die Datensätze des Schnellberichts werden in eine Textdatei exportiert.

4D verwendet die Spaltentitel automatisch als den ersten exportierten "Datensatz".

Um den Bericht in einem Textdokument zu generieren:

1. Wählen Sie im Menü **Datei/Ablage>Ziel** und dann im Untermenü den Eintrag **Textdatei**.
2. Wählen Sie im Menü **Datei/Ablage>Erzeugen**.
4D zeigt den Standard Öffnen-Dialog mit blinkendem Cursor im Eingabebereich für Dateiname.
3. Geben Sie einen Dateinamen ein und klicken auf die Schaltfläche **OK**.
4D zeigt ein Dialogfenster mit einem Ablaufbalken, der das Fortschreiten der Operation angibt. Ist der Bericht in eine Datei übertragen, kehrt 4D zum Schnellberichteditor zurück. Denken Sie daran, das Ausgabeziel zu wechseln, wenn Sie den Schnellbericht zum Drucker senden wollen.

4D View Dokument

Diese Option leitet den Bericht an 4D View, das Plug-In zur Anzeige der Daten. Der Bericht erscheint dann als **4D View** Fenster und nicht in Tabellenform. 4D View verwendet alle Daten im Bericht.

Hinweis: Das Ausgabeziel **4D View** ist nur verfügbar, wenn dieses Plug-In in Ihrer 4D Umgebung installiert ist und Sie die entsprechenden Lizenzen erworben haben.

Um den Bericht als 4D View Dokument zu erstellen:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Ziel** und dann im Untermenü den Eintrag **4D View**.
2. Wählen Sie im Menü **Datei/Ablage** den Befehl **Erzeugen**.

4D erstellt ein 4D View Fenster mit den Daten. Sie können den Inhalt verändern und über 4D View Befehle drucken und sichern.

HTML Dokument

Diese Option sendet Ihren Schnellbericht an eine HTML Datei. Das ist die standardmäßige HTML Vorlage (template), außer sie wurde per Programmierung geändert.

4D verwendet die Spaltentitel automatisch als ersten exportierten "Datensatz".

Um eine HTML Datei zu drucken:

1. Wählen Sie im Menü **Datei/Ablage>Ziel** und dann im Untermenü den Eintrag **HTML-Datei**.
2. Wählen Sie im Menü **Datei/Ablage** den Befehl **Erzeugen**.
4D zeigt den Standard Öffnen-Dialog mit blinkendem Cursor im Eingabebereich für Dateiname.

3. Geben Sie einen Dateinamen ein und klicken auf die Schaltfläche **OK**.
4D zeigt ein Dialogfenster mit einem Ablaufbalken, der das Fortschreiten der Operation angibt.
Ist der Bericht in eine Datei übertragen, kehrt 4D zum Schnellberichteditor zurück. Denken Sie daran, das Ausgabeziel zu wechseln, wenn Sie den Schnellbericht zum Drucker senden wollen.

Der Zeichensatz zum Generieren von HTML Code wird in den Datenbank-Eigenschaften über den Parameter "Standardsatz" definiert (siehe Abschnitt **Seite Web/ Optionen (II)**).

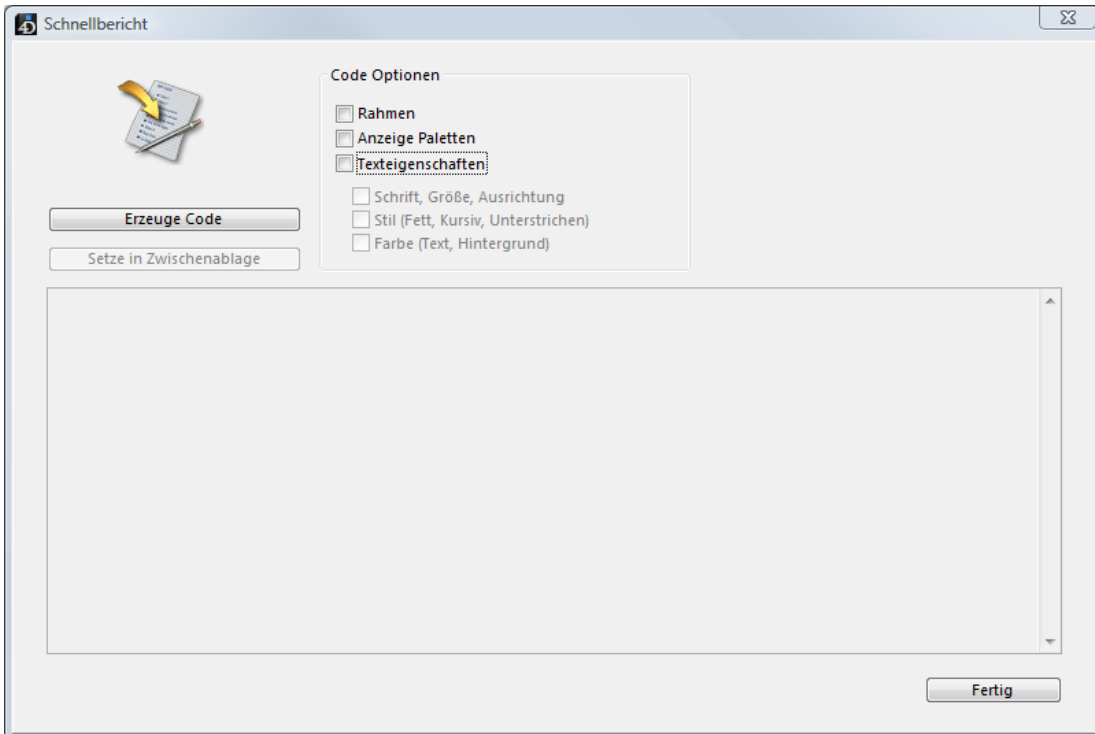
4D Code generieren

Im Schnellberichteditor lässt sich automatisch 4D Code generieren, der zu einem bestimmten Bericht gehört und in die Zwischenablage kopiert werden kann. Auf diese Weise können Sie manuell erstellte Vorgaben für Schnellberichte über die 4D Programmiersprache an Ihre Bedürfnisse anpassen (Befehle im Kapitel **Schnellbericht** der *4D Programmiersprache*). Diese Funktionalität ist nur im Designmodus verfügbar.

Die Schaltfläche zum Generieren von Code erscheint auf der letzten Seite des Schnellberichteditors:



Sie können die letzte Seite auch anzeigen und diese Funktionalität verwenden, wenn Sie den Schnellbericht manuell erstellt haben. Klicken Sie auf diese Schaltfläche, erscheint folgendes Dialogfenster:

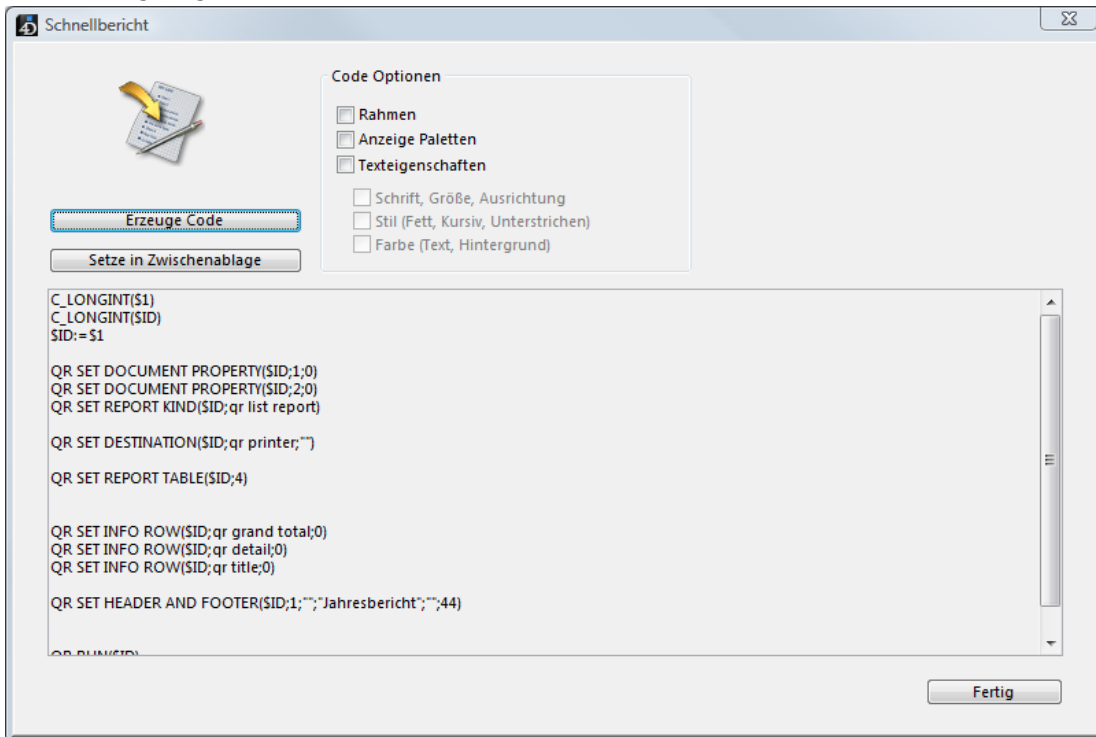


Sie können folgende Optionen einstellen:

- **Rahmen:** Damit integrieren Sie die Eigenschaften für Zellenränder im generierten 4D Code.
- **Anzeige Paletten:** Damit integrieren Sie den Anzeigestatus der aktuellen Paletten (Standard, Stile, Spalten...) im generierten 4D Code.
Wir empfehlen, diese Option zu deaktivieren, wenn der Code in einem Offscreen Bereich verwendet werden soll.
- **Texteigenschaften:** Damit geben Sie an, dass die für den Bericht definierten Schrifteigenschaften in den 4D Code integriert werden sollen. Ist diese Option nicht markiert, bleiben die drei darunterliegenden Optionen ohne Wirkung.
 - **Schrift, Größe, Ausrichtung:** Damit berücksichtigt der generierte 4D Code die Angaben für Schriftart, Schriftgröße und Ausrichtung aus der Berichtvorlage.
 - **Stil (Fett, Kursiv, Unterstrichen):** Damit berücksichtigt der generierte 4D Code die Angaben für Schriftstil aus der Berichtvorlage.

- **Farbe** (Text, Hintergrund): Damit berücksichtigt der generierte 4D Code die Angaben für Farben aus der Berichtsvorlage.

Haben Sie diese Optionen definiert, klicken Sie auf die Schaltfläche **Erzeuge Code**. Der 4D Code wird erstellt und in der Vorschau angezeigt:



Sie können die Optionen bei Bedarf verändern und erneut auf die Schaltfläche **Erzeuge Code** klicken; in der Vorschau erscheint dann der geänderte Code.











Entspricht der Code Ihren Vorstellungen, klicken Sie auf die Schaltfläche **Setze in Zwischenablage** und klicken auf die Schaltfläche **Fertig**, um das Dialogfenster zu schließen.

Wählen Sie unter Windows die Kombination **Strg+V**, auf Mac OS **Befehlstaste+V** oder im Menü **Bearbeiten> Einfügen**, um den Code in eine Methode oder Textdatei zu übertragen.

Schnellberichte (64-bit)

Hinweis zur Kompatibilität

Willkommen zum neuen Schnellberichteditor von 4D. Dieser Editor ist in 64-bit Versionen der 4D Developer Edition verfügbar und wird in zukünftigen 4D Versionen Schritt-für-Schritt den standardmäßigen Editor in 32-bit ersetzen.

-  Überblick
-  Schnellberichte verwalten
-  Spalten und Bezeichnungen hinzufügen
-  Grafische Attribute eines Schnellberichts
-  Datensätze sortieren und Umbrüche hinzufügen
-  Berechnungen hinzufügen
-  Anzeigeformate für Spalte setzen
-  Zeilen oder Spalten ausblenden
-  Kopf- und Fußteile hinzufügen
-  Schnellbericht ausführen

Eine der wichtigsten Aufgaben in der Datenverwaltung ist das Erstellen von Berichten. Der Schnellberichteditor ist eine Möglichkeit zum Gestalten von Berichten. Damit erstellen Sie ad-hoc Berichte in der Design- und Anwendungsumgebung. Die andere Möglichkeit ist der **Formulareditor**. Damit gestalten Sie Berichte in der Designumgebung (Sie können diese anschließend in der Anwendungsumgebung verändern). Verwenden Sie ein Ausgabeformular für Berichte, die ein komplexes Layout oder Programmierung erfordern. Weitere Informationen dazu finden Sie im Kapitel **Ausgabeformulare und Berichte**.

Mit dem Schnellberichteditor können Sie:

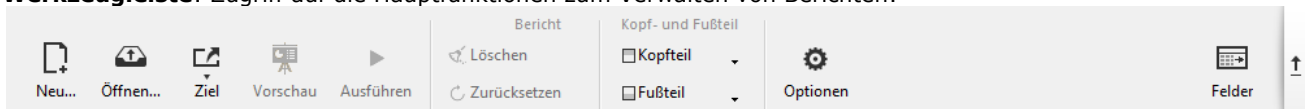
- Datensätze als Liste erstellen
- Umbrüche einfügen
- Berichte als Kreuztabelle erstellen
- Berechnungen in Summenzeilen durchführen
- Den Bericht mit Schriftarten und -stilen gestalten
- Für jede Zelle die Hintergrundfarbe definieren
- Gestaltete Berichte auf der Festplatte sichern und erneut aufrufen
- Zwischen verschiedenen Ausgabearten wählen, z.B. HTML oder Textdatei, Drucken oder auf Festplatte sichern

Der Schnellberichteditor lässt sich auch per Programmierung über die Befehle des Kapitels **Schnellbericht** steuern.

Beschreibung des Editors

Der Schnellberichteditor enthält folgende Elemente:

- **Werkzeugleiste:** Zugriff auf die Hauptfunktionen zum Verwalten von Berichten:



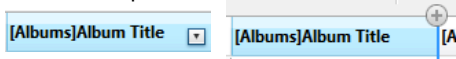
- **Neu...:** Erstellt ein neues leeres Bericht-Layout. Sie können Liste oder Kreuztabelle auswählen (siehe unten).
 - **Öffnen:** Verwalten die Schnellberichtdateien. Weitere Informationen dazu finden Sie im Abschnitt **Schnellberichte verwalten**.
 - **Ziel:** Bestimmt den Ausgabebetyp des Berichts. Weitere Informationen dazu finden Sie im Abschnitt **Schnellbericht ausführen**.
 - **Vorschau:** Verwendet die Vorschau Ihres Betriebssystems zum Anzeigen der Druckvorschau des Berichts.
 - **Ausführen:** Generiert den Bericht mit den aktuellen Daten im definierten Zielformat (siehe Abschnitt **Schnellbericht ausführen**).
 - **Löschen:** Hebt alle Spalten und Zeilen des Berichtlayouts auf, so dass Sie erneut mit einem leeren Fenster starten können. Diese Schaltfläche zeigt zuerst ein Dialogfenster zum Bestätigen.
 - **Zurücksetzen:** Ersetzt das aktuelle Berichtlayout mit der zuletzt auf der Festplatte gesicherten Version. Diese Schaltfläche ist deaktiviert, wenn das angezeigte Layout nicht in einer Datei auf der Festplatte gesichert wurde.
 - **Kopfteil** und **Fußteil:** Setzt Kopf- und Fußteilbereich des Schnellberichts. Weitere Informationen dazu finden Sie im Abschnitt **Kopf- und Fußteile hinzufügen**.
 - **Optionen:** Zeigt Optionen für die Ausgabe des Berichts. Die Anzeige richtet sich nach der aktuellen Ausgabeoption des Berichts.
 - **Felder:** Öffnet das Fenster mit allen Feldern der aktuellen Tabelle sowie den verknüpften Tabellen im linken Teil, und allen Spalten des Berichts im rechten Teil. Sie können leicht Felder im Bericht hinzufügen, verschieben oder löschen (siehe **Bericht über Liste der Felder definieren**).
 - **Leiste ausblenden:** Klicken Sie auf die Schaltfläche mit vertikalem Pfeil, um die Werkzeugleiste ein- oder auszublenden.
- **Berichtsgestaltung und Vorschaubereich:** Hier können Sie die Vorlage Ihres Berichts erstellen und über das Kontextmenü oder die Schaltfläche (+) Felder einfügen; Sie können auch die Spaltenbreite anpassen, Summenzeilen und Formeln hinzufügen/entfernen, Farbe für Zellenhintergrund oder Text setzen, u.v.m.

Column data sources

	[Albums]Album Title	[Albums]Musician	[Albums]Purchase Price
Title row	Title	Musician	Price
Detail row (body)	Format		
Subtotal row	[Albums]Musician changed		
Total row	Grand total		

- **Zeile Titel:** Zeigt die Namen der Felder oder Formeln, die im Bericht eingefügt wurden. Diese Zeile wird für jede Seite des Berichts wiederholt. Der Schnellberichteditor fügt Feldnamen standardmäßig ein, Sie können den Inhalt verändern.
- **Zeile Format:** Enthält Daten aus dem Datensatz und wird im Bericht für jeden Datensatz wiederholt. Hier können Sie, je nach Datentyp, ein Anzeigeformat zuweisen.
- **Summenzeile(n):** Diese Zeilen zeigen Zwischen- und Gesamtsumme und die jeweils zugeordnete Bezeichnung. Es wird eine Zeile pro Sortierung angelegt.

- **Datenquellen für Spalte:** Diese Titel geben die Datenquelle für jede Spalte an.
- **Kontextmenüs und lokale Menüs:** Diese Menüs bieten Zugriff auf verschiedene Funktionen des Editors, wie z.B. Neue Spalte oder Zelleneigenschaften ändern.



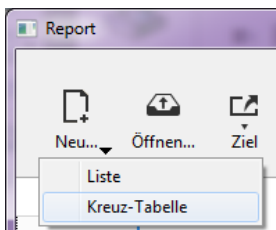
Die folgende Grafik zeigt den Zusammenhang zwischen den Spezifikationen im Layout und der gedruckten Ausgabe:

[Albums]Music Category	[Albums]Album Title	[Albums]Purchase Price
Title	Genre	Title Purchase Price
Format		
[Albums]Music Category changed		Total: Σ
Grand total		Grand total: Σ

Genre	Title	Purchase Price
Blues	Best of B. B. King	19.55
	Total :	19.55
Classical	Rhapsody in Blue, An American in Paris	12.95
	Virtuoso - Ludwig Van Beethoven	10.55
	Brahms Piano Quintet - Clarinet Quintet	0
	Season for Love	12.99
	Total :	36.49
Country	Lucille and Other Classics by Kenny Rogers	11.5
Rock	The Long Run	14.99
	Total :	14.99
Soul	The Best of the Stylistics	8.2
	Temptations 25th Anniversary Volume II	10.55
	Best of Gladys Knight & the Pips, 1973-1988	6.99
	Bad	20.99
	Double Good Everything	12.99
	Gettin' Ready	14
	Kool & The Gang Spin Their Top Hits	15.5
	Total :	89.22
	Grand total :	273.53

Modus Liste und Kreuztabelle

Der Schnellberichteditor bietet zwei Modi zum Erstellen von Berichten: Liste und Kreuz-Tabelle



• Liste

Dies ist der Standardmodus. Er zeigt Datensätze als Liste mit Umbrüchen an der Stelle, wo Summen berechnet wurden. Hierzu ein Beispiel:

Vorname	Nachname	Abteilung	Gehalt
Franz	Becker	Technik	4.000,00 €
Peter	Eigenrauch		5.000,00 €
Hubert	Hagen		4.000,00 €
Alexander	Maisberger		6.500,00 €
Tom	Müller		4.500,00 €
Martin	Redlin		6.000,00 €
Franz	Westermeier		5.500,00 €
		Gehalt für Abteilung Technik	35.500,00 €
			7
Birgitte	Haindl	Vertrieb	7.500,00 €
Henry	Hansen		5.000,00 €
Bernhard	Herlitz		6.900,00 €
Josef	Huber		8.000,00 €
Johanna	Kisch		6.700,00 €
Sabine	Oswald		4.500,00 €
Barbara	Schmitt		4.800,00 €
		Gehalt für Abteilung Vertrieb	43.400,00 €
			7
		Gehalt für alle Abteilungen	78.900,00 €

• Kreuz-Tabelle

Dieser Modus stellt Ihren Bericht als zweidimensionale Tabelle dar. Er eignet sich zum Anzeigen von Daten aus einer Datenquelle mit Kategorien, die aus zwei anderen Datenquellen stammen.

In einer Kreuztabelle können Sie beispielsweise anzeigen, wieviel von jeder Produktart pro Quartal verkauft wurden.

Hierzu ein Beispiel:

	Remote control	Power module	AV Preamplifier	Zwischensumme
Q3	71,00	64,00	65,00	200,00 6,45
Q1	46,00	51,00	21,00	118,00 5,61
Q2	102,00	40,00	17,00	159,00 6,36
Q4	82,00	29,00	17,00	128,00 5,56
Gesamtsumme	301,00 6,02 1,00	184,00 6,57 1,00	120,00 5,45 1,00	605,00 6,05 1,00

Schnellberichte verwalten

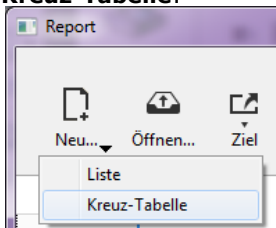
Schnellbericht erstellen

Mit dem Schnellberichteditor können Sie Berichte in der Design- und Anwendungsumgebung erstellen. Sie können den Berichteditor in seinem eigenen Fenster öffnen oder in einem Formularbereich einfügen.

Hinweis zur Kompatibilität: Plug-In Bereiche des bisherigen 4D Schnellberichteditors werden unterstützt und direkt in Unterformularbereiche konvertiert.

Um einen Schnellbericht im Editor anzulegen:

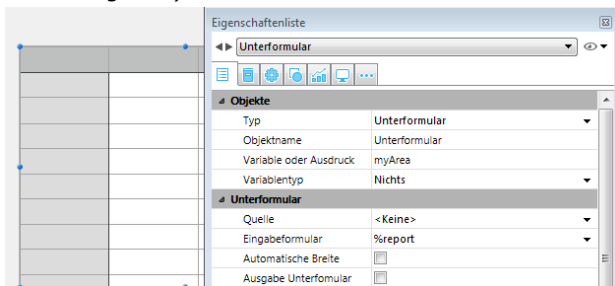
1. Wählen Sie im Menü **Bericht** den Befehl **Schnellbericht** oder klicken Sie in der Werkzeugleiste auf das Icon **Tools>Schnellbericht**.
4D zeigt den Schnellberichteditor standardmäßig im Listenmodus an. Wurde bereits ein Bericht angelegt, erscheint er im Fenster des Editors. Zum Entfernen klicken Sie in der Werkzeugleiste des Editors auf das Icon **Löschen** oder **Neu**.
2. Wollen Sie den Bericht als Kreuztabelle anlegen, klicken Sie in der Werkzeugleiste auf das Icon **Neu** und markieren **Kreuz-Tabelle**:



3. Starten Sie mit dem Anlegen Ihres Berichts.

Um einen Schnellbericht in einem Formular anzulegen:

1. Fügen Sie einen Bereich Unterformular hinzu und weisen folgende Eigenschaften zu:
 - "Ausgabe Unterformular": diese Option deaktivieren
 - "Eingabeformular": das DropDown-Menü aufklappen und den ersten Eintrag **%report (4D Report)** wählen
 - "Variable oder Ausdruck": Referenz des Bereichs eintragen (wird den Befehlen im Kapitel **Schnellbericht** übergeben)



Hinweis: Weitere Informationen dazu finden Sie im Abschnitt **Unterformulare als Seite**

2. In der Formularmethode initialisieren Sie das Feld mit folgendem Code:

```
if(Form event=On Load)
  C_LONGINT(myArea)
  QR SET REPORT TABLE(myArea;1) //Nummer der Berichtstabelle übergeben
End if
```

3. Definieren Sie den Inhalt des Berichts über das Kontextmenü oder über Befehle der Programmiersprache.

Schnellberichtvorlage laden und sichern

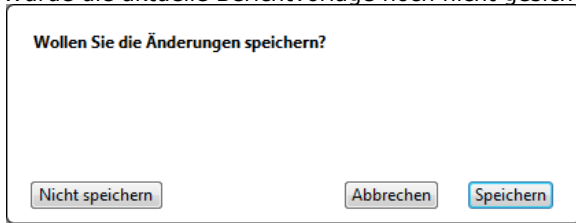
Sie können eine Berichtvorlage als Datei sichern und dann erneut über den Schnellberichteditor öffnen. Die Vorlage enthält alle Einstellungen für den Bericht ohne die Daten. Sie können verschiedene Vorlagen sichern und so eine Bibliothek mit Berichtvorlagen einrichten, die Sie dann je nach Bedarf aufrufen können.

Ist der Berichteditor geöffnet, können Sie eine gespeicherte Vorlage laden und zum Drucken eines neuen Berichts verwenden. Sie können dieselbe Vorlage auch mehrmals verwenden, um unterschiedliche Auswahlen von Datensätzen zu drucken.

Um eine Berichtvorlage zu sichern:

1. Schließen Sie den Schnellberichteditor oder klicken Sie in der Werkzeugleiste des Editors auf die Schaltfläche **Neu** oder **Öffnen**.

Wurde die aktuelle Berichtvorlage noch nicht gesichert, erscheint folgendes Fenster:



2. Klicken Sie auf **Speichern**.
4D zeigt einen Sichern-Dialog, wo Sie der Vorlage einen Namen geben können. Dateien für Schnellberichte haben die Endung ".4qr".
Hinweis: Schließen Sie den Editor ohne Sichern des Berichts, bleibt er trotzdem bis zum Beenden der Anwendung im aktuellen Speicher erhalten. Er wird also beim erneuten Öffnen des Editors weiterhin angezeigt.
3. Geben Sie einen Namen für die Vorlage ein und klicken auf die Schaltfläche **Speichern**.
4D sichert den Bericht als Datei, die Sie mit dem Schnellberichteditor erneut öffnen können. Sie können die Datei auf Ihrer Festplatte sichern. Verändern Sie einen Bericht und sichern ihn erneut, wird der alte Bericht überschrieben. Der Standard Sichern-Dialog erscheint nicht.

Um eine auf der Festplatte gespeicherte Berichtvorlage zu laden:

1. Klicken Sie in der Werkzeugleiste des Editors auf die Schaltfläche **Öffnen**
ODER
Klicken Sie in der Werkzeugleiste zur Dateiverwaltung auf die Schaltfläche **Öffnen**.
4D zeigt den Datei-Öffnen-Dialog mit der Liste der verfügbaren Berichtvorlagen.
2. Doppelklicken Sie auf einen Dateinamen oder wählen Sie einen Dateinamen in der Liste und klicken auf die Schaltfläche **Öffnen**.
4D ersetzt die aktuelle Vorlage durch die neu geöffnete Vorlage.

Hinweis zur Kompatibilität: Berichte, die mit dem vorigen Schnellberichteditor von 4D erstellt wurden, werden unterstützt; Sie können bisherige Berichte öffnen, bearbeiten und sichern, ohne ihr internes Format zu ändern.

Spalten und Bezeichnungen hinzufügen

Sie erstellen einen Bericht als Liste durch Hinzufügen von Spalten mit Feldreferenzen oder Formeln, einen Bericht als Kreuztabelle durch Hinzufügen von Feldreferenzen oder Formeln in den Zellen mit Daten.

Für beide Berichtstypen können Sie Spalten mit Feldern auf zwei Arten hinzufügen und organisieren:

- direkt im Vorschaubereich des Schnellberichteditors
- über das Fenster mit der Liste der Felder mit Namen "field picker"

Weitere Informationen zum Hinzufügen von Formeln finden Sie im Abschnitt **Berechnungen hinzufügen**.

Arbeitsweise für Datenfelder

- Arbeiten Sie im Designmodus, bewertet der Schnellberichteditor alle Verknüpfungen als automatisch, so dass Sie auf alle verknüpften Daten der Datenbank zugreifen können. Wird der Editor über den Befehl **QR REPORT** im Anwendungsmodus aufgerufen, muss der Entwickler selbst den Status der Verknüpfungen verwalten.
- Ausgeblendete Tabellen und Felder erscheinen nicht in der Liste der Felder des Schnellberichteditors. Weitere Informationen dazu finden Sie in der Beschreibung "Invisible" in den Abschnitten **Tabelleneigenschaften** und **Feldeigenschaften**.
- Der Schnellberichteditor unterstützt keine Felder vom Typ Objekt und Blob.

Spalten hinzufügen

Sie können mit Spalten direkt in der Berichtvorlage arbeiten oder das Fenster mit der Liste der Felder verwenden (siehe nächster Abschnitt).

Spalten hinzufügen

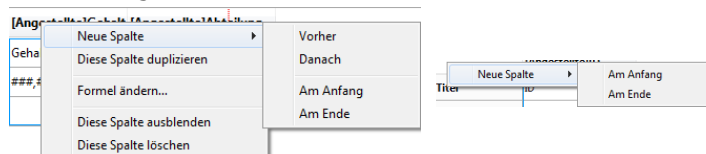
Sie können in der Berichtvorlage Spalten hinzufügen, einfügen oder duplizieren. Beim Hinzufügen verwendet der Editor automatisch die nächste verfügbare Spalte der aktuellen Tabelle, in der Reihenfolge der Erstellung.

Sie können das Icon (+) verwenden, das am Anfang des Spaltentrenners erscheint:



- Klicken Sie direkt darauf, wird eine neue Spalte angefügt
- Anklicken mit rechter Maustaste öffnet ein Kontextmenü, um entweder alle Felder der aktuellen Tabelle oder eine Formel einzufügen.

Jede Spalte des Berichts hat ein Kontextmenü, über das Sie eine andere Spalte einfügen können, und zwar vor oder nach der vorhandenen Spalte oder am Anfang oder Ende des Berichts. Sie können auch in der oberen linken Ecke der Berichtvorlage ein Kontextmenü aufrufen:



Sie können eine Spalte auch duplizieren. Dazu klicken Sie mit rechter Maustaste in den Kopfteil und wählen im Kontextmenü den Eintrag **Diese Spalte duplizieren**. Dann wird eine identische Spalte rechts neben die ursprüngliche Spalte gesetzt.

Sie können eine Spalte jederzeit verschieben. Dazu klicken Sie in den Kopfteil und ziehen die Spalte an die neue Position.

Spalten löschen

Beim Definieren von Feldern für den Schnellbericht können Sie angelegte Spalten wieder entfernen.

Doppelklicken Sie in den Kopfteil der gewünschten Spalte oder mit rechter Maustaste und wählen im Kontextmenü den Eintrag **Diese Spalte löschen**.

Spalten ersetzen

Sie können ein Feld bzw. ein Feld mit einer Formel ersetzen.

1. Doppelklicken Sie in den Spaltentitel oder mit rechter Maustaste und wählen im Kontextmenü den Eintrag **Formel ändern**. Der standardmäßige Formeleditor von 4D erscheint, wo Sie eine neue Datenquelle bestimmen können (Datenfeld oder Formel). 4D druckt das Formelergebnis für jeden Datensatz, der in der Zeile **Format** erscheint. Weitere Informationen dazu finden Sie im Abschnitt **Formeln hinzufügen**.

Bericht über Liste der Felder definieren (Listenbericht)

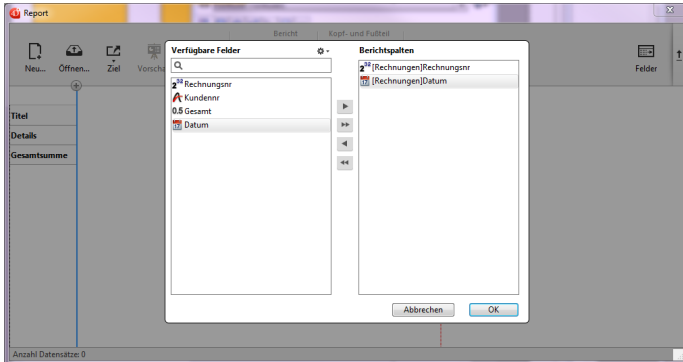
Klicken Sie auf die Schaltfläche **Felder**, um das Fenster mit den verfügbaren Feldern und zum Organisieren der Spalten zu öffnen:



Felder

Es enthält zwei Listen:

- **Verfügbare Felder** zeigt alle Felder der aktuellen Tabelle und die Felder der verknüpften Tabellen
- **Berichtspalten** zeigt die Felder, die aktuell im Bericht eingefügt sind

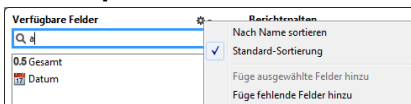


Über die Pfeilsymbole zwischen beiden Listen können Sie Felder in der Liste **Berichtspalten** hinzufügen und entfernen. Sie können Felder auch per Doppelklick hinzufügen oder Felder direkt nach oben oder unten ziehen und so die Reihenfolge verändern.

Klicken Sie auf die Schaltfläche **OK**, werden die Änderungen automatisch im Bericht selbst wiedergegeben. Fügen Sie z.B. in der Liste **Berichtspalten** ein Feld hinzu oder entfernen es, wird die entsprechende Spalte auch im Bericht hinzugefügt bzw. entfernt; dasselbe gilt für eine geänderte Reihenfolge.

Hinweis: Der Suchbereich zu Beginn der Liste **Verfügbare Felder** führt eine Suche vom Typ "enthält" aus; er zeigt z.B. alle Felder, die im Feldnamen den/die eingegebenen Buchstaben enthalten.

Über das Menü Optionen  können Sie die Liste **Verfügbare Felder** sortieren oder rasch mehrere Felder in die Liste **Berichtspalten** setzen.



- **Nach Name sortieren:** Sortiert die Felder alphabetisch
- **Standard-Sortierung:** Sortiert die Felder in der Reihenfolge der Erstellung
- **Füge ausgewählte Felder hinzu:** Fügt die Felder, so wie sie ausgewählt sind, in der Liste **Berichtspalten** hinzu
- **Füge fehlende Felder hinzu:** Fügt alle noch nicht enthaltenen Felder in der Liste **Berichtspalten** hinzu

Werte in Datenzellen hinzufügen oder ändern (Kreuztabellenbericht)

Sie können Werte in verschiedenen Datenzellen (Spalten, Zeilen und Zellen) hinzufügen oder ändern, entweder über das Dialogfenster **Felder** oder durch Doppelklick auf die jeweilige Datenzelle.

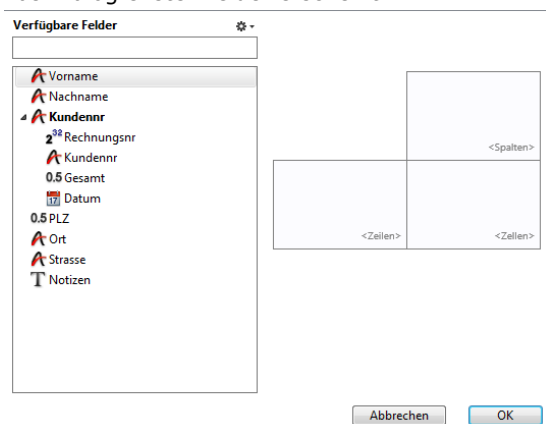
Über das Dialogfenster **Felder**

Um Datenzellen Felder zuzuweisen, klicken Sie in der Werkzeugleiste des Schnellberichteditors auf die Schaltfläche **Felder**:



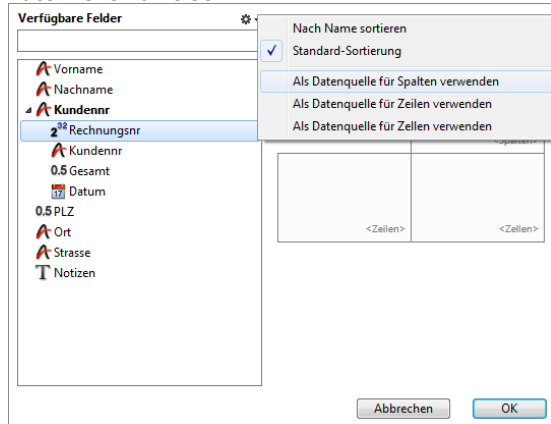
Felder

Das Dialogfenster **Felder** erscheint:



Es gibt drei Möglichkeiten, um Felder in Datenzellen hinzuzufügen oder zu organisieren:

- Doppelklick auf einen Eintrag in der Liste Verfügbare Felder, um ihn der ersten leeren Datenzelle zuzuweisen (Reihenfolge ist Spalten, Zeilen, Zellen)
- Einen Eintrag in der Liste Verfügbare Felder markieren und in eine Datenzelle im rechten Bereich ziehen.
- Einen Eintrag in der Liste Verfügbare Felder wählen und über den entsprechenden Befehl im Kontextmenü einer Datenzelle zuweisen:



Formeleditor verwenden

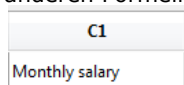
Sie können einer Datenzelle eine Formel zuweisen oder die vorhandene bearbeiten. Durch Doppelklicken in den Zellenbereich erscheint der Formeleditor. Hier können Sie ein Feld auswählen oder eine Formel für die Zelle schreiben. Der Formeleditor lässt sich auch im Kontextmenü der Datenzelle über den Befehl **Formel ändern** aufrufen.

Formeln hinzufügen

Sie können eine Formel im Schnellbericht hinzufügen. Sie können z.B. eine Formel hinzufügen, die aus dem Feld Jahreslohn den Monatslohn eines Angestellten berechnet

Um eine Spalte mit Formel hinzuzufügen:

1. Fügen Sie eine neue Spalte ein (siehe oben)
2. Doppelklicken Sie auf den Spaltentitel oder klicken Sie mit rechter Maustaste darauf und wählen im Kontextmenü den Eintrag **Formel ändern**.
4D zeigt den **Formeleditor** an, er zeigt den aktuellen Inhalt der Spalte. Die Formel überschreibt den bisherigen Inhalt der Spalte.
Hinweis: Stellen Sie sicher, dass die Formel nicht die aktuelle Auswahl verändert. Denn das verursacht Probleme bei Drucken des Berichts, da er auf der aktuellen Auswahl basiert.
3. Erstellen Sie die Formel durch Auswählen von Felder, Operatoren, 4D Befehlen bzw. Methoden ODER
Klicken Sie auf die Schaltfläche **Laden**, um eine vorhandene Formel von der Festplatte zu laden.
Weitere Informationen zum Formeleditor finden Sie im Kapitel **Formeleditor**.
Um die Formel als Datei zu sichern, die Sie später für andere Spalten oder Berichte wiederverwenden können, klicken Sie auf die Schaltfläche **Sichern** und geben im Sichern-Dialog einen Namen ein.
4. Klicken Sie auf die Schaltfläche **OK**, wird die Formel der Spalte zugewiesen.
4D fügt in der Spalte eine neue Bezeichnung ein, die sie als Formel kennzeichnet. Sie können die Spalte umbenennen, also im Kopfteil der Spalte eine andere Bezeichnung eingeben. Die Formeln haben die Bezeichnung **C1** bis **Cn**. Das sind die Variablennamen, die den aktuellen Wert der Spalte enthalten. Diese Variablen können Sie in anderen Formeln verwenden.

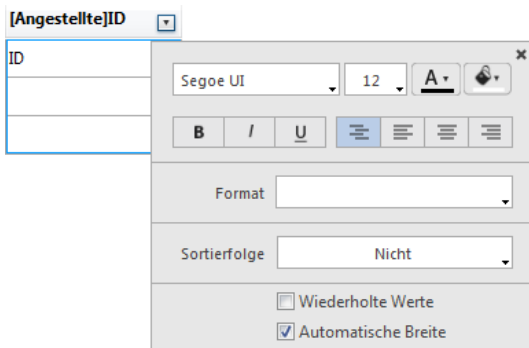


Hinweis: Die Cx Variablen für Schnellberichte müssen zum Kompilieren der Anwendung typisiert werden (siehe **Variablen im Schnellbericht**).

Spalten anpassen

Der Schnellberichteditor passt die Spaltenbreite automatisch an, d.h. die Option **Automatische Breite** ist standardmäßig aktiviert. Er richtet jede Spalte nach der max. Länge des Spaltentitels, der Daten sowie eingegebenen Bezeichnungen innerhalb der Spalte aus. Die Anpassung erfolgt beim Drucken des Berichts.

Ob die Option **Automatische Breite** zugewiesen wurde, sehen Sie, wenn Sie das DropDown-Menü im Spaltentitel aufrufen:



Wollen Sie die Breite einzelner Spalten sehen, klicken Sie in der Werkzeugleiste auf die Schaltfläche **Vorschau**.

Haben Sie die Option **Automatische Breite** gewählt, richtet sich die Spaltenbreite nach der maximalen Breite der Daten in den ausgewählten Datensätzen. Wählen Sie andere Datensätze, kann das eine andere Spaltenbreite ergeben.

Sie können eine Spalte manuell anpassen, dann wird die Option **Automatische Breite** deaktiviert. Der Text in der Spalte wird dann in den angegebenen Bereich eingepasst.

Text hinzufügen oder ändern

Sie können in der Berichtvorlage Bezeichnungen hinzufügen oder ändern. Haben Sie z.B eine Berechnung definiert, können Sie in anderen Zellen mit der Zeile Zwischensumme bzw. Gesamtsumme erklärenden Text hinzufügen. Es gibt folgende Möglichkeiten:

- Sie bearbeiten den Text, welchen 4D automatisch in der Titelzeile des Berichts vorgibt
- Sie fügen Text in leere Zellen der Zeile für Umbruch oder Summe ein
- Sie fügen den Wert eines Feldes Zwischensumme in die Zeilen für die Zwischensumme ein
- Sie definieren Schriftart, -größe, -stil oder Ausrichtung für beliebigen Text im Bericht

Um Text hinzuzufügen, doppelklicken Sie in der Vorlage des Schnellberichts auf eine leere Zelle. In der Zelle erscheint ein Einfügepunkt, so dass Sie eine Bezeichnung eingeben können:



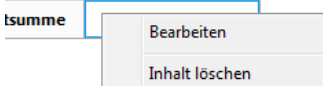
Hinweis: Es erscheint auch ein lokales Menü, so dass Sie auch eine Berechnung hinzufügen können:



Geben Sie eine Bezeichnung für die Berechnung ein, wählen Sie eine Zelle in derselben Zeile wie die Symbole für Berechnung. In einer Zelle mit Berechnung können Sie keinen Text eingeben.

Um Text zu ändern, doppelklicken Sie in die Zelle, damit dort ein Einfügepunkt erscheint und ziehen diesen durch den Text, den Sie ändern wollen. 4D markiert den gewählten Text.

Um Text in einer Zelle zu ändern, doppelklicken Sie in die Zelle, um in den Bearbeitungsmodus zu wechseln. Mit rechtem Mausklick können Sie ein Kontextmenü anzeigen, um den Zelleninhalt zu bearbeiten oder zu entfernen:



Grafische Attribute eines Schnellberichts

Sie können das Erscheinungsbild eines Schnellberichts verändern. Mit dem Schnellberichteditor können Sie folgende Attribute definieren:

- Schriftart, -größe, -stil, Ausrichtung und Farbe
- Hintergrundfarbe der Zellen, sowie wechselnde Hintergrundfarbe

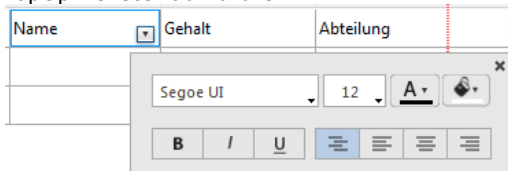
Hinweis: Die Farbangaben für die Darstellung des Berichts werden nur für die Ausgabeziele „Drucker“ und „HTML“ berücksichtigt. Weitere Informationen dazu finden Sie im Abschnitt **Schnellbericht ausführen**.

Schrift- und Farbeigenschaften zuweisen

Sie können beim Anlegen des Schnellberichts verschiedene Schriftarten, -größen, -stile, die Ausrichtung, sowie Farben für Schrift und Hintergrund festlegen. Das ist möglich für Text, Daten und Summenberechnungen in Zeilen, Spalten oder Zellen.

Um eine Schrifteigenschaft oder Farbe zu definieren:

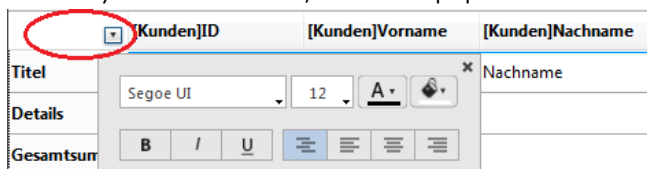
1. Setzen Sie die Maustaste in die untere rechte Ecke des entsprechenden Elements und klicken auf den Pfeil, um das PopUp-Fenster aufzurufen:



Hinweis: Je nach Art des Elements zeigt das Fenster weitere Optionen im unteren Teil

2. Im oberen Teil wählen Sie die gewünschten grafischen Eigenschaften aus:
 - Schrift
 - Schriftgröße
 - Schriftfarbe
 - Hintergrundfarbe
 - Schriftstil
 - Ausrichtung des Inhalts

4D weist die Eigenschaften dem Text, den Daten oder Berechnungen zu, die im gewählten Bereich enthalten sind. Sie können Eigenschaften für alle Bereiche des Berichts definieren. Dazu klicken Sie links oben in der ersten Zelle des Berichtlayouts auf den Pfeil, um das PopUp-Fenster aufzurufen.

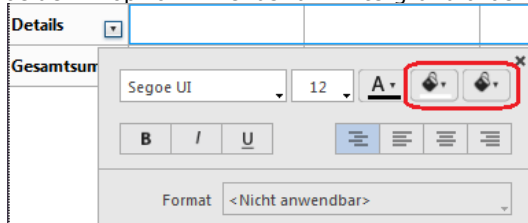


Hinweis: Gewählte Optionen für die Ausrichtung erscheinen nicht in der Berichtsvorschau, werden aber in den generierten Berichten angewandt.

Wechselnde Hintergrundfarbe festlegen

Im Schnellberichteditor können Sie zwei wechselnde Farben setzen. Das macht die Tabellen leichter lesbar.

Um eine wechselnde Hintergrundfarbe zu setzen, öffnen Sie das PopUp-Fenster der Zeile **Detail** und definieren über die beiden DropDown-Menüs für Hintergrundfarbe zwei Farben:



Im folgenden sehen Sie eine Vorschau für einen Bericht mit wechselnden Hintergrundfarben:

Music Category	Album Title	Musician
Soul	Bad	Michael Jackson
Blues	Best of B. B. King	B. B. King
Soul	Best of Gladys Knight & the Pips, 1973-1988	Gladys Knight & the Pips
Classical	Brahms Piano Quintet - Clarinet Quintet	Benda Musicians, The
Easy Listening	Carpenters - Their Greatest Hits	Carpenters, The
Soul	Double Good Everything	Smokey Robinson
Soul	Gettin' Ready	Temptations
Jazz	Jazzis Magazine April 1995 Collection	Various
Easy Listening	Johnny Mathis, 16 Most Requested Songs	Johnny Mathis

Hintergrundfarben werden auch in Berichten im HTML-Format berücksichtigt.

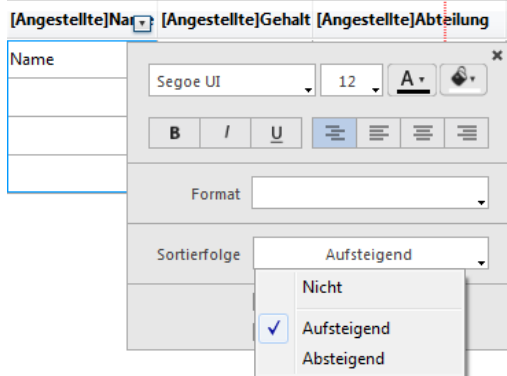
Datensätze sortieren und Umbrüche hinzufügen

Mit dem Schnellberichteditor können Sie die Datensätze in Ihrem Bericht auch sortieren. Dafür gibt es zwei Gründe:

- Sie wollen Datensätze in einer bestimmten Reihenfolge betrachten
- Sie wollen Datensätze in Gruppen mit Zwischensummen erstellen, um bestimmte Berechnungen durchzuführen

Sortierreihenfolge im Modus Liste erstellen

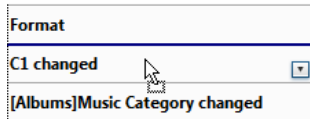
Sie können jederzeit eine Sortierreihenfolge definieren. Klicken Sie auf den Pfeil im Spaltentitel, öffnet sich das PopUp-Fenster mit dem Bereich Sortierfolge. Sie können **Aufsteigend**, **Absteigend** oder **Nicht** wählen:



Sie können Spalten mit Datenfeld oder mit Formel sortieren. Weitere Informationen dazu finden Sie im Abschnitt **Formeln hinzufügen**.

Sortierebene ändern

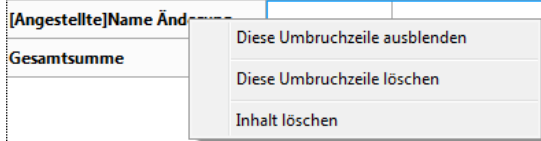
Sie können einen Bericht in mehreren Ebenen sortieren. Sortierebenen erscheinen in der Berichtvorlage als Umbruchzeilen (Zwischensummen). Wollen Sie die Position einer Sortierebene ändern, ziehen Sie die entsprechende Umbruchzeile per Drag-and-Drop nach oben oder nach unten:



Sortierung oder Umbruchzeile löschen

Sie können jederzeit eine Umbruchebe mit zugeordneter Sortierung der Spalte entfernen. Es gibt zwei Möglichkeiten:

1. Sie öffnen das PopUp-Menü der entsprechenden Spalte und wählen für Sortierfolge den Eintrag **Nicht**.
2. Sie wählen im Kontextmenü der Zeile den Eintrag **Diese Umbruchzeile löschen**.



Die Umbruchebe wird dann entfernt. Beachten Sie, dass die Spalte in beiden Fällen nicht mehr sortiert wird. Wollen Sie eine Spalte sortieren und keine Umbruchebe hinzufügen, wählen Sie im Kontextmenü **Diese Umbruchzeile ausblenden** (siehe **Zeilen oder Spalten ausblenden**).

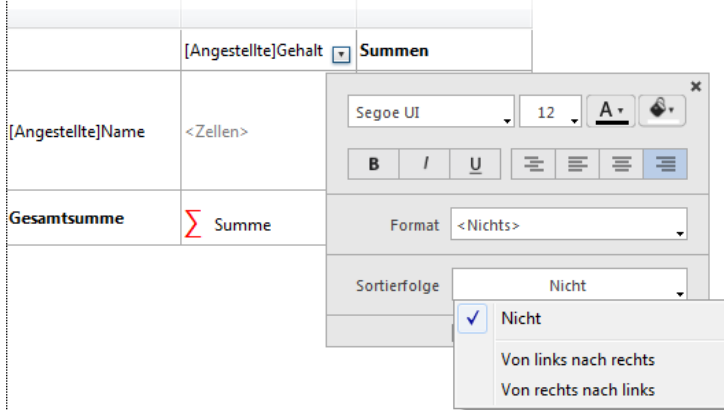
Werte in Kreuztabellen sortieren

In einem Kreuztabellen-Bericht können Sie nur Werte aus den horizontalen und vertikalen Datenquellen sortieren. Sie werden in der Endtabelle als Kategorien verwendet.

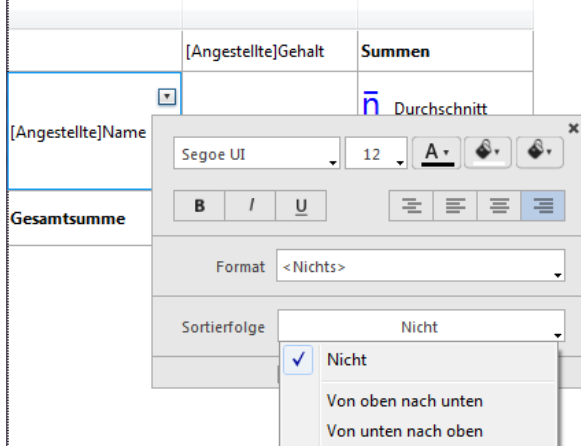
Um in einem Kreuztabellen-Bericht nach Kategorien zu sortieren, klicken Sie im Bereich Sortierfolge auf die Sortierbalken.

öffnen Sie das Kontextfenster für die Kategorie und wählen im DropDown-Menü **Sortierfolge** einen Eintrag:

- Zum Sortieren der horizontalen Datenquelle wählen Sie **Von links nach rechts** oder **Von rechts nach links**:



- Zum Sortieren der vertikalen Datenquelle wählen Sie **Von oben nach unten** oder **Von unten nach oben**:



Folgender Bericht wurde z.B. von rechts nach links und von unten nach oben sortiert:

Average duration of intervention

	Training	System	Software	Network	Hardware	Totals
NEY	01:14:04 179	00:17:49 341	00:50:34 155	00:51:31 387	00:25:28 411	00:39:05 1473
MURAT	01:15:04 248	00:17:29 386	00:49:21 180	00:51:26 494	00:23:37 530	00:39:16 1838
MUIRON	01:13:21 238	00:16:45 430	00:50:10 201	00:49:53 522	00:24:54 582	00:38:09 1973
MASSENA	01:13:42 218	00:17:51 360	00:49:38 190	00:49:10 429	00:25:03 506	00:38:34 1703
LANNES	01:14:19 188	00:16:07 372	00:48:58 143	00:50:17 431	00:25:14 499	00:37:30 1633
BONAPARTE	01:13:59 160	00:18:22 325	00:48:25 178	00:51:25 417	00:23:37 510	00:37:40 1590
BERTHIER	01:14:21 191	00:16:19 332	00:51:07 146	00:48:45 389	00:24:38 461	00:37:47 1519
Grand Total	1422	2546	1193	3069	3499	11729

Ebenen für Zwischensummen

In einem Schnellbericht setzen Sie Umbrüchebenen, um Datensätze nach Werten in einem oder mehreren Sortierfeldern zu trennen. Für jede Umbrüchebene wird ein Umbruchbereich gedruckt. Sie können im Umbruchbereich Berechnungen drucken. Die Berechnungen – Summe, Durchschnitt, Minimum, Maximum, Anzahl und Standardabweichung – werden für jede Datensatzgruppe berechnet.

Umbrüchebenen werden durch Sortier- und Umbrüchebenen bestimmt. Sortieren Sie z.B. Datensätze nach Verkaufszonen, fügt 4D nach jeder Datensatzgruppe mit derselben Zone einen Umbruch ein. Diese Zeilen werden automatisch eingefügt, wenn eine Sortierung definiert wurde.

Sie können in jeder Umbruchzeile eine Berechnung durchführen, z.B. Zwischensummen für den Produktverkauf jedes Landes innerhalb einer Verkaufszone. Weitere Informationen dazu finden Sie im Abschnitt **Summenberechnungen hinzufügen**.

	[Customers]Country	[Products]Label	[InvoiceLines]Quanti...
Title	Country	Article	Quantity
Format			
[Customers]Country changed			
[InvoiceLines]Quantity changed			
Grand total			

Die Bezeichnung einer Zeile mit Zwischensumme gibt an, welche Änderung im Wert den Umbruch auslöst.

Werte aus Umbruchfeldern für Bezeichnungen

Sie können die Darstellung und Verständlichkeit Ihrer Berichte verbessern, wenn Sie die Umbruchzeile mit dem Wert des Umbruchfeldes benennen.

Um den Wert eines Umbruchfeldes als Bezeichnung zu drucken, verwenden Sie das Nummernzeichen (#). Beispiel: Der Text "Gesamtlohn der Abteilung #" setzt den Namen der Abteilung, also den Wert des Feldes Abteilung an die Stelle des Platzhalters, wenn der Bericht gedruckt wird.

Der Platzhalter muss nicht in derselben Spalte wie das Umbruchfeld liegen. Er zeigt den Wert dieses Feldes in jeder beliebigen Zelle der Umbruchzeile an. Hierzu ein Beispiel:

	[Customers]Country	[Products]Label	[InvoiceLines]Quanti...
Title	Country	Article	Quantity
Format			
[Products]Label changed			
[Customers]Country changed		Total quantity for #	Σ
Grand total			Σ

In generierten Bericht wird pro Ländergruppe eine Umbruchzeile hinzugefügt:

Rye	91
	19
Soy	50
Total quantity for France	612

Berechnungen hinzufügen

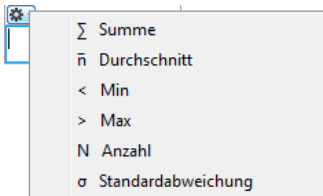
Summenberechnungen hinzufügen

Sie können in jeder Zeile für Summe oder Zwischensumme, im Modus Kreuztabelle in der Spalte Gesamtsumme, eine Berechnung auf den Inhalt von Feldern und Formeln hinzufügen.

Im Schnellberichteditor stehen im PopUp-Menü für eine Zeile Gesamtsumme bzw. Zwischensumme folgende Optionen zur Verfügung:



Diese Optionen erscheinen auch im Kontextmenü. Es erscheint, wenn Sie per Doppelklick in eine Zelle der Zeilen Zwischensumme und Gesamtsumme klicken:



- **Sum** berechnet die Gesamtsumme der Werte im Bericht oder Umbruch.
- **Durchschnitt** berechnet den Durchschnitt der Werte im Bericht oder Umbruch.
- **Min** zeigt den niedrigsten Wert im Bericht oder Umbruch an.
- **Max** zeigt den höchsten Wert im Bericht oder Umbruch an.
- **Anzahl** berechnet die Anzahl der Datensätze im Bericht oder Umbruch.
- **Standardabweichung** zeigt die Quadratwurzel der Varianz des Berichts oder Umbruchs (Varianz ist ein abweichender Wert des Durchschnitts).

Die Berechnungen **Summe**, **Minimum**, **Maximum** und **Durchschnitt** funktionieren nur in einem Feld vom Typ Zahl oder einer Formel.

Bericht als Liste

Setzen Sie die Berechnung in der Zeile Gesamtsumme, wird die Berechnung für alle Datensätze im Bericht ausgeführt. Setzen Sie die Berechnung in der Zeile Zwischensumme, wird für die Datensätze zu jeder Umbruchzeile eine separate Berechnung ausgeführt.

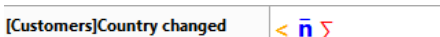
Bericht als Kreuztabelle

Summenberechnungen werden folgendermaßen verwendet:

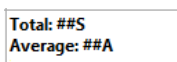
	[Invoices]Item	Totals
[Invoices]Quarter	[Invoices]Quantity Σ Sum 1 n̄ Average	Σ Sum 2 n̄ Average
Grand Total	Σ Sum 3 n̄ Average < Min	Σ Sum 4 n̄ Average < Min

- 1** = Gilt für jede Zelle der Tabelle
- 2** = Gilt für Werte in jeder Zeile
- 3** = Gilt für Werte in jeder Spalte
- 4** = Gilt für Werte in der letzten Spalte und letzten Zeile

4D setzt das/die markierten Icons in die gewählte Zelle (wenn keine Bezeichnung hinzugefügt ist). Sie können in derselben Zelle mehrere Berechnungen setzen. Sie werden im generierten Bericht gestapelt. Nachfolgende Grafik zeigt die Icons Sum, Durchschnitt und Min in einer Zeile Zwischensumme:



Beachten Sie, dass die Berechnungen als Code erscheinen, wenn Sie in der Zelle Textbezeichnungen einfügen (siehe unten). Fügen Sie mehrere Berechnungen hinzu, können Sie die Zelle formatieren, also z.B. Komma, Doppelpunkt oder Zeilenumbruch hinzufügen.



Platzhalter für Berechnung

Sie können Berechnungen mit folgenden Platzhaltern eingeben:

- ##S wird ersetzt durch die Summe in der Zeile Zwischensumme oder Gesamtsumme
- ##A wird ersetzt durch den **Durchschnitt**
- ##C wird ersetzt durch die **Anzahl**
- ##X wird ersetzt durch **Max**
- ##N wird ersetzt durch **Min**
- ##D wird ersetzt durch **Standardabweichung**
- ##xx, wobei xx eine Spaltennummer ist. Dieser Platzhalter wird ersetzt durch den Wert der Spalte mit der definierten Formatierung. Gibt es diese Spalte nicht, wird die Aktion nicht ausgeführt.

Diese Platzhalter sind hilfreich, wenn Sie in einer Zelle Bezeichnung und Werte mischen wollen.

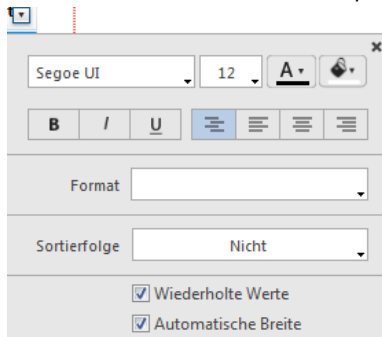
Wiederholte Werte für Umbruchspalten

In einem Bericht mit Zwischensummen heißen die Spalten, welche Datensätze in Gruppen für Berechnungen zusammenfassen, Umbruchspalten. Im nachfolgenden Bericht ist das Feld Firma eine Umbruchspalte. Hier werden die Datensätze nach Firma gruppiert.

Beim Drucken dieses Berichts werden die Werte in der Umbruchspalte nur einmal pro Umbruch gedruckt, d.h. der Name der Firma wird pro Gruppe nur jeweils für den ersten Datensatz gedruckt und nicht für jeden Datensatz wiederholt.

Firma	Artikel Name	MengenPreis
OSTERMAN, Heinz	Weißkohl	600
	Weizen	1450
	Maiz	6000
	Hafer	2000
	Weißkohl	1332
	Weizen	3219
	Maiz	13320
	Hafer	4440
Unterrhoff Brauerei	Weißkohl	12000
	Reis	20000
	Weißkohl	12000
WILLCOX Distribution	Maiz	1440
	Roggen	654
	Hopfen	20

In manchen Fällen wollen Sie jedoch die Werte in Umbruchspalten wiederholen, so dass sie für jeden Datensatz angezeigt werden. Dazu öffnen Sie das PopUp-Menü im Spaltentitel und markieren die Option **Wiederholte Werte**:



Im folgenden sehen Sie den o.a. Bericht mit **Wiederholten Werten**, d.h. der Name der Firma wird für jeden Datensatz angezeigt:

Firma	Artikel Name	MengenPreis
OSTERMAN, Heinz	Weißkohl	600
OSTERMAN, Heinz	Weizen	1450
OSTERMAN, Heinz	Maiz	6000
OSTERMAN, Heinz	Hafer	2000
OSTERMAN, Heinz	Weißkohl	1332
OSTERMAN, Heinz	Weizen	3219
OSTERMAN, Heinz	Maiz	13320
OSTERMAN, Heinz	Hafer	4440
OSTERMAN, Heinz	Weißkohl	12000
Unterrhoff Brauerei	Reis	20000
Unterrhoff Brauerei	Weißkohl	12000
Unterrhoff Brauerei	Maiz	1440
WILLCOX Distribution	Roggen	654
WILLCOX Distribution	Hopfen	20

Anzeigeformate für Spalte setzen

Sie können Anzeigeformate definieren für Spalten vom Typ Zahl, Ganzzahl, lange Ganzzahl, alphanumerisch, Datum, Zeit und Bild.

Zeigen Sie z.B. in einer Spalte Löhne an, können Sie dafür ein Zahlenformat festlegen. Das folgende Format setzt ein sechsstelliges Format mit 1 Kommastelle: ###,##0.00.

Es kann Beträge bis zu 999.999,99 anzeigen.

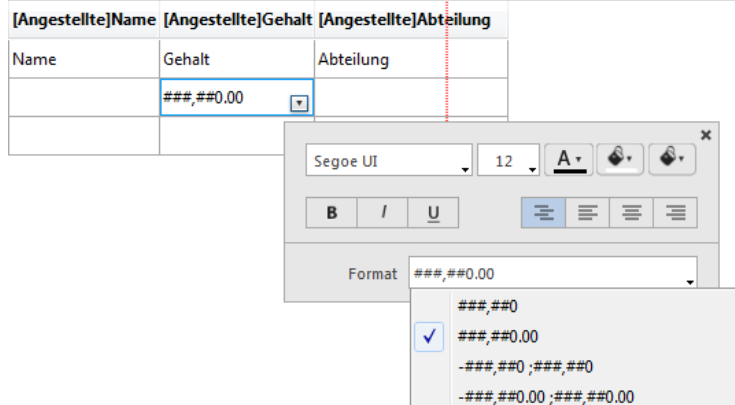
Enthält Ihr Bericht alphanumerische Felder wie Telefon- oder Sozialversicherungsnummer, können Sie ein alphanumerisches Format festlegen. Zeigt Ihr Bericht Datum, Uhrzeit oder Bilder, können Sie ebenfalls passende Anzeigeformate zuweisen

4D gibt verschiedene Standardformate vor. Sie können auch eigene Formate erstellen und dann für Schnellberichte verwenden. Weitere Informationen dazu finden Sie im Abschnitt [Anzeigeformate](#).

Um ein Format für ein Datenfeld vom Typ Zahl, alphanumerisch, Datum, Zeit oder Bild einzugeben:

1. Öffnen Sie das PopUp-Menü in der gewünschten Zelle.

Sie können eine einzelne Zelle formatieren oder den Kopfteil wählen, um alle Zellen in der Spalte zu formatieren.



2. Wählen Sie im Untermenü **Format** das gewünschte **Anzeigeformat**.

Das Untermenü zeigt je nach Datentyp der Spalte unterschiedliche Formate: Zahl, alphanumerisch, Datum, Zeit oder Bild. Enthält die Spalte nicht formatierbare Daten, erscheint der Befehl **Format** nicht im Kontextmenü.

Haben Sie auch eine Berechnung für die Spalte zugewiesen, gilt das Format der Zelle automatisch auch dafür. Davon ausgenommen ist das Format „Anzahl“. Es wird immer als Ganzzahl angezeigt und akzeptiert keine anderen Formatierungssymbole, wie z.B. das Euro-Zeichen.

Sie können auch eigene Formate eingeben. Doppelklicken Sie dazu in die Zelle und geben die Elemente des Formats über die Tastatur ein.

Sie können jeder Spalte im Bericht ein anderes Format zuweisen.

Zeilen oder Spalten ausblenden

Mit 4D können Sie im Schnellbericht Spalten bzw. Zeilen ein- und ausblenden. Zeilen ausblenden ist hilfreich, wenn der Bericht nur Berechnungen anzeigen soll. In diesem Fall blenden Sie den Datenteil aus, es erscheinen nur die Berechnungen in den Zeilen Gesamtsumme und Zwischensumme. Sie können ebenso die Zeile Umbruch oder Gesamtsumme ausblenden. Sie können eine Spalte ausblenden, wenn sie nur zur Sortierung dient und im Bericht nicht erscheinen soll.

Sie können eine Spalte/Zeile entweder über das Kontextmenü des Schnellberichteditors ausblenden

Um eine Zeile/Spalte auszublenden:

1. Klicken Sie mit der rechten Maustaste in die Titelzeile einer Zeile oder Spalte und dann im Kontextmenü jeweils auf den Eintrag **Diese Spalte ausblenden**, **Diese Zeile ausblenden** oder **Diese Umbruchzeile ausblenden**. 4D zeigt die Spalte/Zeile in grau an, d.h. sie erscheint nicht, wenn Sie den Schnellbericht auf dem Bildschirm anzeigen oder ausdrucken.

	[Customers]Country	[Products]Label	[Products]Price	[InvoiceLines]Quanti...
Title	Country		Price	Quantity
Format	(###) ###.####		\$###,###,##0.00	
[Products]Label changed				
[Customers]Country changed		Total quantity for #		
Grand total				

Zeile/Spalte einblenden

Ist eine Zeile/Spalte ausgeblendet, ist der jeweilige Eintrag **Diese Spalte ausblenden**, **Diese Zeile ausblenden** oder **Diese Umbruchzeile ausblenden** im Kontextmenü mit einem Häkchen gekennzeichnet.

Wählen Sie im Kontextmenü erneut den Eintrag **Diese Spalte ausblenden**, **Diese Zeile ausblenden** oder **Diese Umbruchzeile ausblenden**, wird die Zeile/Spalte in der Berichtvorlage wieder eingeblendet.

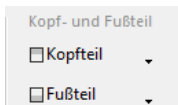
Kopf- und Fußteile hinzufügen

Sie können im Schnellbericht Kopf- und Fußteile hinzufügen. Sie werden dann beim Ausführen des Berichts eingefügt. Sie können folgendes definieren:

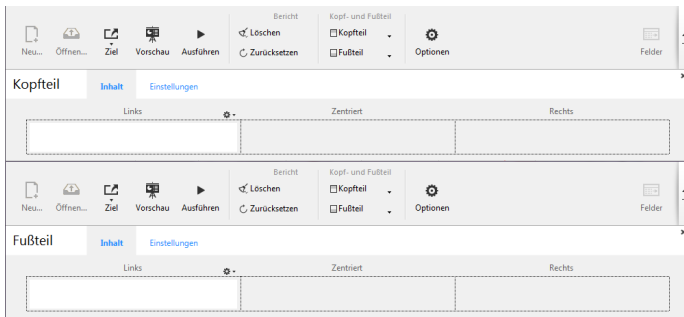
- Im Kopf- oder Fußteil Text oder ein Bild einfügen
- Die Höhe des Kopf- bzw. Fußteils definieren
- Eingaben im linken, mittleren oder rechten Teil setzen
- Für Text im Kopf- bzw. Fußteil Schriftart, -größe und -stil angeben
- Code eingeben, der im Bericht Seitennummer, Datum oder Uhrzeit hinzufügt

Kopf- und Fußteile können Sie nur für Berichte anlegen, die ausgedruckt werden. Wurden sie einmal angelegt, bleiben sie im Bericht erhalten, auch wenn sich das Ausgabeziel ändert. Weitere Informationen zu den verschiedenen Ausgabearten finden Sie im Abschnitt [Schnellbericht ausführen](#).

Um Kopf- und Fußteile hinzuzufügen, markieren Sie in der Werkzeugleiste des Schnellberichteditors die Option **Kopfteil** oder **Fußteil**.



Auf dem Bildschirm erscheint das jeweilige Dialogfenster mit verschiedenen Optionen, die für Kopfteil oder Fußteil gleich sind:



Inhalt definieren

Klicken Sie auf die Option **Inhalt**, erscheinen drei Eingabebereiche.

In den Bereichen **Links**, **Zentriert** und **Rechts** können Sie Informationen zum Bericht einfügen. Klicken Sie in einen Bereich, erscheint darüber ein Icon zum Öffnen des lokalen Menüs. Hier können Sie Variablen für die Seitennummer und für Datum bzw. Zeit der Ausführung des Berichts setzen:

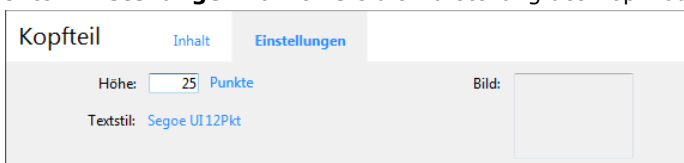


Sie können den Code für die Variablen auch direkt eingeben:

- #H für die Zeit der Erstellung
- #D für Datum der Erstellung
- #P für Seitennummer

Die Einstellungen definieren

Unter **Einstellungen** können Sie die Darstellung des Kopf- oder Fußteils festlegen:



Sie können für jeden Bereich folgendes definieren:

- Höhe
- Schriftart und -stil
- Bild

Höhe

Im Bereich **Höhe** geben Sie die gewünschte Höhe für den Kopf- oder Fußteil ein. Standardmäßig wird als Höhe 25 Punkte angegeben. Sie können sowohl den Wert als auch die Einheit ändern. Klicken Sie auf die verwendete Einheit, können Sie zwischen verschiedenen Optionen wählen.

Höhe:

Textstil

Klicken Sie auf die vorgegebene Schrift, öffnet sich die Liste der verfügbaren Schriften. Um die Schriftart, -größe oder -farbe zu definieren, klicken Sie am Ende der Liste auf den Eintrag **Schrift anzeigen**. Auf dem Bildschirm erscheint das standardmäßige Dialogfenster des Systems.

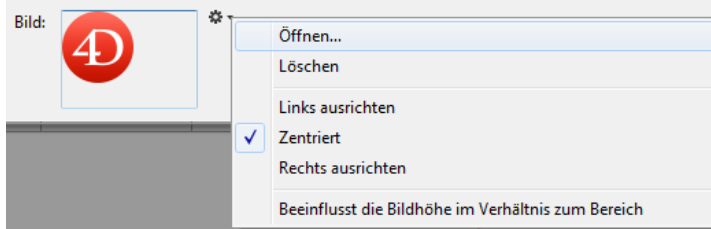
Bild

In diesem Bereich können Sie ein Bild einfügen:



Es gibt zwei Möglichkeiten:

- Sie ziehen ein Bild oder eine Bilddatei per Drag-and-Drop in den Bereich
- Sie öffnen das lokale Menü rechts neben dem Eingabebereich und wählen den Eintrag **Öffnen**:



Das Bild wird standardmäßig in der Mitte des Kopfteils bzw. Fußteils eingesetzt. Über das lokale Menü können Sie die Ausrichtung verändern (**Links ausrichten**, **Zentriert**, **Rechts ausrichten**):

Soll das Bild automatisch die Höhe des Kopf- bzw. Fußteils haben, wählen Sie den Eintrag **Beeinflusst die Bildhöhe im Verhältnis zum Bereich**:



Schnellbericht ausführen

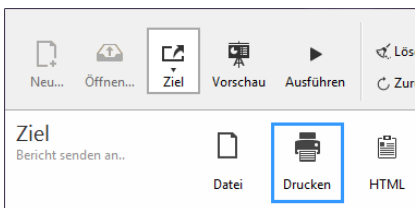
Haben Sie das Layout für Ihre Berichtsvorlage fertiggestellt, können Sie den Schnellbericht „drucken“. Dafür gibt es verschiedene Möglichkeiten:

- In ein Textdokument auf der Festplatte
- Auf dem gewählten Standarddrucker
- In ein HTML Dokument

Um ein Ausgabeziel zu definieren, klicken Sie in der Werkzeugleiste des Berichteditors auf die Schaltfläche **Ziel**:



Auf dem Bildschirm erscheint das Auswahlfenster mit den Optionen **Datei**, **Drucken** und **HTML**:



Über die Schaltfläche **Optionen** rufen Sie die verfügbaren Optionen auf. Sie sind je nach dem gewählten Ausgabedokument unterschiedlich.

Beim Ziel **Drucken** können Sie auch die Schaltfläche **Vorschau** verwenden, um sich erst die Vorschau des gestalteten Bericht anzusehen.



Datei

Diese Option sendet Ihren Schnellbericht an die Festplatte. Sie können ihn mit anderen Anwendungen öffnen und verändern, z.B. Texteditoren und Tabellenkalkulation. Die Datensätze des Schnellberichts werden in eine Textdatei exportiert.

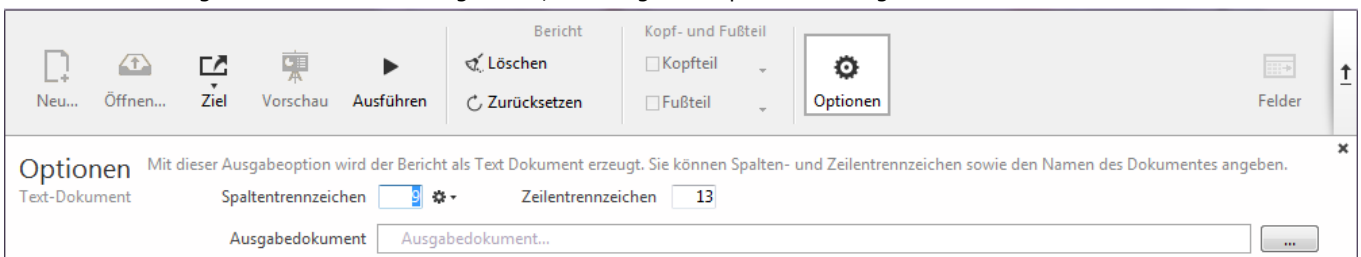
4D verwendet die Spaltentitel automatisch als den ersten exportierten "Datensatz".

Klicken Sie auf die Schaltfläche **Ausführen**, erscheint ein Standard Sichern-Dialog. Haben Sie Name und Speicherort auf der Seite **Optionen** angegeben (siehe unten), werden sie standardmäßig vorgeschlagen. Geben Sie einen Dateinamen ein und klicken auf **Speichern**. Ist der Bericht in eine Datei übertragen, kehrt 4D zum Schnellberichteditor zurück. Denken Sie daran, das Ausgabeziel zu wechseln, wenn Sie den Schnellbericht zum Drucker senden wollen.

Hinweis: Für diese Option ist keine Vorschau verfügbar.

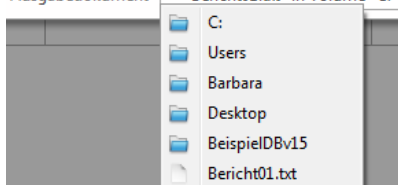
Optionen für Textdatei

Haben Sie als Ausgabedokument "Datei" gewählt, sind folgende Optionen verfügbar:



- **Spaltentrennzeichen** und **Zeilentrennzeichen**: Hier können Sie die von 4D vorgegebenen Trenner für Spalten und Zeilen in der erzeugten Textdatei verändern.
4D verwendet standardmäßig das Zeichen 9 (Tab) als Spaltentrenner und 13 (Zeilenschaltung) für Zeilenumbruch.
- **Ausgabedokument**: Hier setzen Sie Name und Ziel für die erzeugte Datei fest. Diese Angaben werden beim Ausführen des Berichts standardmäßig vorgegeben.
Ist der Speicherort gesetzt, können Sie auf den Pfeil im Eingabebereich klicken und den Pfad nachverfolgen:

Ausgabedokument ▾ "Bericht01.txt" in Volume "C:"



Beachten Sie, dass diese Option keine aktuelle Datei generiert; sie sichert nur den Pfad.

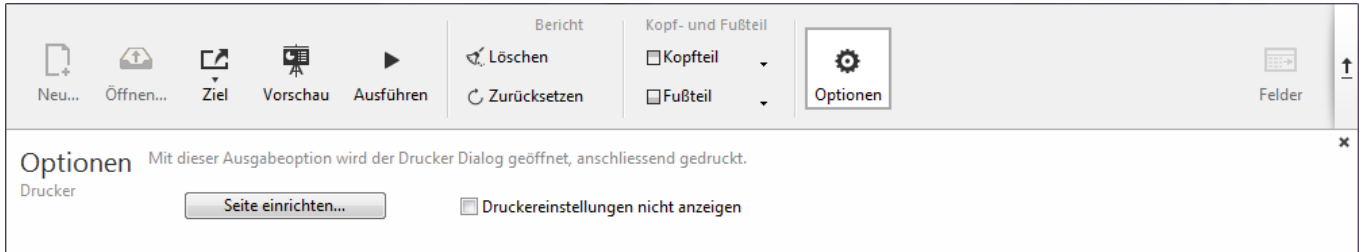
Drucken

Diese Option verwendet den Drucker, den Sie in Ihren Druckereinstellungen gewählt haben. Vor dem Ausdruck können Sie eine Vorschau auf dem Bildschirm ansehen.

Klicken Sie in der Werkzeugleiste des Editors auf die Schaltfläche **Ausführen**, zeigt 4D den Standard Drucken-Dialog ihres Betriebssystems. Wollen Sie diesen Dialog ausblenden, markieren Sie auf der Seite **Optionen** die Option **Druckereinstellungen nicht anzeigen** (siehe unten).

Optionen für Drucken

Ist als Ausgabeziel "Drucken" gesetzt, sind folgende Optionen verfügbar:



- **Seite einrichten...:** Zeigt die Druckereinstellungen Ihres Betriebssystems. Hier können Sie Papiergröße, Ausrichtung und Druckränder festlegen.
- **Druckereinstellungen nicht anzeigen:** Ist diese Option markiert, wird der Bericht direkt auf dem aktuellen Drucker gedruckt, wenn Sie auf die Schaltfläche **Ausführen** klicken. Es werden keine Druckereinstellungen angezeigt.

HTML Dokument

Diese Option sendet Ihren Schnellbericht an eine HTML Datei. Das ist die standardmäßige HTML Vorlage (template), außer sie wurde per Programmierung geändert. Sie können auch eigene Vorlagen einrichten und über die Seite Optionen laden (siehe unten)

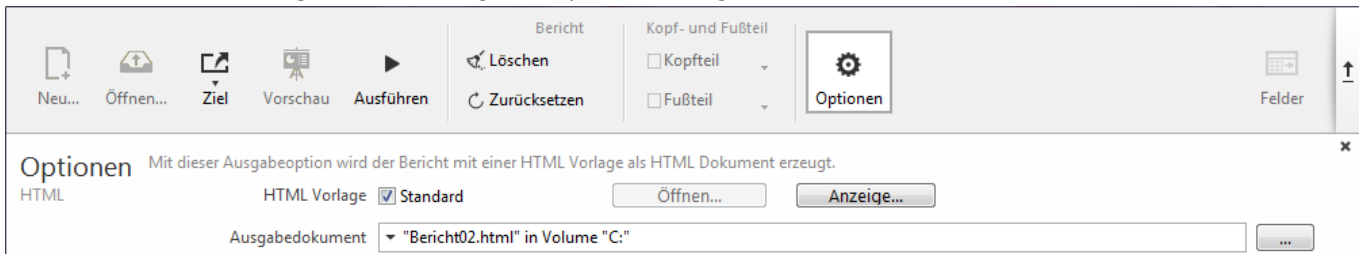
4D verwendet die Spaltentitel automatisch als ersten exportierten "Datensatz".

Klicken Sie auf die Schaltfläche **Ausführen**, erscheint ein Standard Sicherungs-Dialog. Haben Sie Name und Speicherort auf der Seite **Optionen** angegeben (siehe unten), werden sie standardmäßig vorgegeben. Tragen Sie einen Dateinamen ein und klicken auf **Speichern**. Ist der Bericht in eine HTML Datei übertragen, kehrt 4D zum Schnellberichteditor zurück. Denken Sie daran, das Ausgabeziel zu wechseln, wenn Sie den Schnellbericht zum Drucker senden wollen.

Der Zeichensatz zum Generieren von HTML Code wird in den Datenbank-Eigenschaften über den Parameter "Standardsatz" definiert (siehe Abschnitt [Seite Web/ Optionen \(II\)](#)).

Optionen für HTML

Haben Sie als Ziel "HTML" gewählt, sind folgende Optionen verfügbar:




- **HTML Vorlage:** Setzt die HTML Vorlage zum Generieren des Berichts fest. Sie können entweder:
 - die Standardvorlage verwenden (Option **Standard**) oder
 - über die Schaltfläche **Öffnen...** eine eigene Vorlage verwenden.


Über die Schaltfläche **Anzeige...** können Sie sich den Code der gewählten Vorlage in einem separaten Fenster ansehen.


Weitere Informationen zu eigenen HTML Vorlagen finden Sie unter den Befehlen **QR SET HTML TEMPLATE** und **QR Get HTML template**.

- **Ausgabedokument:** Hier setzen Sie Name und Ziel für die erzeugte Datei fest. Diese Angaben werden beim Ausführen des Berichts standardmäßig vorgegeben. Ist der Speicherort gesetzt, können Sie auf den Pfeil im Eingabebereich klicken und den Pfad nachverfolgen. Beachten Sie, dass diese Option keine aktuelle Datei generiert; sie sichert nur den Pfad.

Etiketteneditor

 Beschreibung des Etiketteneditors

 Etikett erstellen

 Etikettenbogen einrichten

Beschreibung des Etiketteneditors

Der Etiketteneditor von 4D ist das ideale Werkzeug zum Gestalten und Drucken einer Vielzahl verschiedener Etiketten. Sie können folgendes durchführen:

- Etiketten für Briefe, Aktenordner und Aktendeckel und für viele andere Anwendungen gestalten
- Schrift, Schriftgröße und Stil für die Etiketten festlegen
- Dekorative Elemente in Etikettenvorlagen erstellen und einfügen
- Anzahl der Etiketten nebeneinander und untereinander pro Seite festlegen
- Mehrere Etiketten pro Datensatz drucken
- Seitenränder der Etiketten festlegen
- Eine Methode festlegen, die beim Drucken eines Etiketts oder Datensatzes ausgeführt wird,
- Etikettenvorlagen laden und speichern
- Etiketten drucken

Etiketten lassen sich auch mit dem Formulareditor erstellen. Damit können Sie spezielle Etiketten gestalten, die Variablen enthalten oder die Optionen der Zeichenwerkzeuge im Formulareditor nutzen. Weitere Informationen dazu finden Sie im Abschnitt **Etiketten erstellen**.

Mit dem Etiketteneditor erstellen, formatieren und drucken Sie Etiketten. Er enthält Einstellungen zum Anlegen von Etiketten und das Positionieren der Etiketten auf Etikettenbögen. Sie möchten zum Beispiel für Adressetiketten eine Vorlage erzeugen mit dem Vornamen und Nachnamen der Person in der ersten Zeile, der Straße in der zweiten usw. In der Vorlage können Sie die Anzahl der Etiketten auf einer Seite und die Ränder des Etikettenpapiers festlegen, so dass der Etikettentext auf den Etiketten zentriert wird.

Haben Sie ein passendes Layout erstellt, können Sie es auch auf der Festplatte speichern und später wieder verwenden.

Um den Etiketteneditor zu öffnen:

Wählen Sie in der Designumgebung im Menü **Bericht** oder in der Werkzeugleiste über das Icon **Tools** den Befehl

Etiketten

ODER

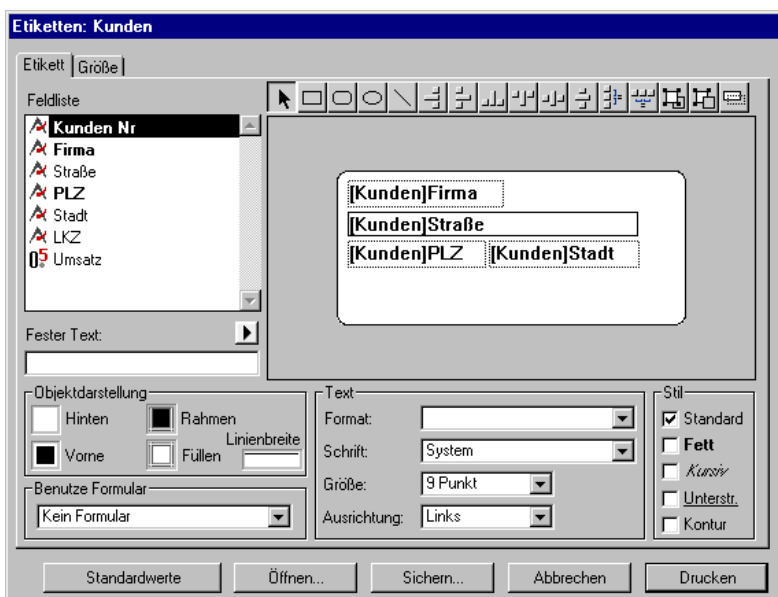
Wählen Sie in der Anwendungsumgebung den 4D Befehl **PRINT LABEL**.

Der Etiketteneditor besteht aus den beiden Seiten **Etikett** und **Größe** in Form von Registerkarten. Auf der Seite **Etiketten** legen Sie den Inhalt des Etiketts fest, auf der Seite **Größe** die Position der Etiketten innerhalb der Seite.

- **Etikett**: Diese Seite enthält Einstellungen zum Gestalten und Formatieren von Etiketten. Sie können auch ein Formular bestimmen, über das Sie Ihre Etiketten aufbauen können.
- **Größe**: Diese Seite enthält Optionen zum Anordnen der Etiketten auf der Seite und zum Drucken. Das können Sie selbst ausführen oder das Programm berechnen lassen, wieviel Etiketten auf die Seite passen. Sie können der Seite auch eine Methode zuordnen.

Seite **Etikett**

Diese Seite **Etikett** enthält Einstellungen zum Gestalten und Formatieren von Etiketten. Sie besteht aus folgenden Elementen:



- **Feldliste**: Dieser Bereich zeigt die Namen der Datenfelder in der aktuellen Tabelle in hierarchischer Reihenfolge. Wenn diese Tabelle mit anderen Tabellen verknüpft ist, sind die verknüpften Datenfelder unter Windows mit einem Pluszeichen, auf Macintosh mit einem Pfeil markiert. Die Datenfelder in der verknüpften Tabelle erscheinen eingerückt.

Um ein Feld aus dieser Liste für das Etikett zu verwenden, ziehen Sie es einfach in den Vorschaubereich.

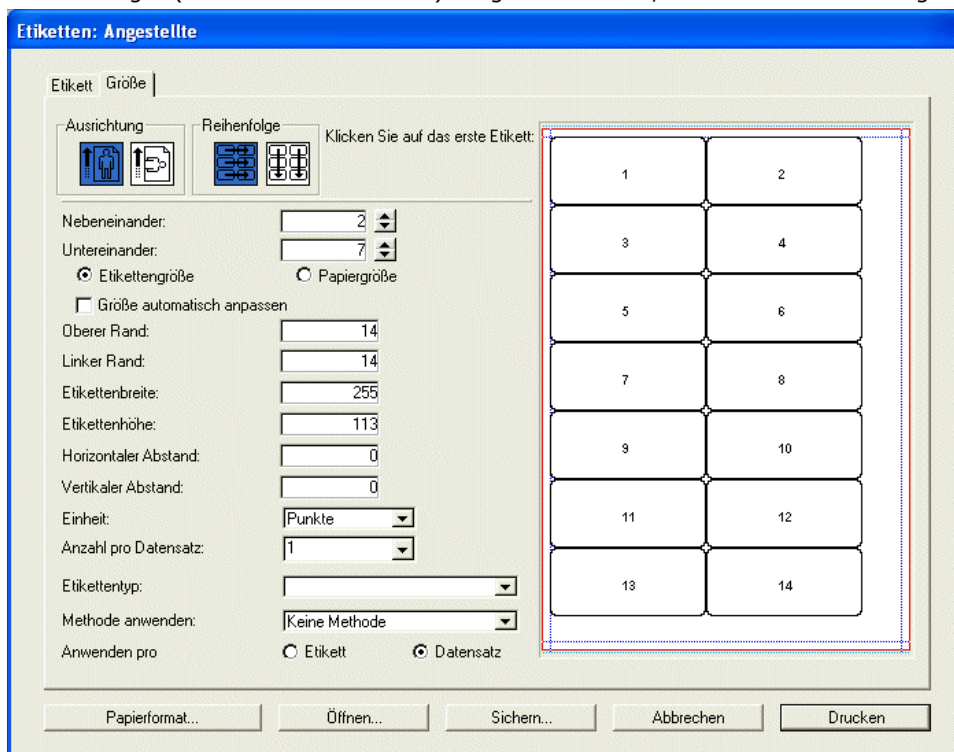
Hinweise:

- Im Etiketteneditor erscheinen nur sichtbare Tabellen und Datenfelder. Weitere Informationen zum Ausblenden von Tabellen und Datenfeldern finden Sie im Abschnitt "Unsichtbar" unter **Tabelleneigenschaften** und **Feldeigenschaften**
- Der Etiketteneditor unterstützt keine Datenfelder vom Typ Objekt
- **Fester Text:** Hier können Sie statischen Text für das Etikett eingeben. Klicken Sie auf den Pfeil über dem Eingabebereich, wird der Text in der Vorschau eingefügt. Mit der Maus können Sie ihn an die passende Stelle setzen.
- **Etikettenvorschau:** Hier gestalten Sie Ihr Etikett. Sie setzen alle Einträge für Ihr Etikett an die passende Stelle. Das weiße Rechteck stellt ein einzelnes Etikett dar. Seine Größe wird auf der Seite "Größe" festgelegt. Sie können Felder in die Vorschau ziehen oder über den Bereich **Fester Text** statischen Text eingeben. Sie können auch zwei Felder zusammensetzen, z.B. Vor- und Nachname. Setzen Sie dazu das zweite Feld neben das erste. Die Felder werden automatisch durch einen Abstand voneinander getrennt. Über die Werkzeugleiste über der Vorschau können Sie Objekte zeichnen, auswählen, ausrichten, verteilen, übereinanderlegen oder duplizieren. Sie können auch Einträge aus der Zwischenablage übernehmen.
- **Objektdarstellung, Text und Stil:** Diese drei Bereiche enthalten Optionen, um die Darstellung der Einträge in der Vorschau zu konfigurieren. Im Bereich "Objektdarstellung" legen Sie Vordergrund- und Hintergrundfarben, Füllmuster und Rahmen für einzelne Objekte auf dem Etikett fest. In den Bereichen "Text" und "Stil" legen Sie Schrift, Schriftgröße, Formatierung und Textstil fest. Sie können diese Parameter auf Felder oder statischen Text anwenden.
- **Benutze Formular:** Mit dieser DropDown Liste können Sie den Etiketteneditor umgehen und ein Formular zum Drucken von Etiketten verwenden. Standardmäßig ist die Option **Kein Formular** vorgegeben, d.h. die Vorgaben des Etiketteneditors werden verwendet. Wenn Sie ein Formular verwenden wollen, wählen Sie dieses aus der Liste. 4D ignoriert dann die Etiketteneigenschaften des Etiketteneditors und druckt die Etiketten nach der Vorlage bzw. dem festgesetzten Formular. Wie bei anderen Druckaufträgen führt 4D alle mit dem Formular verknüpften Formular- oder Objektmethoden aus. Weitere Informationen dazu finden Sie im Abschnitt **Über den Etiketteneditor**.
- **Standardwerte:** Diese Schaltfläche wendet die Standarddarstellung auf die gewählten Objekte an. Alle im Bereich "Objekt" definierten Parameter werden auf ihre Standardwerte zurückgesetzt.

Hinweis zur Kompatibilität: Der Etiketteneditor unterstützt nur ASCII Zeichen. Verwenden Sie einen erweiterten Zeichensatz, müssen Sie Etiketten über das aktuelle Ausgabeformular drucken.

Seite Größe

Die Seite **Größe** enthält Optionen zum Drucken von Etiketten gemäß den Einstellungen des Druckers, den Sie im Druckmanager (Auswahl auf Macintosh) ausgewählt haben, so dass die Seite richtig formatiert wird.



- **Papierformat:** Damit legen Sie das Seitenformat zum Drucken der Etiketten fest. Klicken Sie auf diese Schaltfläche, erscheint das Dialogfenster des Druckers, der in Ihrem Betriebssystem eingerichtet ist.
Hinweis: Das vom Editor angelegte Blatt basiert auf der logischen Seite des Druckers, d.h. die tatsächliche Seite minus der nicht bedruckbaren Ränder. Diese Ränder erscheinen in der Vorschau als gestrichelte Linien.
- **Ausrichtung und Reihenfolge:** Damit legen Sie die Ausrichtung und die Reihenfolge fest, in der Informationen den Etiketten zugeordnet werden. Beachten Sie, dass die Layoutausrichtung von der Seitenausrichtung unabhängig ist. Ändern Sie die Ausrichtung des Layout, müssen Sie auch im Druckdialog die dazu passende Ausrichtung der Seite wählen

- Eingabefelder **Nebeneinander und Untereinander**: Damit legen Sie die Etikettenanzahl für Ihren Etikettenbogen fest. Ist **Größe automatisch anpassen** markiert, wird die Größe eines Etiketts dem Etikettenbogen angepasst.
- **Bereich zum Setzen der Etikettengröße**: Dieser Bereich enthält verschiedene Optionen, um Etikettengröße und Ränder aufeinander abzustimmen.
 - **Einheit**: In dieser Dropdown-Liste legen Sie die Maßeinheit für Ihr Etikett bzw. Ihre Etikettenseiten fest. Sie können Punkte, Millimeter, Zentimeter oder Zoll angeben
 - **Etikettengröße und Papiergröße**: Über diese Optionsfelder legen Sie das Etikett bzw. die Seite zur Bestimmung der Etiketten- und Seitenabmessungen fest. Markieren Sie Etikettengröße, können Sie die Etikettenbreite und Etikettenhöhe in den entsprechenden Bereichen eingeben. Markieren Sie Seitengröße, können Sie Werte für rechten Rand und unteren Rand eingeben, — außer die Option "Größe automatisch anpassen" ist markiert (siehe nächster Absatz). Klicken Sie auf **Papiergröße**, können Sie Werte für **Rechten** und **Unteren Rand** eingeben. Diese Ränder erscheinen in der Vorschau als blaue Linien.
 - Markieren Sie die Option **Größe automatisch anpassen**, berechnet 4D die Etikettengröße automatisch, z.B. die Parameter **Etikettenbreite** und **Etikettenhöhe**, gemäß den in allen anderen Parametern gesetzten Werten. Ist diese Option markiert, wird die Etikettengröße bei jeder Änderung eines Seitenparameters angepasst. Die Parameter **Etikettenbreite** und **Etikettenhöhe** lassen sich dann nicht mehr manuell setzen.
 - **Oberer Rand** und **Unterer Rand**: Damit setzen Sie die entsprechenden Ränder für das Etikettenblatt. Sie erscheinen in der Vorschau als blaue Linien.
 - **Horizontaler Abstand** und **Vertikaler Abstand**: Diese Parameter bestimmt den Abstand zwischen den Etikettenspalten und -reihen.
- **Anzahl pro Datensatz**: Über diese DropDown Liste können Sie jedes Etikett mehrfach ausdrucken. 4D druckt dann die Kopien hintereinander, statt Kopien der Etikettenseiten auszudrucken.
- **Etikettentyp**: Über diese DropDown Liste legen Sie die Etiketten-, Seitenabmessungen und Ränder durch Wählen eines kommerziell erhältlichen Standardetikettenbogens fest.
- **Methode anwenden** und **Anwenden pro**: Damit legen Sie fest, dass beim Drucken eine bestimmte Methode ausgelöst wird. Sie können z.B. eine Methode anwenden, die Druckdatum und -zeit für jedes Etikett ausgibt. Das DropDown Menü enthält alle Projektmethoden der Datenbank. Die Optionsfelder **Anwenden pro** bestimmen, ob die Methode pro Etikett oder pro Datensatz angewendet wird. Diese Optionsfelder haben nur eine Auswirkung, wenn Sie mehr als ein Exemplar von jedem Etikett drucken und gleichzeitig beim Drucken eine Methode anwenden.
- **Vorschau Layout**: Dieser Bereich stellt die verkleinerte Ansicht einer komplette Etikettenseite dar, und zwar gemäß der eingegebenen Werte. Die Seitenvorschau zeigt auch die Papiergröße an, die im Dialogfenster Druckeinstellung festgelegt wird. Sie können auch das erste Etikett auf der Seite auswählen, das gedruckt werden soll. Der rote Rahmen markiert die Größe der tatsächlichen Seite, der blaue Rahmen die Größe des druckbaren Bereichs.

Etikettenvorlage sichern und laden

4D ermöglicht, eine erstellte Etikettenvorlage als Datei zu sichern, die Sie im Etiketteneditor öffnen können. Durch das Sichern von Vorlagen können Sie eine Etikettenbibliothek einrichten, die Sie je nach Bedarf einsetzen können. Vorlagen sichern die Parameter der Seiten Etikett und Größe.

- Um eine Etikettenvorlage zu sichern, klicken Sie am unteren Rand auf die Schaltfläche **Speichern** und geben Sie einen Namen und einen Speicherort für die Vorlage ein.
- Um eine Etikettenvorlage zu laden, klicken Sie auf die Schaltfläche **Öffnen** und wählen im Standard Öffnen-Dialog die gewünschte Vorlage. 4D ersetzt das aktuelle Layout durch das der gewählten Vorlage.

Etikettendateien erhalten die Endung .4LB.

Etikett erstellen

Dieser Abschnitt beschreibt die Möglichkeiten, die der Etiketteneditor zum Erstellen eigener Etiketten bietet. Sie können sowohl ein individuelles Etikett als auch ein Etikettenblatt anlegen. In welcher Reihenfolge Sie vorgehen, richtet sich nach Ihren Prioritäten:

- Müssen Ihre Etiketten eine bestimmte Größe haben, legen Sie zuerst die Größe auf der Seite "Größe" an und gestalten dann den Inhalt auf der Seite "Etikett".
- Müssen Ihre Etiketten ein bestimmtes Aussehen mit spezifischem Inhalt haben, beginnen Sie auf der Seite "Etikett" und passen dann die Größe auf der Seite "Größe" an.

Hinweis: Damit alle oben beschriebenen Funktionen verfügbar sind, müssen Sie auf der Seite "Etikett" unter **Benutze Formular** die Option **Kein Formular** gewählt haben. Weitere Informationen dazu finden Sie auf der [Seite Etikett](#).

Sie können das Format eines Etiketten-Layouts in der Vorschau verändern. Gehen Sie dazu auf die Seite "Größe". Weitere Informationen dazu finden Sie im Abschnitt [Etikettenbogen einrichten](#).

Felder einfügen

Um Felder in Ihr Etiketten-Layout einzufügen:

1. Gehen Sie zur Seite "Etikett".
2. Ziehen Sie das gewünschte Datenfeld aus der Feldliste in den Bereich Etikettenvorschau.
Ist das Datenfeld in einer verknüpften Tabelle, erweitern Sie das verknüpfte Datenfeld, damit die Datenfelder der verknüpften Tabelle erscheinen. Ihr Datenfeld wird in die Vorschau gesetzt. Auswahlgriffe zeigen an, dass es ausgewählt ist.
3. Ziehen Sie weitere Datenfelder in die Vorschau.

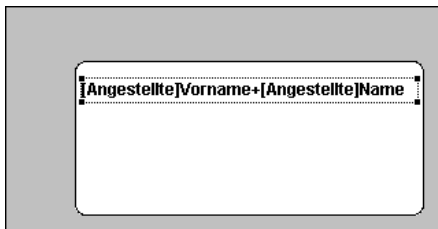
Felder zusammenfügen

Beim Erstellen von Etiketten können Sie auch mehrere Felder miteinander verketteten. In einem Adressetikett können Sie z.B. Vor- und Nachnamen in eine Zeile setzen. Sie können Felder auch vertikal zusammensetzen.

Beim Zusammenfügen von Feldern setzt 4D automatisch einen Abstand dazwischen.

Um mehrere Felder zusammenzufügen:

1. Ziehen Sie das erste Datenfeld in den Bereich Etikettenvorschau.
2. Ziehen Sie ein zweites Datenfeld auf das erste.
ODER
ziehen Sie das neue Datenfeld bei gedrückter **Umschalttaste** auf das erste Datenfeld.
 - Verketteten Sie zwei Felder per Drag-and-Drop, wird der Separator verwendet, den Sie in den Ressourcen von 4D definiert haben. Das ist standardmäßig das Leerzeichen. Ein Pluszeichen (+) zwischen den Datenfeldnamen zeigt an, dass die Datenfelder in einer Zeile verkettet sind. Wenn 4D das Etikett druckt, wird ein Leerraum zwischen den Datenfeldern in einer Zeile eingefügt.
 - Verketteten Sie zwei Felder per Drag-and-Drop bei gedrückter **Umschalttaste**, wird die Zeilenschaltung als Separator verwendet. Ist beim Drucken der Etiketten das zweite Feld leer, wird diese Position gelöscht und durch das nächste Feld ersetzt. Enthält ein Feld Text mit Zeilenschaltung, wird dies berücksichtigt. So können Sie Adressetiketten mit mehreren übereinanderliegenden Feldern erstellen ([Kunden]Adresse1, [Kunden]Adresse2, etc.) und müssen keine Leerzeile einrichten, wenn eine Adresse nur ein ausgefülltes Feld enthält.Sie können eine Gruppe verketteter Felder automatisch anpassen, so dass diese der enthaltenen Anzahl Zeilen entspricht. Klicken Sie dazu unter Windows mit der **Strg-Taste+Mausklick**, auf Macintosh mit der **Befehlstaste+Mausklick** auf den rechten unteren Griff des Gruppenrahmens.



Um ein verkettetes Feld zu entfernen:

1. Wählen Sie die Zeile mit einem oder mehreren verketteten Feldern.
2. Drücken Sie die **Rückschritttaste**.
Das letzte Feld in der Zeile wird entfernt.


Grafische Elemente einfügen

Im Etiketteneditor können Sie grafische Elemente erzeugen und einfügen, um die Darstellung zu verbessern oder zu jedem Datensatz Zusatzinformationen einzufügen. Sie können folgende Elemente hinzufügen:

- statischen Text
- Linien, Kreise oder andere geometrische Formen, die im Editor gezeichnet werden
- Bilder aus der Zwischenablage

Festen Text einfügen

Sie können festen Text in Ihr Etiketten-Layout einfügen, z.B. um Felder mit Bezeichnungen versehen:

1. Geben Sie einen Text im Bereich Fester Text ein.
2. Klicken Sie auf den Pfeil  über dem Bereich.
Das Objekt mit festem Text wird in der unteren linken Ecke in die Vorschau eingefügt.
3. Sie können dieses Objekt dann durch Ziehen mit gedrückter Maustaste verschieben und mit anderen Objekten ausrichten.

Linien und geometrische Formen einfügen

Die Vorschau enthält eine Werkzeugleiste, über die Sie geometrische Formen erstellen können. Sie können z.B. einen Rahmen für den Inhalt Ihres Etiketts anlegen oder Rechtecke, Ovale und Linien einsetzen.


Um ein grafisches Element einzufügen, klicken Sie auf das gewünschte Symbol und zeichnen es in der Vorschau. Mit gedrückter **Umschalttaste** zeichnen Sie Elemente mit Einschränkung, also Kreise, Rechtecke oder gerade Linien.

Hinweis: Wollen Sie einen Rahmen um ein Feld im Etikett zeichnen, können Sie folgende Tastenkombination verwenden: Wählen Sie das Feld und drücken unter Windows **Strg+1**, auf Mac OS **Befehl+1**. Der Rahmen wird mit 1 Pixel Abstand vom Feld erstellt. Drücken Sie **Strg+2** bzw. **Befehl+2**, wird der Rahmen mit 2 Pixel Abstand erstellt, usw..

Bild übertragen

Sie können auch ein Bild aus der Zwischenablage in das Etikett einfügen, z.B. um ein Logo hinzuzufügen. Wählen Sie dazu die Vorschau aus und übertragen das Bild mit der Tastenkombination **Strg+v** unter Windows, auf Macintosh **Befehl+v** auf Macintosh.

Objekte duplizieren

Über die Schaltfläche  können Sie das bzw. die ausgewählten Objekte duplizieren. Sie können Felder und grafische Elemente duplizieren.

Objekte auswählen

Wollen Sie Stilarten oder Änderungen auf ein oder mehrere Objekte in Ihrem Etikett anwenden, müssen Sie es zuerst auswählen. Es gibt verschiedene Möglichkeiten zum Auswählen von Feldern oder grafischen Elementen.

- Um nicht-zusammenhängende Objekte zu wählen, klicken Sie auf das erste Objekt und wählen weitere Objekte mit gedrückter **Umschalttaste** aus.
- Um zusammenhängende Objekte zu wählen, klicken Sie in die Vorschau und ziehen Sie den Cursor, um einen Auswahlrahmen zu zeichnen. Alle von diesem Rechteck berührten Objekte werden ausgewählt. Zeichnen Sie den Rahmen mit gedrückter **Alt**-Taste unter Windows, bzw. **Wahltaste** auf Mac OS, werden nur die Objekte gewählt, die innerhalb des Rahmens liegen.

Objektdarstellung ändern

Haben Sie Elemente in die Vorschau gesetzt, können Sie ihre Darstellung verändern. Sie können sowohl Felder als auch grafische Elemente in der Größe ändern, bewegen, ausrichten, verteilen oder übereinanderlegen.

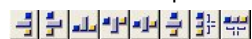
Objekte bewegen

Sie können Objekte bewegen, indem Sie sie auswählen und mit gedrückter Maustaste ziehen. Sie können auch die Pfeiltasten der Tastatur verwenden:

- Um ein Objekt jeweils um ein Pixel zu verschieben, wählen Sie das Objekt aus und drücken die Pfeiltaste für die gewünschte Richtung.
- Um ein Objekt jeweils um zehn Pixel zu verschieben, wählen Sie das Objekt unter Windows mit gedrückter **Strg-Taste**, auf Mac OS mit gedrückter **Befehlstaste** aus und drücken die Pfeiltaste für die gewünschte Richtung.

Objekte ausrichten

Mit den Ausrichtungssymbolen können Sie Objekte untereinander ausrichten. Wenn Sie Objekte ausrichten, können Sie diese zum entsprechenden Objekt nach oben, nach unten, zur Seite oder horizontal oder vertikal mittig ausrichten.



Die Werkzeugleiste bietet zwei Icons, über die Sie drei oder mehr Objekte in gleichmäßigen Abständen ausrichten und verteilen können.

Standardmäßig ist ein konstanter Abstand gesetzt, der auf den Rändern der gewählten Objekte basiert. Sie können Objekte auch gemäß dem oberen bzw. unteren Rand oder ihrer Mitte verteilen. Sehen Sie hierzu nachfolgendes Schema:

Aktion	Klicken	Umschaltt.+Klick Wahl.+Klick(Mac OS)	Alt+Klick(Windows)	Umschaltt.+Alt+Klick(Windows) Umschaltt.+Wahl+Klick(Mac OS)
Vertikal verteilen	Standard 	Oberkanten 	Unterkanten 	Mitten
Horizontal verteilen	Standard 	Linke Kanten 	Rechte Kanten 	Mitten

Objekte übereinanderlegen

Sie können eine Vorlage erstellen, die Objekte in verschiedenen Ebenen verwendet. Sie können z.B. im Etikett ein schattiertes Rechteck hinter den Datenfeldern einfügen. Dazu verwenden Sie die Icon **Hinten** und **Vorne** in der Werkzeugleiste. Die folgende Abbildung zeigt Objekte vor und hinter anderen Objekten:



Inhalt der Objekte ändern

Sie können den Inhalt von Objekten in der Vorschau verändern. Sie können z.B. die Farbe oder das Muster von Objekten ändern oder Schriftart und -größe von Textobjekten.

Hinweis: Der Inhalt von Bildern, die aus der Zwischenablage übernommen wurden, lässt sich nicht verändern. Sie können also z.B. nicht die Farbe verändern. Nur Elemente, die direkt im Editor erstellt wurden, lassen sich verändern.

Objekte in der Größe ändern

Sie können jedes Objekt im Etikett in der Größe ändern, indem Sie es auswählen und mit gedrückter Maustaste einen Auswahlgriff ziehen:

- **Alt-Taste** unter Windows, Wahl taste auf Mac OS + Pfeiltasten der Tastatur, um ein Objekt in 1-Pixel Schritten zu verkleinern oder zu vergrößern.
- **Alt+Strg** unter Windows, **Wahl- + Befehlstaste** auf Mac OS + Pfeiltasten der Tastatur, um ein Objekt in 10-Pixel Schritten zu verkleinern oder zu vergrößern.

Objektdarstellung ändern

Über die Funktionen im Bereich "Objektdarstellung" legen Sie Vordergrund- und Hintergrundfarben, Füllmuster und Rahmen für einzelne Objekte auf dem Etikett fest:

- Vorder- und Hintergrundfarben für ausgewählte Objekte setzen Sie über die Schaltflächengitter **Hinten** und **Vorne**. Ist das Objekt ein Datenfeld oder statischer Text, bestimmt die Vordergrundfarbe die Farbe des Textes, die Hintergrundfarbe die Farbe des Objektrechtecks. Haben Sie ein Objekt ausgewählt, das im Editor gezeichnet wurde, wird die Vordergrundfarbe auf die Ränder sowie Muster angewandt, die Hintergrundfarbe auf den "weißen" Hintergrund des Objekts.
Hinweis: Um Vorder- und Hintergrundfarbe nur auf den Rahmen des Objekts anzuwenden, wählen Sie es mit gedrückter **Umschalttaste** aus.
- Das Schaltflächengitter **Füllen** bestimmt das Füllmuster für das ausgewählte Objekt. Das Schaltflächengitter **Rahmen** bestimmt das Füllmuster für den Rahmen. Diese Funktionen haben keine Auswirkung auf ein Datenfeld oder statischen Text.
- Das PopUp-Menü **Breite** bestimmt die Breite einer Linie oder bei einem zweidimensionalen Objektrahmen die Breite eines Rahmens.

Fügen Sie ein Objekt in ein Etikett ein, nimmt es die Eigenschaften des zuletzt erzeugten Objekts an. Wollen Sie dagegen die Standardeinstellungen zuweisen, wählen Sie das Objekt aus und klicken am unteren Rand auf die Schaltfläche **Standardwerte**. Die Vorder- und Hintergrundfarben, Füllmuster und Linienbreite auf die Standardeinstellungen zurückgesetzt.

Textstil ändern

In den Bereichen "Text" und "Stil" definieren Sie das Aussehen des gewählten Textes: Schriftart, Schriftgröße, Format und Textstil. Diese Parameter gelten für Felder und statischen Text.

- Über die DropDown-Menüs **Schrift** und **Größe** können Sie diese Parameter für den gewählten Text setzen. Sie zeigen die Schriftarten und -größen, die in Ihrem Betriebssystem verfügbar sind. Sie können eine Schriftgröße auch manuell eingeben.
- Über das DropDown-Menü **Ausrichtung** setzen Sie die Ausrichtung der Titel in Bezug auf den um sie gezogenen Auswahlrahmen.
- Über die Combobox **Format** können Sie dem ausgewählten Feld ein Anzeigeformat zuweisen. In diesem Fall werden die Werte der Datensätze gemäß diesem Parameter formatiert, wenn das Etikett gedruckt wird.

Die anwendbaren Formate richten sich nach dem Typ des gewählten Feldes. Im Eingabebereich der Combobox können Sie auch eigene Formate definieren. Weitere Informationen hierzu finden Sie im Abschnitt **Anzeigeformate**.

- Im Bereich **Stil** können Sie standardmäßige Schriftstile auf den ausgewählten Text anwenden.

Etikettenbogen einrichten

Im Etiketteneditor können Sie das Blatt konfigurieren, auf dem die Etiketten gedruckt werden. Sie können die Blattgröße sowie die Größe eines einzelnen Etiketts setzen, die Anzahl der Etiketten pro Blatt, die Ausrichtung, Reihenfolge, u.v.m. bestimmen.

Um ein Etikettenblatt zu erstellen:

1. Gehen Sie zur Seite "Größe".
2. Setzen Sie Ausrichtung und Reihenfolge der Etiketten über die entsprechenden Icons im oberen Bereich. Die Ausrichtung des Blatts ist von der Seitenausrichtung unabhängig. Wollen Sie die Seitenausrichtung ändern, klicken Sie am unteren Rand auf die Schaltfläche **Papierformat**. Auf dem Bildschirm erscheint dann der Druckdialog für Ihr Betriebssystem, wo Sie die Seitenausrichtung direkt ändern können. Falls erforderlich, wird die Vorschau entsprechend Ihrer Auswahl angepasst.
3. Wählen Sie bei Bedarf in der DropDown-Liste **Etikettentyp** einen vordefinierten Standardetikettenbogen aus. Die Eingabebereiche auf der Seite verändern sich entsprechend des gewählten Etikettenbogens. Weitere Informationen dazu finden Sie im Abschnitt **Standardwerte (optional)**. Sie können diese Angaben bei Bedarf verändern. Ansonsten können Sie direkt zu Schritt 9 gehen.
4. In den Eingabefeldern **Nebeneinander** und **Untereinander** legen Sie die Etikettenanzahl pro Reihe und Spalte fest. Sie können direkt eine Zahl eintragen oder die Pfeile neben dem Eingabefeld verwenden. Um die Angaben zu bestätigen, klicken Sie in einen anderen Bereich oder betätigen die **Tabulatortaste** oder die **Zeilenschaltung**. Ist **Größe automatisch anpassen** markiert, wird die Größe eines Etiketts dem Etikettenbogen angepasst. Die Vorschau wird entsprechend angepasst, um die Darstellung der Etiketten auf einer gedruckten Seite anzuzeigen.
5. In der DropDown-Liste **Einheit** können Sie bei Bedarf die Maßeinheit für Ihr Etikett bzw. Ihre Etikettenseiten ändern. Die Werte der darüberliegenden Bereiche werden in die neue Einheit umgewandelt.
6. Geben Sie Werte in die Eingabebereiche **Oberer Rand**, **Linker Rand**, **Vertikaler Abstand** und **Horizontaler Abstand** ein. Die gesetzten Ränder gelten für das Etikettenblatt, die Abstände werden zwischen die einzelnen Etiketten gesetzt.
7. Klicken Sie auf das Optionsfeld **Papiergröße**, um rechten Rand und unteren Rand zu setzen. Diese Ränder erscheinen in der Vorschau als blaue Linien.
8. Klicken Sie auf das Optionsfeld **Etikettengröße**, um die Etikettenbreite und Etikettenhöhe zu definieren. Sie können die Etikettengröße auch automatisch berechnen. Weitere Informationen dazu finden Sie im Abschnitt **Größe automatisch anpassen**.
9. Wollen Sie mehr als eine Kopie pro Etikett ausdrucken, wählen Sie in der DropDown-Liste **Anzahl pro Datensatz** die gewünschte Anzahl zum Drucken. Die Kopien werden dann hintereinander auf dem Etikettenbogen gedruckt. 4D dupliziert nicht die gesamte Etikettenseite.
10. Wird die erste Seite des Etikettenpapier nur teilweise verwendet, klicken Sie auf das erste leere Etikett in der Vorschau. 4D druckt die Etiketten dann ab diesem Etikett. Das gilt nur für die erste Seite.
11. Soll beim Drucken von Etiketten eine Methode laufen, wählen Sie in der DropDown-Liste **Methode anwenden** die gewünschte Methode aus. Weitere Informationen dazu finden Sie im Abschnitt **Etiketten eine Methode zuweisen**.
Das Etikettenblatt ist vollständig. Sie können auf die Registerkarte "Etikett" klicken, um zu prüfen, ob der Inhalt des Etiketts in die definierte Größe passt.

Standardwerte (optional)

Über diese Schaltfläche können Sie Etiketten-, Seitengröße und Ränder durch Auswählen eines kommerziellen Standardformats aus der DropDown-Liste wählen.

Sofern in 4D Standardvorlagen installiert sind, erscheinen sie in der DropDown-Liste zur Schaltfläche **Standardwerte**. Wählen Sie hier eine Option aus, werden alle Werte für den Etikettenbogen automatisch gesetzt und der Bogen ist sofort druckfertig. Sie können aber trotzdem die Anzahl Etiketten, die pro Datensatz gedruckt werden soll, sowie die anzuwendende Methode bestimmen.

Größe automatisch anpassen

Sie können die Breite und Höhe Ihrer Etiketten manuell setzen oder diese über den Etiketteneditor anhand der anderen definierten Parameter berechnen lassen. In diesem Fall verwendet 4D den Bereich so, dass pro Blatt soviel Etiketten wie möglich gedruckt werden.

Markieren Sie die Option **Größe automatisch anpassen**, wird die Etikettengröße gemäß folgender Parameter berechnet:

- Anzahl Etiketten pro Zeile und Spalte
- Oberer und linker Rand
- Horizontaler und vertikaler Abstand

Hinweis: Haben Sie einen rechten oder unteren Rand eingegeben, werden sie nicht berücksichtigt.

Ändern Sie einen dieser Parameter, wenn die Option **Größe automatisch anpassen** markiert ist, werden die Etiketten automatisch in der Vorschau angepasst.

Etiketten eine Methode zuweisen

Sie können Ihren Etiketten eine Projektmethode zuweisen, die beim Drucken ausgeführt wird und definieren, ob sie pro Etikett oder pro Datensatz angewendet wird.

Wählen Sie dazu im DropDown-Menü **Methode anwenden** eine Methode aus. Es enthält die Namen aller Projektmethoden, die in der Datenbank angelegt wurden. Gibt es keine, ist dieses Menü leer.



Hinweis: Dieses DropDown-Menü zeigt keine Komponenten- oder Projektmethoden, die als unsichtbar deklariert sind.

Haben Sie die Methode gewählt, müssen Sie unter **Anwenden pro** festlegen, ob Sie einmal **pro Etikett** oder **pro Datensatz** angewendet wird. Diese Option hat keine Wirkung, wenn **Anzahl pro Datensatz** auf 1 steht.

Etiketteneditor (64-bit)

Hinweis zur Kompatibilität

Die 4D Version für 64-bit bietet einen neuen Etiketteneditor, der in zukünftigen 4D Versionen den derzeitigen Etiketteneditor (32-bit) ersetzen wird. Dieses Kapitel beschreibt den neuen Etiketteneditor.

-  Beschreibung des Etiketteneditors
-  Etikettendateien verwalten

Beschreibung des Etiketteneditors

Der Etiketteneditor von 4D ist das ideale Werkzeug zum Gestalten und Drucken einer Vielzahl verschiedener Etiketten. Sie können folgendes durchführen:

- Etiketten für Briefe, Aktenordner, Namensschilder und für viele andere Anwendungen gestalten
- Schrift, Schriftgröße und Stil für die Etiketten festlegen
- Dekorative Elemente in Etikettenvorlagen erstellen und einfügen
- Anzahl der Etiketten nebeneinander und untereinander pro Seite festlegen
- Mehrere Etiketten pro Datensatz drucken
- Seitenränder der Etiketten festlegen
- Eine Methode festlegen, die beim Drucken eines Etiketts oder Datensatzes ausgeführt wird
- Eine Vorschau erstellen und Etiketten drucken

Etiketten lassen sich auch mit dem Formulareditor erstellen. Damit können Sie spezielle Etiketten gestalten, die Variablen enthalten oder die Optionen der Zeichenwerkzeuge im Formulareditor nutzen. Weitere Informationen dazu finden Sie im Abschnitt **Etiketten erstellen**.

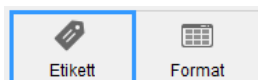
Mit dem Etiketteneditor erstellen, formatieren und drucken Sie Etiketten. Er enthält Einstellungen zum Anlegen von Etiketten und das Positionieren der Etiketten auf Etikettenbögen. Sie möchten zum Beispiel für Adressetiketten eine Vorlage erzeugen mit dem Vornamen und Nachnamen der Person in der ersten Zeile, der Straße in der zweiten usw. In der Vorlage können Sie die Anzahl der Etiketten auf einer Seite und die Ränder des Etikettenpapiers festlegen, so dass der Etikettentext auf den Etiketten zentriert wird.

Haben Sie ein passendes Layout erstellt, können Sie es auch auf der Festplatte speichern und später wieder verwenden.

Um den Etiketteneditor zu öffnen:

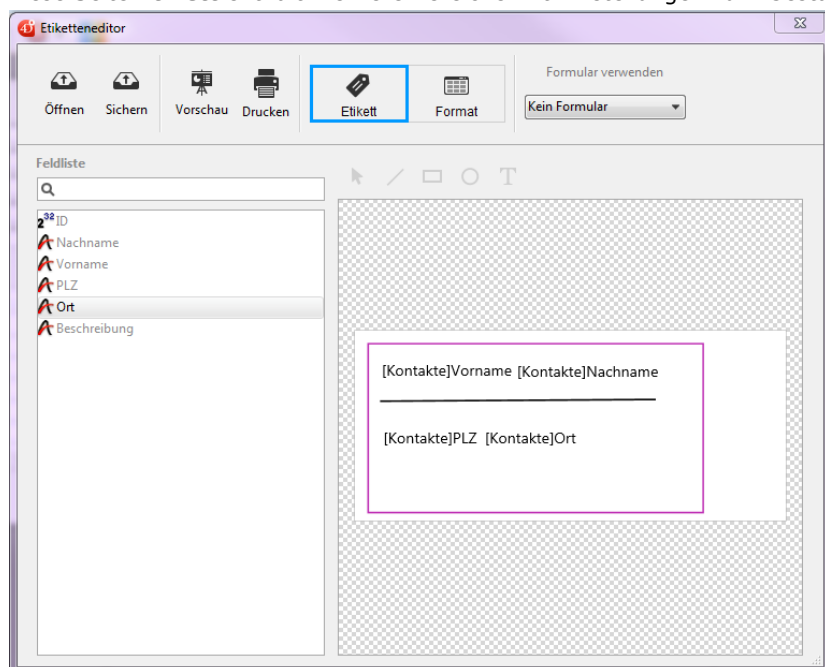
- Wählen Sie in der Designumgebung im Menü **Bericht** oder in der Werkzeugleiste über das Icon **Tools** den Befehl **Etiketten**
ODER
- Wählen Sie in der Anwendungsumgebung den 4D Befehl **PRINT LABEL**.

Der Etiketteneditor enthält eine Leiste mit mehreren Schaltflächen. Über die Schaltfläche **Etikett** legen Sie den Inhalt eines Etiketts fest, über **Format** Größe und Position mehrerer Etiketten auf einer Seite.



Seite Etikett

Diese Seite **Etikett** enthält mehrere Bereiche mit Einstellungen zum Gestalten und Formatieren von Etiketten:



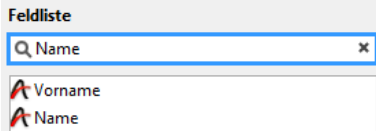
- **Feldliste:** Dieser Bereich zeigt die Namen der Datenfelder in der aktuellen Tabelle in hierarchischer Reihenfolge. Wenn diese Tabelle mit anderen Tabellen verknüpft ist, sind die verknüpften Datenfelder unter Windows mit einem Pluszeichen, auf Macintosh mit einem Pfeil markiert. Die Datenfelder in der verknüpften Tabelle erscheinen eingerückt.

Um ein Feld aus dieser Liste für das Etikett zu verwenden, ziehen Sie es einfach in den Vorschaubereich.

Hinweise:

- Im Etiketteneditor erscheinen nur sichtbare Tabellen und Datenfelder. Weitere Informationen zum Ausblenden von Tabellen und Datenfeldern finden Sie im Abschnitt "Unsichtbar" unter **Tabelleneigenschaften** und **Feldeigenschaften**
- Der Etiketteneditor unterstützt keine Datenfelder vom Typ Objekt

Über den Suchbereich können Sie die Feldliste eingrenzen:



- **Etikettenvorschau:** Hier gestalten Sie Ihr Etikett. Sie setzen alle Einträge für Ihr Etikett an die passende Stelle. Das weiße Rechteck stellt ein einzelnes Etikett dar. Seine Größe wird auf der Seite **Format** festgelegt. Sie können Felder in die Vorschau ziehen.
 - Sie können auch zwei Felder zusammensetzen, z.B. Vor- und Nachname. Setzen Sie dazu das zweite Feld auf das erste. Die Felder werden automatisch durch einen Abstand voneinander getrennt. Mit gedrückter Shift-Taste werden sie durch Zeilenschaltung getrennt. Auf diese Weise können Sie z.B. Adressetiketten mit mehreren überlappenden Feldern erstellen (Adresse1, Adresse2, etc.), ohne eine Leerzeile zu erzeugen, wenn eine Adresse nur ein Feld benötigt.
 - Sie können Bilddateien und Etikettendateien (nur mit der Endung .4lbp) per Drag-and-Drop vom Desktop in den Gestaltungsbereich des Etiketts ziehen.
 - Felder und eingefügter Text lassen sich verändern. Durch Doppelklick auf den gewünschten Teil wechseln Sie in den Bearbeitungsmodus. Auf diese Weise können Sie zusammengesetzte Einträge entfernen oder ändern.

[Angestellte]Vorname+[Angestellte]Name

- Sie können auch ein Format auf ein Feld anwenden. Dazu setzen Sie den Befehl **String** ein, z.B.:

String([Employees]Date of birth;Internal date long)

- **Zoom:** Sie können den Gestaltungsbereich des Etiketts variieren. Dazu ziehen Sie ihn mit dem Cursor an der unteren rechten Ecke größer oder kleiner:



- **Formular verwenden:** Mit dieser DropDown Liste können Sie ein Tabellenformular als Etikettenvorlage verwenden. Das gewählte Formular muss an das Erstellen von Etiketten angepasst sein. Der Etiketteneditor ist dann teilweise deaktiviert: nur Funktionen der Seite **Format** sind verwendbar, damit Sie die Seite anhand des Formulars konfigurieren können. In der Vorschau erscheint die Darstellung des gewählten Formulars und 4D führt alle mit dem Formular verknüpften Formular- oder Objektmethoden aus. Weitere Informationen dazu finden Sie im Abschnitt **Über den Etiketteneditor**. Mit dieser Option können Sie auch eine Projektmethode zum Ausführen jedes Datensatzes oder Etiketts bestimmen und dann Variablen zuweisen (siehe unteren Absatz **Etiketten mit Formularen und Methoden drucken (Beispiel)**).

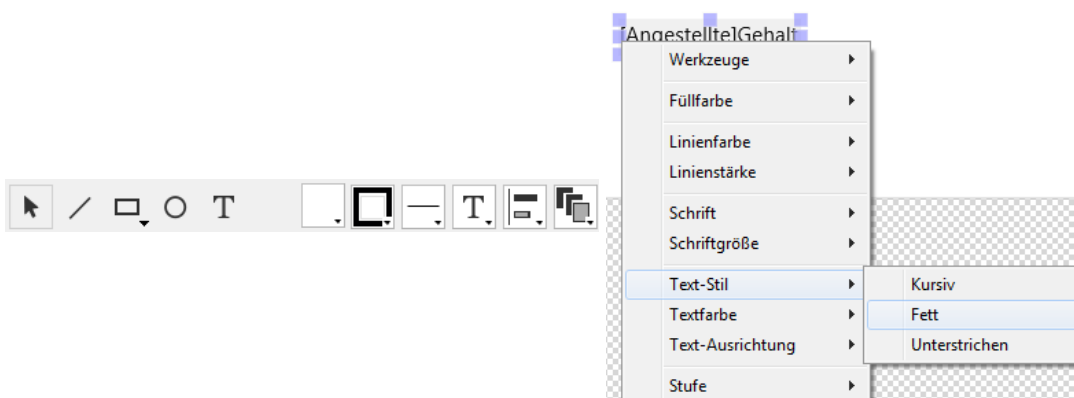
Wollen Sie Ihre Etiketten mit dem Editor erstellen, müssen Sie **Kein Formular** auswählen.

Hinweise:

- Sie können die hier aufgelisteten Formulare über eine spezifische JSON Datei einschränken (siehe **Verfügbare Formulare und Methoden definieren**)
- Gibt es in der Anwendung keine Tabellenformulare, erscheint dieses Menü nicht

Funktionen des Gestaltungsbereichs

Der Gestaltungsbereich des Editors bietet eine **Werkzeugleiste** und ein **Kontextmenü**. Damit können Sie im Handumdrehen eine ansprechende Etikettenvorlage erstellen.



Der linke Teil der Werkzeugleiste enthält Funktionen zum Auswählen und Einfügen von Objekten. Sie lassen sich auch über das Kontextmenü des Gestaltungsbereichs aufrufen.

- ▣ Werkzeug für Auswahl. Klicken Sie auf ein einzelnes Objekt oder ziehen Sie einen Auswahlrahmen um mehrere Objekte. Wollen Sie verschiedene Objekte separat auswählen, wählen Sie die passenden Objekte mit gedrückter Shift-Taste aus.
- ▣ Werkzeug für Linie
- ▣ Werkzeug für Rechteck und Rechteck mit abgerundeten Ecken
- ▣ Werkzeug für Kreis
- ▣ Werkzeug zum Einfügen von Text. Ziehen Sie ein Rechteck und geben Sie darin Text ein. Sie können jeden Textbereich bearbeiten, mit Doppelklick auch solche mit Feldreferenzen.

Über Tastenkürzel oder die Pfeiltasten der Tastatur können Sie Objekte exakter bewegen oder die Größe verändern:

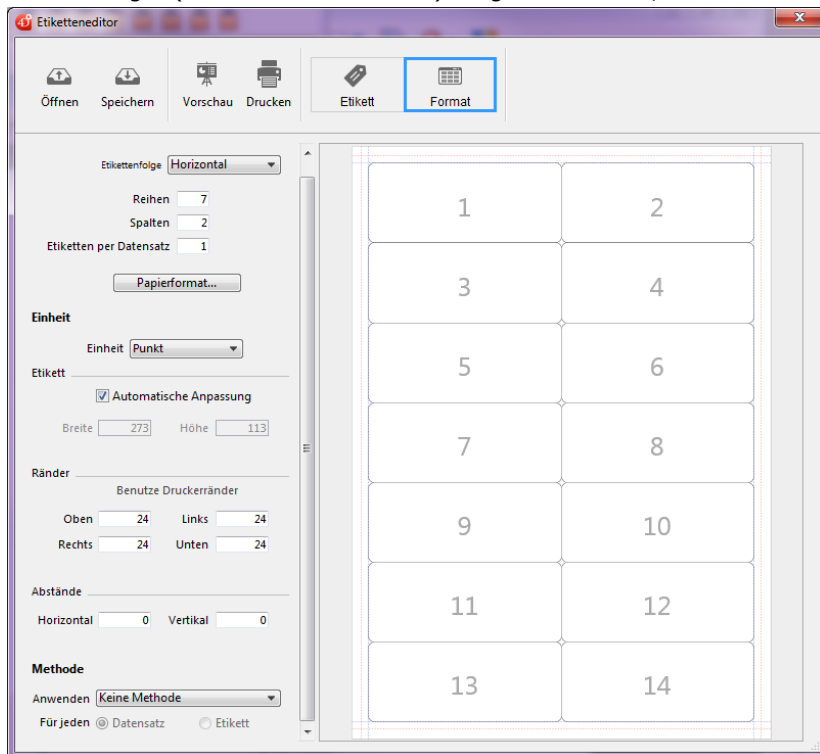
- Mit den Pfeiltasten bewegen Sie die Auswahl der Objekte jeweils um 1 Pixel
- **Umschalttaste** + Pfeiltaste bewegt die Auswahl um jeweils 10 Pixel
- **Strg** + Pfeiltaste vergrößert oder verkleinert die Auswahl der Objekte jeweils um 1 Pixel
- **Strg** + **Großschreibtaste** + Pfeiltaste vergrößert oder verkleinert die Auswahl der Objekte jeweils um 10 Pixel

Der rechte Teil der Werkzeugleiste enthält Funktionen zum Gestalten einzelner Elemente in der Etikettenvorlage:

- Füllfarbe (Farbicons zeigen die ausgewählte Farbe)
- Linienfarbe
- ≡ Linienstärke
- ⌘ Schrift. Setzt Schrifttyp und Schriftgröße, sowie Stil, Farbe und Ausrichtung für die ausgewählten Textblöcke. Ausrichten und Verteilen. Zum Ausrichten müssen zwei oder mehr Objekte ausgewählt sein. Zum Verteilen müssen mindestens drei Objekte ausgewählt sein. Dann werden zwischen die Objekte in der Auswahl automatisch gleiche horizontale oder vertikale Abstände gesetzt.
- ⌘ Stufe (Objektebene). Setzt Objekte ganz oder jeweils eine Ebene nach vorne bzw. nach hinten.

Seite Format

Die Seite **Format** enthält Optionen zum Drucken von Etiketten gemäß den Einstellungen des Druckers, den Sie im Druckmanager (Auswahl auf Macintosh) ausgewählt haben, so dass die Seite richtig formatiert wird.



- **Etikettenfolge:** Gibt an, ob die Etiketten nach Reihen (horizontal) oder nach Spalten (vertikal) gedruckt wird.
- **Reihen** und **Spalten:** Setzt die Anzahl Etiketten, die pro Reihe und pro Spalte auf einem Blatt gedruckt werden soll. Ist **Automatische Anpassung** markiert, wird die Größe eines Etiketts dem Etikettenbogen angepasst.
- **Etiketten per Datensatz:** Setzt die Anzahl Kopien, die pro Etikett gedruckt werden soll (Kopien werden hintereinander gedruckt).
- **Papierformat:** Damit legen Sie das Seitenformat zum Drucken der Etiketten fest. Klicken Sie auf diese Schaltfläche, erscheint das Dialogfenster des Druckers, der in Ihrem Betriebssystem eingerichtet ist. Standardmäßig ist A4 im Hochformat vorgegeben.
Hinweis: Das vom Editor angelegte Blatt basiert auf der logischen Seite des Druckers, d.h. die tatsächliche Seite minus der nicht bedruckbaren Ränder. Diese Ränder erscheinen in der Vorschau als gestrichelte Linien.
- **Einheit:** In dieser Dropdown-Liste legen Sie die Maßeinheit für Ihr Etikett bzw. Ihre Etikettenseiten fest. Sie können Punkt, Millimeter, Zentimeter oder Zoll angeben.
- **Automatische Anpassung:** Damit berechnet 4D die Etikettengröße automatisch, d.h. die Etikettenbreite und -höhe gemäß den in allen anderen Parametern gesetzten Werten. Ist diese Option markiert, wird die Etikettengröße bei jeder Änderung eines Seitenparameters angepasst. Die Parameter **Breite** und **Höhe** lassen sich dann nicht mehr manuell setzen.
- **Breite und Höhe:** Damit legen Sie Breite und Höhe des Etiketts manuell fest. Ist **Automatische Anpassung** markiert, können Sie hier keine Werte eingeben.
- **Ränder** (Oben, Rechts, Links, Unten): Setzt die Ränder für das Etikettenblatt. Sie erscheinen in der Vorschau als blaue Linien.

Klicken Sie auf **Benutze Druckerränder** erscheinen in der Vorschau die im ausgewählten Drucker definierten Ränder (Diese Werte lassen sich verändern).

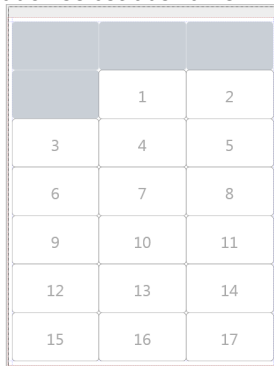
- **Abstände:** Bestimmt den vertikalen bzw. horizontalen Abstand zwischen den Etikettenreihen und -spalten.
- **Methode anwenden:** Damit können Sie beim Drucken eine bestimmte Methode auslösen. Sie können z.B. eine Methode anwenden, die Druckdatum und -zeit für jedes Etikett ausgibt. Dieses Feature ist auch hilfreich, um Etiketten mit einem bestimmten Tabellenformular zu drucken (siehe "Formular verwenden" im oberen Abschnitt **Seite Etikett**). In diesem Fall können Sie Variablen über eine Methode füllen.
Damit eine Projektmethode für Etiketten auswählbar ist, muss sie folgende Eigenschaften haben:
 - Sie muss für die Anwendung zugelassen sein (zugelassene Methoden richten sich nach Datenbank-Eigenschaften und dem Befehl **SET ALLOWED METHODS**), andernfalls erscheint sie nicht im Menü **Anwenden**.
 - Die Option "Von Komponenten und Host benutzt" muss aktiviert sein.

Ein Beispiel dazu finden Sie im folgenden Abschnitt **Etiketten mit Formularen und Methoden drucken (Beispiel)**.

Hinweis: Für fortgeschrittenen Einsatz können Sie auch eine json Datei anlegen, um die Liste der verfügbaren Methoden einzuschränken, siehe unten **Verfügbare Formulare und Methoden definieren**.

Die Optionsfelder **Für jeden: Datensatz** oder **Etikett** bestimmen, ob die Methode einmal pro Etikett oder pro Datensatz angewendet wird. Sie haben nur eine Auswirkung, wenn Sie von jedem Etikett mehrere Exemplare drucken und während dem Drucken eine Methode anwenden.

- **Vorschau Layout:** Dieser Bereich stellt die verkleinerte Ansicht einer kompletten Etikettenseite dar, die auf den eingegebenen Werten basiert. Die Seitenvorschau zeigt auch die Papiergröße an, die im Dialogfenster Druckeinstellung festgelegt ist. Sie können auch das erste Etikett auf der Seite auswählen, das gedruckt werden soll (gilt bei mehreren Seiten nur für die 1. Seite). Das ist z.B. beim Drucken von selbstklebenden Etikettenbögen hilfreich, wenn ein Teil davon bereits verwendet wurde. Durch Anklicken können Sie das erste Etikett auf der Seite auch selbst auswählen.

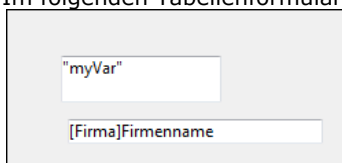


	1	2
3	4	5
6	7	8
9	10	11
12	13	14
15	16	17

Etiketten mit Formularen und Methoden drucken (Beispiel)

Sie können dedizierte Tabellenformulare und Projektmethoden verwenden, um Etiketten mit berechneten Variablen zu drucken. Dieses einfache Beispiel zeigt, wie Sie die verschiedenen Elemente konfigurieren:

1. Fügen Sie in einem dedizierten Tabellenformular Ihre Felder für das Etikett und Variablen hinzu.
Im folgenden Tabellenformular mit Namen "Etikett" haben wir die Variable *myVar* hinzugefügt:

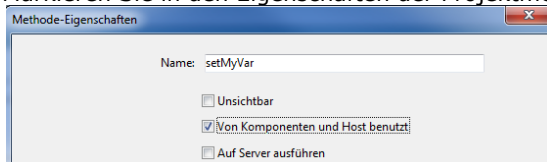


"myVar"
[Firma]Firmenname

2. Legen Sie die Projektmethode **setMyVar** mit folgendem Code an:

```
C_LONGINT(myVar)
myVar:=myVar+1
```

3. Markieren Sie in den Eigenschaften der Projektmethode die Option "Von Komponenten und Host benutzt":



Methode-Eigenschaften

Name: setMyVar

Unsichtbar

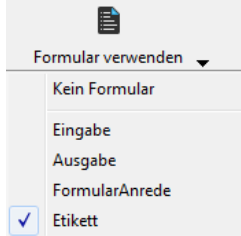
Von Komponenten und Host benutzt

Auf Server ausführen

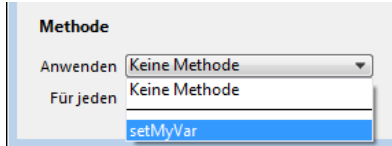
4. Legen Sie eine 2. Projektmethode mit folgendem Code an, um sicherzustellen, dass die Projektmethode **setMyVar** zugelassen ist:

```
ARRAY TEXT($methods;1)
$methods{1}:="setMyVar"
SET ALLOWED METHODS($methods)
```

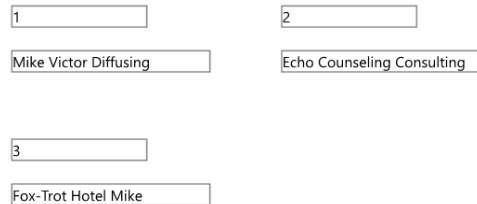
5. Öffnen Sie nun den Etiketteneditor und verwenden Ihr Formular:



6. Wählen Sie auf der Seite **Format** die Methode aus:



7. Nun können Sie Ihre Etiketten drucken:



Verfügbare Formulare und Methoden definieren

Über ein zusätzliches Feature im Etikettenassistenten lässt sich festlegen, welche Projektformulare und Methoden (innerhalb der zugelassenen Methoden) im Dialogfenster auswählbar sind:

- Auf der Seite "Etikett" rechts oben im PopUp-Menü **Formular verwenden**
- Auf der Seite "Format" links unten im PopUp-Menü **Methode anwenden**

Dazu müssen Sie lediglich eine JSON Datei im Projektordner hinzufügen.

Um für Etiketten auswählbare Formulare bzw. Methoden zu definieren:

1. Erstellen Sie eine JSON Datei mit Namen **labels.json** und legen sie im Datenbankordner in den Ordner **Resources**.
2. Fügen Sie in dieser Datei die Namen von Formularen bzw. von Projektmethoden hinzu, die in den Menüs des Etikettenassistenten auswählbar sein sollen.

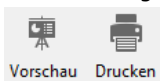
Der Inhalt der Datei **labels.json** sollte lauten wie folgt:

```
[ {"tableId":2,"forms":[],"methods":["myMethod1","myMethod2"]}, {"tableId":1,"forms":["Sample Label 1","Sample Label 2"],"methods":[]} ]
```

Wurde keine Datei **labels.json** definiert, gibt es keine Einschränkung.

Vorschau und Drucken

Die Werkzeugleiste enthält zwei Schaltflächen für Vorschau und Drucken der Etikettenseite:



Die Schaltfläche **Vorschau** zeigt die erste Etikettenseite, auf der jedes Etikett mit gestrichelten Linien angezeigt wird.

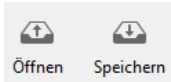
Etikettendateien verwalten

Etikettenlayouts speichern und laden

Sie können in 4D ein Etikettenlayout abspeichern und dann später im Etikettenassistenten wieder laden. Auf diese Weise können Sie eine an Ihre Anforderungen angepasste Etikettenbibliothek einrichten. Die Datei wird mit den im Etikett definierten Einstellungen und den Layoutseiten gespeichert.

Sie können Etikettendateien per Drag-and-Drop vom Desktop in den Layoutbereich des Etiketts ziehen.

Etikettenlayouts werden über die Schaltflächen **Öffnen** und **Speichern** in der Toolbar verwaltet:



- Zum Laden eines Etikettenlayouts klicken Sie auf die Schaltfläche **Öffnen** und wählen im DropDown-Menü das gewünschte Etikett. Gibt es im Etiketteneditor bereits ein Etikett, wird es durch das geladene ersetzt.
- Zum Sichern eines Etikettenlayouts klicken Sie auf die Schaltfläche **Speichern** und geben Name und Speicherort für das Etikett an.

Dateiformat

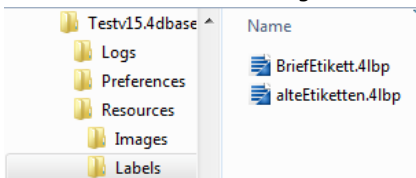
Die Dateiendung für gesicherte 4D Etiketten ist ".4lbp". Dieses Format gilt, seit es intern in XML geschrieben wird.

Hinweis zur Kompatibilität: 4D Etikettendateien, die mit dem bisherigen Etiketteneditor erstellt wurden (mit der Endung ".4lb") werden unterstützt. Der Editor kann solche Etikettenlayouts ohne Änderung des Formats laden, ändern und speichern.

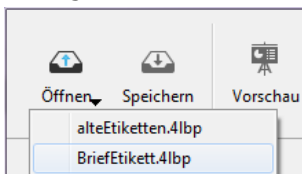
Gespeicherte Etikettendateien laden

Im Etikettenassistenten können Sie Etikettendateien innerhalb Ihrer Anwendung speichern, so dass der Benutzer über die Schaltfläche **Öffnen** direkt Etikettenlayouts auswählen und öffnen kann.

Hierfür müssen Sie lediglich im Ordner **Resources** Ihrer Anwendung den Unterordner **Labels** einrichten und Ihre Etikettendateien darin ablegen:



Wird der Etikettenassistent gestartet und dieser Ordner mit gültigen Etikettendateien gefunden, erhält die Schaltfläche **Öffnen** den Pfeil zum Öffnen des PopUp-Menüs. Im PopUp-Menü lassen sich die gespeicherten Etikettenlayouts auswählen. Unterstützt werden sowohl die Standarddateien mit der Endung ".4lbp" als auch die bisherigen mit der Endung ".4lb":



Daten importieren und exportieren

-  Einleitung
-  Dateiformate
-  Begrenzer für Datenfeld, Datensatz und Datei
-  Daten aus Datei importieren
-  Daten in Datei exportieren
-  Import und Export über ODBC Datenquelle
-  Einstellungen für Import-Export speichern und laden

Die Import- und Exportfunktionalitäten von 4D bieten Ihnen einen schnellen und sicheren Weg, um Informationen in bzw. aus Ihrer Anwendung zu übertragen. Sollten Sie von einer anderen Datenbank oder Tabellenkalkulation zu 4D wechseln, erspart das Importieren das erneute Eintippen der Daten. Sie können Daten aus Datenverwaltungsprogrammen unter Windows importieren, aber auch aus jeder Anwendung, die unter einem anderen Betriebssystem arbeitet und in Windows übertragen wurde.

Sie können auch Daten aus 4D exportieren, die danach anderen Programmen zur Verarbeitung zur Verfügung stehen. So können Sie beispielsweise Daten in ein spezialisiertes Grafik- oder Statistikprogramm exportieren. Sie können die exportierten Daten auch auf einen anderen Rechnertyp übertragen.

Sie importieren oder exportieren Daten mit den Einstellungen der entsprechenden Dialogfenster oder über ein Formular. Importieren und Exportieren über Dialogfenster ist schneller als mit einem Formular. Dagegen können Sie in einem Formular Objekt- oder Formularmethoden zur Verarbeitung der Daten einsetzen. Beim Import oder Export mit einem Formular werden die Objektmethoden für jeden bearbeiteten Datensatz ausgeführt.

Sie können die Einstellungen für Import und Export in der Datei Voreinstellungen sichern und dann erneut laden.

Beim Importieren oder Exportieren über eine Datei geben Sie das Dateiformat, die Begrenzer für Datenfelder und Datensätze sowie die entsprechenden Datenfelder an.

Schließlich können Sie Daten aus bzw. in eine ODBC-Quelle ohne dazwischengeschaltete Datei importieren bzw. exportieren. Weitere Informationen dazu finden Sie im Abschnitt **Import und Export über ODBC Datenquelle**.

Hinweise:

- Sie können Datensätze auch über den Schnellberichteditor exportieren. Weitere Informationen dazu finden Sie im Abschnitt **Textdatei** über die Option in **Schnellbericht ausführen**.
- Sie können Datensätze auch über die 4D Programmiersprache importieren bzw. exportieren. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Import und Export**.

Das Dateiformat legt die Art und Weise fest, in der die Datenfelder und Datensätze angeordnet werden. Einige Dateiformate enthalten zudem Informationen für die Formatierung. 4D unterstützt verschiedene Dateiformate und bietet auch interne Formatierungsoptionen.

Unterstützte Formate

4D unterstützt folgende Formate zum Importieren und Exportieren der Daten.

Text

Dieses Format trennt die Datenfelder innerhalb eines Datensatzes mit einem Feldbegrenzer und die Datensätze mit einem Datensatzbegrenzer. Der Datensatzbegrenzer richtet sich nach der Zielplattform: Zeilenschaltung/Line return für Windows und Zeilenschaltung für Mac OS.

Dieses Format ist keinem spezifischen Zeichensatz zugeordnet; im Menü **Zeichensatz** können Sie dem am besten passenden Typ wählen. Standardmäßig ist UTF-8 markiert. Sie können auch den 4D Befehl **USE CHARACTER SET** verwenden. UTF-16 wird nicht unterstützt.

Beim Exportieren der Daten schreibt 4D eine Byte Order Mask (BOM), wenn der Zeichensatz UTF-8 ist. Findet 4D beim Datenimport ein BOM, ist der hier angegebene Zeichensatz vorrangig.

Text fester Länge

Dieses Format weist pro Feld eine bestimmte Anzahl Zeichen zu. Hier werden keine Feldbegrenzer verwendet. Werte wie Namen können natürlich nicht immer dieselbe Anzahl Buchstaben haben. Deshalb werden die Leerstellen mit Füllzeichen, in der Regel mit Leerzeichen aufgefüllt. Sie können für jeden Datentyp Füllzeichen bestimmen oder die Standardzeichen verwenden (siehe Abschnitt **Seite Auffüllen**). Werte mit mehr Zeichen als vorgegeben, werden dagegen abgeschnitten.

Beim Import entfernt 4D alle Füllzeichen am Ende jeder Zeichenkette.

Weitere Merkmale dieses Formats sind identisch mit dem Format Text.

DIFF

Die Abkürzung steht für Data Interchange Format. DIF ist ein Standardformat, das in der Regel den Zeichensatz "IBM437" verwendet, auch "Latin-US (DOS)" genannt. Dieser wird von den 4D Befehlen **CONVERT FROM TEXT** und **Convert to text** unterstützt. Wählen Sie dieses Format, verwendet 4D keine aktuellen Import-/Exportfilter, die über den 4D Befehl **USE CHARACTER SET** gesetzt wurden.

Die Datensatzbegrenzer richten sich nach der Zielplattform: Zeilenschaltung+Neue Zeile für Windows und Zeilenschaltung für Mac OS.

Im DIF Format können Zeichenketten keine Zeilenschaltungen enthalten (ASCII Codes 10 oder 13). 4D ersetzt sie beim Export durch Leerzeichen. Außerdem wird das Anführungszeichen " als interner Begrenzer verwendet. Alle Anführungszeichen, die in Zeichenketten für den Export gefunden werden, werden in doppelte Zeichen umgewandelt ("").

Beispiel: Montag\rDienstag"Mittwoch" wird in DIF als "Montag Dienstag ""Mittwoch"" exportiert.

Beim Import entfernt 4D alle Füllzeichen am Ende jeder Zeichenkette.

DBF

Dieser Name steht für das Format "dBase". dBase wird häufig von Datenbanken unter DOS und Windows verwendet.

4D verarbeitet dieses Dateiformat auf dieselbe Weise wie das DIFF Format.

Beachten Sie, dass Zeichenketten im DBF Format als feste Länge mit Füllzeichen exportiert werden (je nach Datentyp ' ' oder '0'). Beim Importieren entfernt 4D alle Füllzeichen am Ende von Zeichenketten.

SYLK

Die Abkürzung steht für SYmbolic LinK format.

Diesem Format ist kein spezifischer Zeichensatz zugewiesen; Sie können im Menü **Zeichensatz** einen passenden Satz definieren (standardmäßig UTF-8) oder den 4D Befehl **USE CHARACTER SET** verwenden. Byte Order Masks werden nicht verarbeitet. UTF-16 wird nicht unterstützt.

Die Datensatzbegrenzer richten sich nach der Zielplattform: Zeilenschaltung+Neue Zeile für Windows und Zeilenschaltung für Mac OS.

Alle Zeilenschaltungen (ASCII Code 13), die in exportierten Zeichenketten gefunden werden, werden folgendermaßen umgewandelt: `<escape><blank>`: unter Windows und `<escape><blank>=` auf Mac OS.

Hinweis: Das Plug-In 4D View unterstützt auch das SYLK Format für Importieren und Exportieren.

4D Anwendung

Dieses Dateiformat ist 4D-spezifisch. Damit tauschen Sie problemlos Datensätze zwischen verschiedenen 4D Anwendungen aus. Dieses Format unterstützt alle Datentypen von 4D, inkl. Bilder und BLOB Felder. Die Endung der 4D Exportdatei ist ".4IE."

Ist dieses Format gewählt, können Sie keine Zielplattform wählen. Sie können also keinen spezifischen Zeichensatz wählen (Unicode wird immer verwendet), Import-/Exportfilter, die über den 4D Befehl **USE CHARACTER SET** installiert wurden, werden ignoriert.

XML

XML (eXtensible Markup Language) ist eine Standardsprache für den Datenaustausch. Sie arbeitet mit Tags, die eine exakte Beschreibung der ausgetauschten Daten sowie der Struktur ermöglichen. XML Dateien sind Textdateien, deren Inhalt die Programme beim Importieren der Daten analysieren. Viele Programme unterstützen dieses Format. Weitere Informationen zur XML Terminologie finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **XML**.

Diesem Format ist kein spezifischer Zeichensatz zugewiesen; Sie können im Exportdialog einen Satz setzen. Er wird direkt im XML Dokument angegeben. Ein Zeichensatz für Import/Export, der über den 4D Befehl **USE CHARACTER SET** gesetzt ist, wird ignoriert.

Die Datensatzbegrenzer richten sich nach der Zielplattform: Zeilenschaltung+Neue Zeile für Windows und Zeilenschaltung für Mac OS.

Optionen zur Formatierung

Für jedes importierte oder exportierte Dateiformat gibt es folgende Optionen.

Zeichensatz

Das Menü **Zeichensatz** ist für Dateiformate vom Typ Text, SYLK und XML (Export) verfügbar. Es enthält eine Liste der standardmäßigen Zeichensätze, wie sie IANA vorgibt. Weitere Informationen dazu finden Sie unter: <http://www.iana.org/assignments/character-sets>.

Hinweis: Das Menü **Zeichensatz** ist für die Formate DIFF und DBF für den Zeichensatz "IBM437" gesperrt. Für das 4D Format ist dieses Menü nicht verfügbar.

- Beim Exportieren verwenden Sie dieses Menü, um die Codierung für die exportierten Daten zu definieren.
- Beim Importieren verwenden Sie dieses Menü, um die Codierung für die importierten Daten zu definieren. Dieses Menü ist inaktiv, wenn der Kopfteil der Exportdateien ein BOM enthält (siehe **Seite Kopf**). In diesem Fall wird die Codierung der importierten Datei automatisch vorgegeben.
Die Standardcodierung für Import- und Exportoperationen ist UTF-8, bzw. der Zeichensatz, definiert durch den 4D Befehl **USE CHARACTER SET** (wenn er ausgeführt wurde). Beachten Sie, dass das Auswählen einer Codierung im Import- oder Exportdialog nicht den aktuellen Zeichensatz der Anwendung verändert.

Zielplattform

Über dieses DropDown-Menü definieren Sie **Begrenzer für Datenfeld, Datensatz und Datei**. Es gibt folgende Optionen:

- **Automatisch:** Wert für Datensatzende wird gemäß der aktuellen Plattform gesetzt.
- **Macintosh:** Feldende = Tab, Datensatzende = Zeilenschaltung, Dateiende = <Keine>
- **Windows:** Feldende = Tab, Datensatzende = Zeilenschaltung+Zeilenvorschub, Dateiende = <Keine>
- **Unix:** Datensatzende = Zeilenvorschub
- **Eigene:** Zeigt die **Seite Trennzeichen**.

Begrenzer für Datenfeld, Datensatz und Datei

Feldbegrenzer werden zwischen die Datenfelder eines Datensatzes gesetzt, Datensatzbegrenzer an das Ende jedes Datensatzes. Dateibegrenzer werden an das Ende der Import- bzw. Export-Datei gesetzt. Sie sind beim Austausch mit bestimmten Anwendungen notwendig.

Begrenzer werden nur mit den Formaten **Text** und **Text fester Länge** eingesetzt. Das Format *Feste Länge* verwendet keine Datenfeldbegrenzer, ein Textformat verwendet keinen expliziten Dateibegrenzer.

Beim Importieren von Daten im Textformat erkennt 4D an den darin enthaltenen Begrenzungszeichen, wo Datenfelder und Datensätze enden. Beim Exportieren fügt 4D diese Begrenzungen automatisch für Sie ein.

Sind beim Exportieren von Daten Begrenzungszeichen für Datenfelder und Datensätze vorhanden, werden sie automatisch durch Leerzeichen ersetzt. Auf diese Weise werden sie nicht fälschlicherweise als Begrenzer interpretiert und unterbrechen nicht den Importprozess. Verwenden Sie z.B. einen Strichpunkt als Datenfeldbegrenzer und enthält das Feld "a;b", ist der exportierte Wert "a b".

Wir empfehlen jedoch, vor dem Export sicherzustellen, dass die Datenfelder keine Zeichen enthalten, die ebenfalls als Feld- oder Datensatzbegrenzer verwendet werden.

Daten aus Datei importieren

Sie können Daten aus Dateien in XML, 4D, Text, SYLK, DIFF, DBF, XML oder Textformate mit fester Länge importieren (siehe **Dateiformate**). Beim Importieren von Daten, die Sie aus einer anderen Anwendung exportiert haben, beachten Sie bitte die Dokumentation der jeweiligen Anwendung für Details zum Exportieren in ein Standardformat. Sollte diese Anwendung ein anderes Exportformat benutzen, müssen Sie die Daten möglicherweise vorher in einem Textverarbeitungsprogramm überarbeiten.

Wollen Sie Daten aus einer anderen Anwendung importieren, die auf derselben Plattform wie Ihre 4D Anwendung läuft, exportieren Sie zuerst die Daten mit dem anderen Programm. Notieren Sie die Reihenfolge, in der die Datenfelder exportiert wurden, das Format, in dem das Programm die Daten abspeicherte und, sofern Sie das Textformat benutzen, die verwendeten Begrenzungszeichen. Beim Importieren der Daten müssen Sie diese Einstellungen exakt in dieser Weise wiederherstellen.

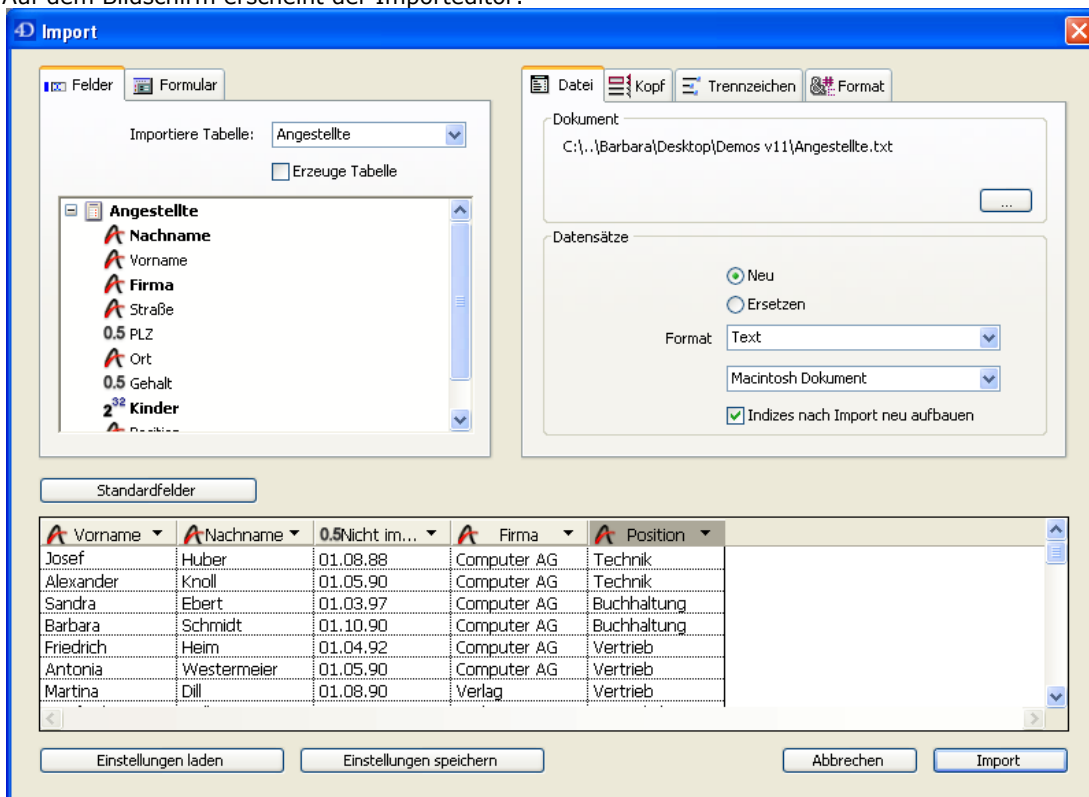
Sie müssen nicht die komplette Tabellenstruktur vor dem Import erzeugen. Sie können die erforderlichen Tabellen direkt im Dialogfenster für den Import erstellen. Sie können wählen, ob Sie die Felder für den Import im Importdialog auswählen, oder ein Eingabeformular angeben, das die Felder für den Import enthält.

Im Designmodus importieren Sie Daten über das Dialogfenster Import. In der Anwendungsumgebung können Sie dieses Fenster über den 4D Befehl **IMPORT DATA** anzeigen oder Daten direkt über die 4D Befehle im Kapitel **Import und Export** importieren.

Daten für den Import festlegen

Um Daten in der Designumgebung zu importieren:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Importieren>Von Datei**.
Wählen Sie am unteren Rand den Dateityp aus, den Sie importieren möchten — Text, SYLK, DIF, DBF, 4IE oder XML. Weitere Informationen dazu finden Sie im Abschnitt **Dateiformate**.
2. Wählen Sie die zu importierende Datei und klicken auf die Schaltfläche **Öffnen**.
Auf dem Bildschirm erscheint der Importeditor:



Rechts im Bereich Dokument erscheint der Dateiname mit dem Zugriffspfad und dem gewählten Dateityp für den Import. Über die Schaltfläche [...] können Sie eine andere Datei auswählen.

3. Wählen Sie im Auswahlbereich für Tabellen und Felder die Tabelle und die Felder, in welche die Daten importiert werden sollen. Es gibt folgende Optionen:
 - **Daten in vorhandene Tabelle und Felder importieren.**
Wählen Sie eine Tabelle in der DropDown Liste „Importiere Tabelle“ (standardmäßig ist die erste Tabelle ausgewählt). Wählen Sie dann in der hierarchischen Liste die Datenfelder, in welche die Daten importiert werden sollen.
 - **Neue Tabelle für den Datenimport erstellen.**
Markieren Sie dazu das Kontrollkästchen **Erzeuge Tabelle**. 4D erstellt dann eine neue Tabelle, in welche die Daten importiert werden sollen. 4D legt anhand der importierten Daten die Anzahl der benötigten Felder und den jeweiligen Typ fest.

Wollen Sie den Namen der neuen Tabelle bzw. des Feldes ändern, doppelklicken Sie im Auswahlbereich darauf, damit sie eingebbar werden oder markieren Sie die Option **Spaltentitel als Feldname** auf der Seite Kopfteil (siehe [Seite Kopf](#)).

Wollen Sie den Datentyp ändern, klicken Sie in der Vorschau in die Titelleiste des entsprechenden Feldes und wählen im DropDown-Menü einen anderen Datentyp:



Diese Tabelle wird nur während dem Import erstellt. Brechen Sie den Import ab oder deaktivieren Sie diese Option, bleibt die Struktur der Datenbank unverändert.

Hinweis: Über das Kennwort-Zugriffssystem können Sie die Option **Erzeuge Tabelle** auch deaktivieren, so dass Benutzer während dem Importieren in die Datenbank keine neue Tabelle erstellen können. Legen Sie dazu in den Datenbank-Eigenschaften eine Gruppe mit Zugriff auf den Designmodus fest - dazu muss natürlich zuvor das Kennwortsystem von 4D aktiviert sein. Benutzer, die nicht zu dieser Gruppe gehören, können dann im Importdialog nicht die Option **Erzeuge Tabelle** auswählen.

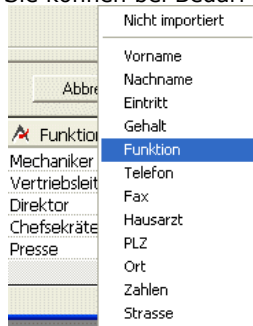
- o **Daten mit Formular importieren:** Informationen dazu finden Sie im Abschnitt **Daten mit Formular importieren**.

Die Vorschau zeigt den Inhalt der Importdatei in Spalten an. Importieren Sie Daten mit einem Formular, erscheinen alle Felder des Formulars. Sie können jede Spalte anpassen.

4. Ändern Sie bei Bedarf im Bereich Vorschau die Felder, in welche die Daten importiert werden sollen.

Hinweis: Diese Option ist nicht verfügbar, wenn Sie für den Import ein Formular verwenden oder das Kontrollkästchen **Erzeuge Tabelle** markiert ist.

Die Kopfzeile jeder Spalte zeigt den Namen des importierten Feldes und über das vorangestellte Icon den Typ an. Sie können bei Bedarf das DropDown-Menü aufklappen und ein anderes Feld auswählen.



Um ein Feld einer Spalte zuweisen, können Sie auch die Spalte auswählen und in der hierarchischen Liste der Felder auf den Feldnamen doppelklicken.

Sie können auch die Option **Nicht importiert** wählen. Dann wird diese Spalte nicht importiert.

- o **Standardfelder**

Mit dieser Schaltfläche weisen Sie den Feldern der 4D Tabelle in "intelligenter Weise" Werte aus der Importdatei zu, d.h. gemäß ihrem Typ und nicht gemäß ihrer Eingabereihenfolge. Der Typ des Importwertes wird analysiert und dann einem passenden 4D Feld zugewiesen. Die Interpretation geschieht folgendermaßen:

Geschätzter Typ des Feldes 4D Typ

Alle Zahlen	Zahl
Wahr/Falsch	Boolean
Datum in einem 4D Format	Datum
Zeit	Zeit
Andere	Alpha

Wird keine Entsprechung gefunden, erhält der Importwert die Bemerkung "Nicht importiert". Wollen Sie ihn importieren, müssen Sie im Bereich Vorschau ein 4D Feld per Hand zuweisen.

- o **Anzahl Zeichen für Formate Text fester Länge:** Importieren Sie eine Datei mit dem Format Text fester Länge, zeigt der Bereich Vorschau für jede Spalte die Anzahl der Zeichen an. Die Anzahl richtet sich nach der ersten Zeile der Daten. Sie können die Verteilung der Zeichen durch Ziehen der Spaltenränder ändern, jedoch keine Zeichen hinzufügen.

5. (Optional) Setzen Sie die Optionen im Bereich "Datensätze".

Weitere Informationen dazu finden Sie im Abschnitt **Bereich Datensätze**.

6. (Optional) Bestimmen Sie weitere Optionen über die Seiten Kopf, Trennzeichen, Füllen bzw.Format.

Weitere Informationen dazu finden Sie im Abschnitt **Importoptionen**.

7. Wollen Sie die Einstellungen sichern, klicken Sie am unteren Rand auf die Schaltfläche **Einstellungen speichern**.

Weitere Informationen dazu finden Sie im Abschnitt **Einstellungen für Import-Export speichern und laden**.

8. Klicken Sie auf die Schaltfläche **Import** um die Datei zu importieren.

Auf dem Bildschirm erscheint ein Laufbalken, der den Verlauf des Imports anzeigt

Importoptionen

Über die Registerkarten bietet 4D bietet verschiedene Importoptionen. Die Anzahl und ihr Inhalt richten sich nach dem gewählten Dateityp sowie den Optionen.

Bereich Datensätze

Die Optionen im Bereich "Datensätze" (Seite **Datei**) geben die Hauptmerkmale der Operation an:

- **Neu** oder **Ersetzen**: Die importierten Datensätze können zu den vorhandenen Daten hinzugefügt werden und eine neue Auswahl bilden (Option **Neu**) oder die aktuelle Auswahl ersetzen (Option **Ersetzen**).
- **Format, Zeichensatz** und **Zielplattform**: Optionen zum Formatieren der Importdatei (siehe Abschnitt **Dateiformate**)
- **Indizes nach Import neu aufbauen**: Diese Option ist standardmäßig markiert, d.h. die Indizes der Datenfelder mit importierten Daten werden nach dem Import neu aufgebaut. Dies kann den Import umfangreicher Datenmengen beschleunigen. Ist die importierte Datenmenge geringer als die bereits vorhandenen Daten, ist es sinnvoll, diese Option vor dem Import zu deaktivieren. Dann wird der Index schrittweise aktualisiert und nicht komplett neu aufgebaut.

Seite Kopf

Diese Seite ist nur für den Import aus Dateien vom Typ Text und Text mit fester Länge verfügbar.

- **Importiere ab Zeile #**: Damit legen Sie fest, ab welcher Zeile 4D Daten importieren soll. Das ist besonders hilfreich, wenn die Importdatei mit unformatierten Zeilen wie Überschrift, Datum, etc. beginnt. Denn die Spalten für den Import werden nach dem Format der ersten Zeile berechnet.
- **Spaltentitel als Feldname**: Diese Option erscheint, wenn die Option **Erzeuge Tabelle** markiert ist. Damit weisen Sie 4D an, die Spaltentitel als Feldnamen zu verwenden.

Seite Trennzeichen

Hier definieren Sie die Trennzeichen für Importdateien vom Typ Text und Text mit fester Länge. Diese Optionen werden gemäß der Plattform, die im Menü "Zielplattform" von der **Bereich Datensätze** angegeben ist, vorkonfiguriert.

Ausführliche Informationen dazu finden Sie im Abschnitt **Begrenzer für Datenfeld, Datensatz und Datei**. Über die DropDown-Liste können Sie die Standard Trennzeichen ändern. Sie zeigt die am häufigsten verwendeten Formate für Trennzeichen. In den Eingabebereichen können Sie auch direkt ein Trennzeichen vorgeben. Über das DropDown-Menü Anzeigeformat lassen sich die Trennzeichen in verschiedenen Formaten anzeigen: **Zeichen**, **Dezimal** (Standardformat) und **Hexadezimal**. Beachten Sie, dass die Option **Zeichen** bei nicht-druckbaren Zeichen als Begrenzer, wie Tabulator, Zeilenvorschub und Zeilenschaltung nichts anzeigt.

Hinweis: Trennzeichen werden wieder auf ihre Standardwerte zurückgesetzt, wenn Sie den Dokumenttyp auf der Seite **Datei** ändern.

Seite XML

Diese Seite erscheint nur für XML Dateien. Hier stellen Sie den Modus zum Analysieren des Inhalts des importierten XML Inhalts ein.

4D interpretiert beim Import den Inhalt des XML Dokuments, um die Information zu entnehmen. Diese Operation wird standardmäßig ohne explizite Bestätigung ausgeführt. 4D geht davon aus, dass das XML Dokument "well-formed" ist, d.h.

dass die Struktur korrekt und seine Interpretation eindeutig ist.

Sie können jedoch beim Import auch die Bestätigung anfordern. Dazu markieren Sie die Option **Prüfung**. In diesem Fall analysiert 4D den Inhalt des Dokuments gemäß der DTD (Document Type Definition) und prüft, ob es dieser Definition entspricht. Der Import wird nur bei Bestätigung des Dokuments ausgeführt.

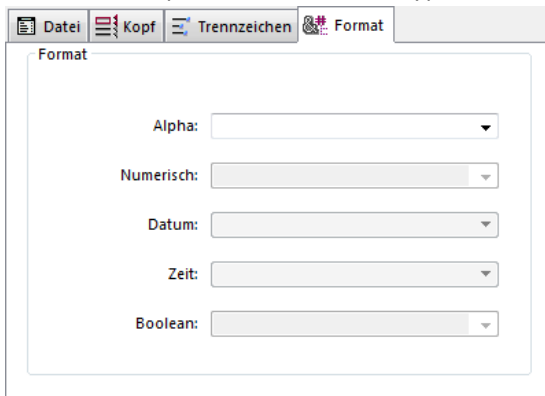
Hinweis: Weitere Informationen zu DTD finden Sie auf der [Seite XML](#).

Ist das DTD des importierten Dokuments nicht im Dokument selbst enthalten, sondern in einer separaten Datei, oder wollen Sie das Dokument über eine andere DTD bestätigen, wählen Sie die Option **Alternative DTD verwenden** und geben über die Schaltfläche [...] die Datei mit der gewünschten DTD an.

Nach Beenden des Importvorgangs wird das Dialogfenster geschlossen. Die Tabelle, in welche die Daten importiert wurden, wird die aktuelle Tabelle.

Seite Format

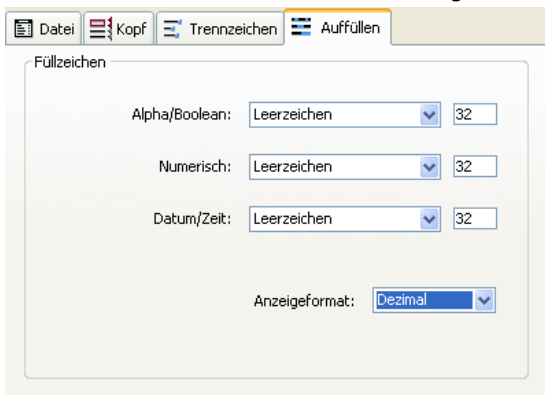
Diese Seite ist für den Import aus Dateien vom Typ Text, Text mit fester Länge und XML verfügbar. Hier können Sie Formate für importierte Felder vom Typ Boolean definieren.



Die Syntax dafür lautet Wert_Wahr;Wert_Falsch. Importieren Sie ein Boolean Feld mit den Werten "Schwarz" (für wahr) und "Weiß" (für falsch), geben Sie Schwarz;Weiß ein. Die Combobox zeigt die häufigsten Boolean Formate. Sie können aber auch selbst ein Format eingeben.

Seite Auffüllen

Diese Seite erscheint nur, wenn für das Dateiformat "Text fester Länge" ausgewählt ist. Damit definieren Sie je nach Art der Werte die Füllzeichen. Standardmäßig ist dafür das Leerzeichen definiert.



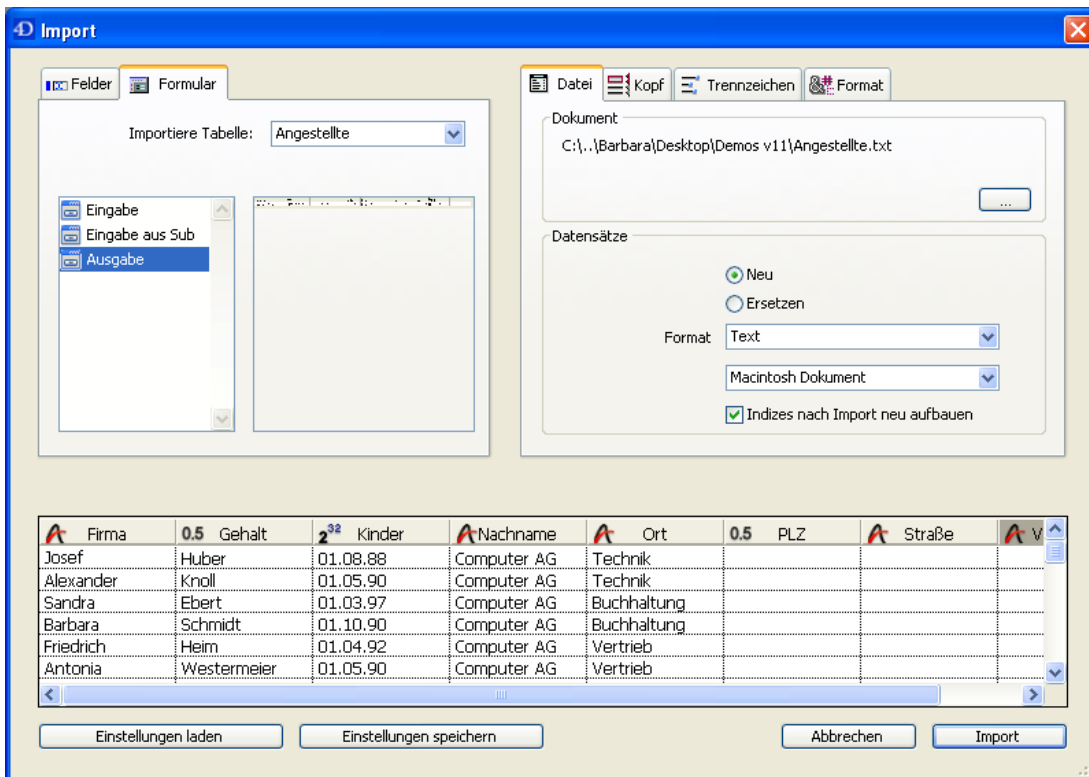
Weitere Informationen dazu finden Sie im Abschnitt [Dateiformate](#).

Für folgende Datentypen können Sie Füllzeichen verwenden: **Alpha/Boolean** (Felder vom Typ Boolean werden als Alpha importiert), **Numerisch** und **Datum/Zeit**. Das Leerzeichen dient normalerweise zum Füllen von festen Formatfeldern. Die DropDown Menüs bieten weitere Werte an. Im Eingabebereich rechts daneben erscheinen die ausgewählten Zeichen. Wollen Sie eigene Zeichen einsetzen, geben Sie diese direkt im jeweiligen Eingabebereich ein.

Über das DropDown Menü Anzeigeformat lassen sich die Füllzeichen in verschiedenen Formaten anzeigen: **Zeichen**, **Dezimal** (Standardformat) und **Hexadezimal**. Ein nicht-druckbares Füllzeichen erscheint nicht, wenn Sie das Menü **Zeichenformat** verwenden.

Daten mit Formular importieren

Wollen Sie für den Import ein Formular verwenden, klicken Sie im oberen linken Bereich auf die Registerkarte Formular. Der Vorteil eines Formulars ist, dass die zugeordneten Formular- oder Objektmethoden bei jedem Datensatz ausgeführt werden, der importiert wird. So können Sie die Daten beim Import steuern und müssen keine eigene Importanweisung schreiben.



Sie können Formulare mit eingebbaren Variablen und Feldern auswählen, jedoch keine Schaltflächen. Sind den Variablen oder Feldern Methoden zugeordnet, werden sie bei jedem Datensatz ausgeführt, der importiert wird. Diese Methoden werden jedoch nicht in der Vorschau ausgeführt.

Beim Import mit Formular sollten Sie folgendes beachten:

- Methoden werden im Formularereignis On Validate ausgeführt.
- Die Eingabereihenfolge des Formulars bestimmt, wie die Daten importiert werden. Achten Sie deshalb darauf, dass die Eingabereihenfolge des Formulars die Reihenfolge der Importdaten widerspiegelt.
- Hat das Formular weniger Felder und Variablen als Spalten in der Textdatei, werden zusätzliche Spalten ignoriert.
- Ein Formular für den Import sollte keine Schaltflächen enthalten. Objekte in Unterformularen werden nicht berücksichtigt.

Hinweis zu 4D Server: In Serverprozeduren lassen sich Daten nicht über Formulare importieren, die Formularmethoden enthalten, da auf dem Server keine Formularereignisse verwaltet werden.

Daten in Datei exportieren

Beim Exportieren von Daten erzeugen Sie eine Datei vom Typ 4D, XML, SYLK, DIFF, DBF, Text bzw. Text mit fester Länge, die von anderen Programmen geöffnet oder importiert werden können. Weitere Informationen über das Importieren in andere Anwendungen finden Sie in der jeweiligen Dokumentation. Wählen Sie ein Dateiformat und Feld oder Datensatzbegrenzer, die mit der anderen Anwendung kompatibel sind.

In manchen Fällen müssen Sie erst bestimmte Exportoptionen einstellen, damit die Exportdatei für die Zielanwendung verwendbar ist. So akzeptieren manche Anwendungen zwar das Textformat, setzen aber voraus, dass der erste Datensatz die Feldnamen enthält. Diesen Datensatz können Sie über die Seite **Kopf** eingeben.

4D exportiert die Datensätze in der aktuellen Sortierfolge. Sie können alle Datensätze einer Tabelle oder nur die aktuelle Auswahl exportieren.

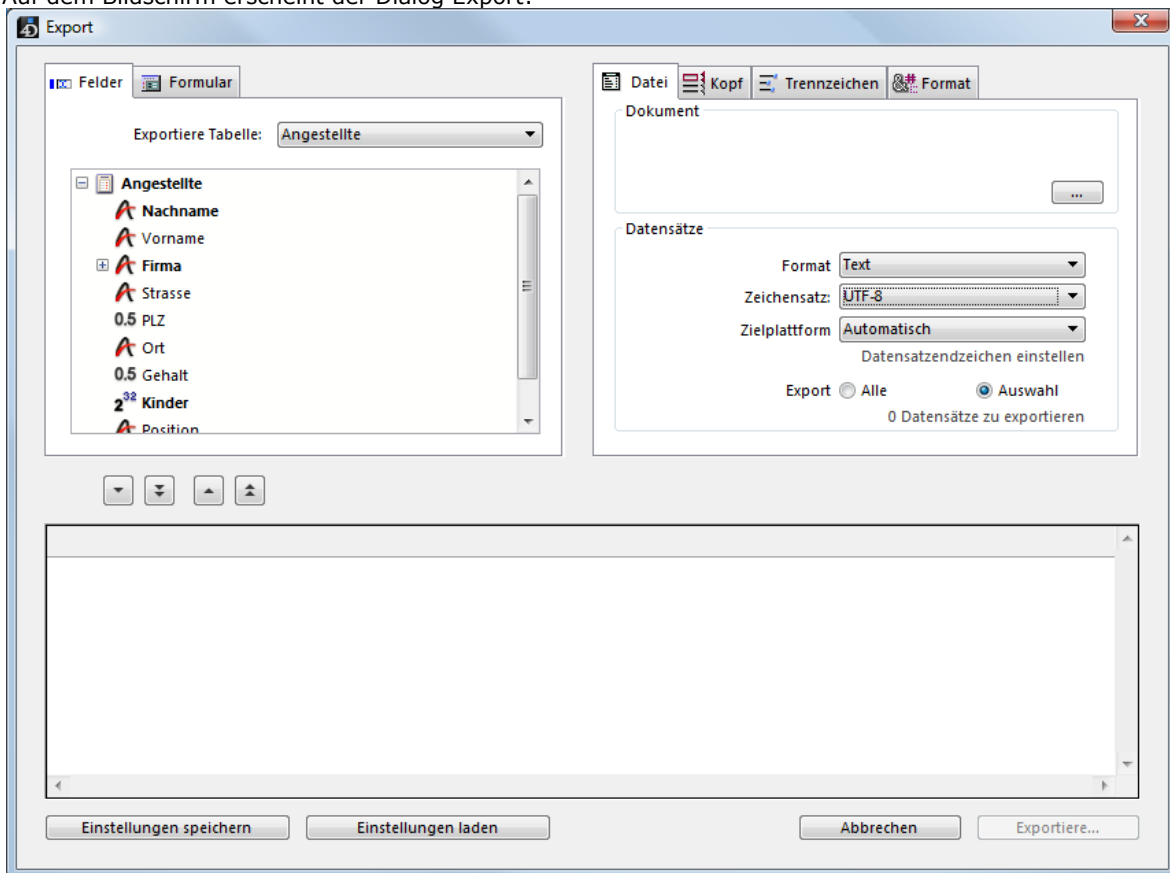
Sie haben die Möglichkeit, die Datensätze für den Export im Dialogfenster Export auszuwählen oder ein Listenformular anzugeben. Verwenden Sie ein Formular, werden die Datenfelder aus diesem Formular exportiert.


In der Designumgebung exportieren Sie Daten über das Dialogfenster Export. In der Anwendungsumgebung können Sie dieses Fenster über den 4D Befehl **EXPORT DATA** aufrufen oder Daten direkt über die 4D Befehle im Kapitel **Import und Export** exportieren.

Daten für den Export festlegen

Um Daten in der Designumgebung zu exportieren:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Exportieren>In Datei**.
Auf dem Bildschirm erscheint der Dialog Export:



2. Wählen Sie im rechten Teil unter Datensätze das Format für die Exportdatei aus — **Text, Text feste Länge, DIFF, SYLK, DBF, 4D** oder **XML** und Formatierungsoptionen (siehe Abschnitt **Dateiformate**).
3. (Optional) Klicken Sie auf die Schaltfläche [...] geben bei Bedarf Name und Speicherort für die Exportdatei ein und klicken auf die Schaltfläche **Sichern**.
Der Exportdialog erscheint erneut, im rechten Teil unter Dokument steht jetzt der Pfadname der Exportdatei. Dieser Schritt startet noch nicht den Export! (Wenn Sie diese Schritte weglassen, können Sie Name und Speicherort der Exportdatei auch über die Schaltfläche **Exportieren** festlegen).
4. Wählen Sie im Auswahlbereich für Tabellen und Felder die Tabelle und die Felder, die Sie exportieren wollen. Sie können zwischen folgenden Optionen wählen:
 - o **Tabelle und Felder für den Export manuell wählen.**
Wählen Sie im DropDown-Menü Exportiere Tabelle eine Tabelle aus und dann in der darunterliegenden Liste die entsprechenden Feldnamen. Sie können sie durch Doppelklick oder durch Markieren und die Pfeiltasten unter der Liste auswählen:


- **Daten über Formular exportieren.**

Damit exportieren Sie die Datenfelder aus dem Formular. Weitere Informationen dazu finden Sie im Abschnitt **Daten mit Formular exportieren**.

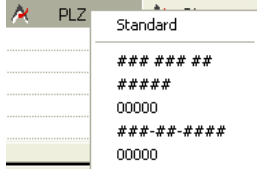
5. (Optional) Ändern Sie in der Vorschau die Felder, aus denen die Daten exportiert werden.

Hinweis: Diese Option ist nicht verfügbar, wenn Sie für den Export ein Formular verwenden.

Die Vorschau zeigt den Inhalt der Exportdatei in Spalten an. Die Kopfzeile jeder Spalte ist ein DropDown-Menü, über das Sie ein anderes Datenfeld wählen können. Sie können jede Spalte in der Größe anpassen.

Das Icon vor dem Feldnamen zeigt den Datentyp an. Klicken Sie mit der rechten Maustaste in die Kopfzeile, erscheint das Kontextmenü mit den verfügbaren Anzeigeformaten. Hier können Sie ein anderes Format wählen.

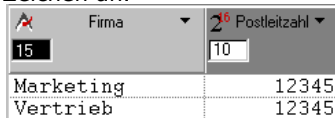
Wählen Sie kein Format, wird das Standardformat verwendet. Diese Option können Sie auch auf der Seite **Format** definieren (siehe **Seite Format**).



Sie können eine Spalte aus der Exportdatei und der Vorschau entfernen. Wählen Sie dazu die Spalte aus und klicken über der Vorschau auf den nach oben zeigenden Pfeil.

- Anzahl Zeichen für Formate Text fester Länge

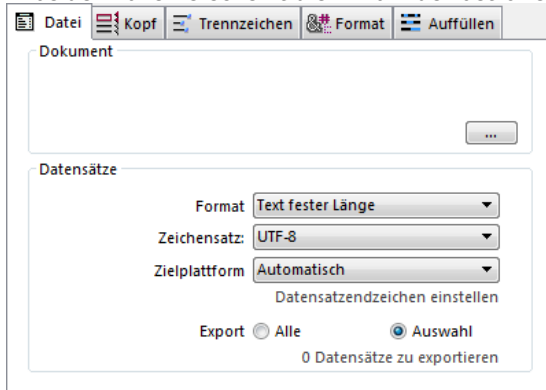
Wählen Sie im rechten Teil das Format Text mit fester Länge, zeigt die Vorschau für jede Spalte die Anzahl der Zeichen an.



Sie können diese Standardvorgaben ändern. Geben Sie dazu im jeweiligen Eingabebereich einen neuen Wert ein oder ziehen Sie die Spaltenränder auf die gewünschte Größe. Füllzeichen werden bei alphanumerischen Feldern immer am Ende, bei numerischen Feldern immer am Anfang angefügt. Standardmäßig wird die Anzahl der Zeichen folgendermaßen gesetzt: Textfelder sind 80 Zeichen, numerische Felder 10 Zeichen lang. Die Länge von alphanumerischen Feldern richtet sich nach der im Struktureditor festgelegten Länge.

6. Geben Sie im rechten Teil unter Datensätze an, ob Sie alle Datensätze der gewählten Tabelle (Option **Export Alle**) oder nur die aktuelle Auswahl exportieren wollen (Option **Export Auswahl**).

In beiden Fällen erscheint die Anzahl der betroffenen Datensätze.



7. (Optional) Bestimmen Sie weitere Optionen über die Seiten **Kopf**, **Trennzeichen**, **Format** bzw. **Auffüllen** und **XML**. Weitere Informationen dazu finden Sie im Abschnitt **Optionen für Export**.

8. Wollen Sie die Einstellungen sichern, klicken Sie am unteren Rand auf die Schaltfläche **Einstellungen speichern**.

Weitere Informationen dazu finden Sie im Abschnitt **Einstellungen für Import-Export speichern und laden**.

9. Klicken Sie auf die Schaltfläche **Exportiere**, um die Datei zu exportieren.

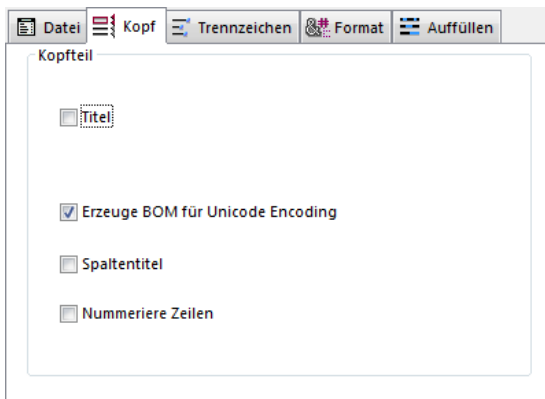
Haben Sie für die Exportdatei noch keinen Namen und Speicherort festgelegt (siehe Schritt 3), erscheint der Standarddialog zum Sichern. Andernfalls wird der Export direkt ausgeführt.

Optionen für Export

Der Exporteditor von 4D bietet im rechten Teil verschiedene Optionen in Form von Registerkarten. Ihre Anzahl variiert je nach Art der Datei oder der auszuführenden Operation.

Seite Kopf

Diese Seite ist nur für den Export in Textdateien verfügbar:

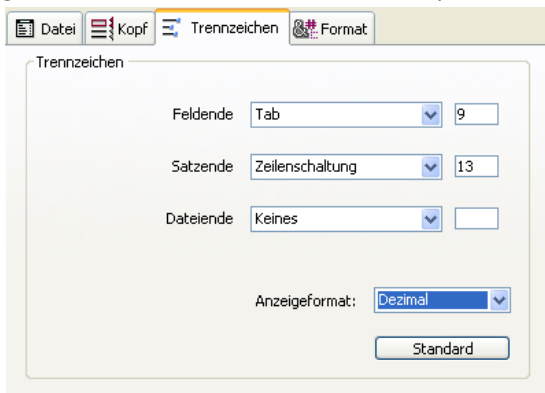


Sie können folgende Optionen definieren:

- **Titel:** Hier legen Sie fest, ob das Exportdokument einen Titel hat, den Sie dann im darunterliegenden Bereich eingeben.
- **Erzeuge BOM für Unicode Encoding:** Fügt eine Byte Order Mark (BOM) in den Kopfteil der Exportdatei ein. Diese zusätzliche Information erleichtert die Interpretation von Text durch die Import-Software, sofern sie diese Funktion unterstützt. Diese Option ist standardmäßig markiert. Sie wird jedoch nur berücksichtigt, wenn für den Export der Zeichensatz Unicode ausgewählt ist. Andernfalls wird keine BOM hinzugefügt.
- **Spaltentitel:** Diese Option exportiert die Feldnamen als ersten 'Datensatz' in der Exportdatei.
- **Nummeriere Zeilen:** Diese Option nummeriert jeden exportierten Datensatz mit einer Zeilennummer, beginnend mit 1 und erhöht jeweils um 1.

Seite Trennzeichen

Hier definieren Sie die Trennzeichen für Importdateien vom Typ Text und Text mit fester Länge. Diese Optionen werden gemäß der Plattform, die im Menü "Zielformat" von der **Bereich Datensätze** angegeben ist, vorkonfiguriert.

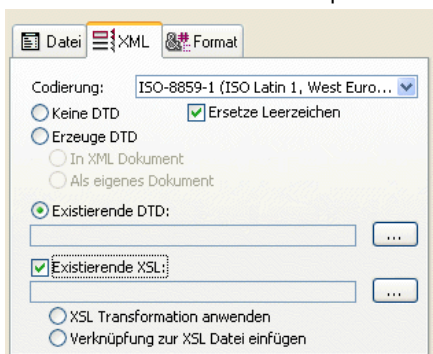


Ausführliche Informationen dazu finden Sie im Abschnitt **Begrenzer für Datenfeld, Datensatz und Datei**. Über die DropDown-Liste können Sie die Standard Trennzeichen ändern. Sie zeigt die am häufigsten verwendeten Formate für Trennzeichen. In den Eingabebereichen können Sie auch direkt ein Trennzeichen vorgeben. Über das DropDown-Menü Anzeigeformat lassen sich die Trennzeichen in verschiedenen Formaten anzeigen: **Zeichen**, **Dezimal** (Standardformat) und **Hexadezimal**. Beachten Sie, dass die Option **Zeichen** bei nicht-druckbaren Zeichen als Begrenzer, wie Tabulator, Zeilenvorschub und Zeilenschaltung nichts anzeigt.

Hinweis: Trennzeichen werden wieder auf ihre Standardwerte zurückgesetzt, wenn Sie den Dokumenttyp auf der Seite **Datei** ändern.

Seite XML

Diese Seite ist nur für XML Exporte verfügbar. Hier konfigurieren Sie den Inhalt der exportierten XML Datei.



DTD

Beim Export im XML Format ermöglicht 4D die Option mit oder ohne Erstellen einer Document Type Declaration (DTD). Eine DTD speichert den notwendigen Satz spezifischer Regeln und Eigenschaften für XML. Sie definieren insbesondere den Namen und den Inhalt jedes Tags sowie dessen Kontext. Über diese Formatierung lässt sich prüfen, ob ein XML Dokument den Regeln entspricht. In diesem Fall wird es als gültig (valid) deklariert. Das ist besonders bei wiederkehrenden Tags in einem XML Dokument hilfreich. Beachten Sie, dass DTD nicht zwingend ist. Zum Verwalten von DTD müssen Sie eine der folgenden Optionen auswählen:

- **Kein DTD** (Standardoption): Diese Option erzeugt während dem Export kein DTD. Das erstellte XML Dokument kann nicht bestätigt werden.
- **Erzeuge DTD in XML Dokument**: Diese Option erzeugt während dem Export ein DTD, das direkt in die XML Datei eingebunden wird (interne DTD). Die generierte XML Datei ist demnach unabhängig.
- **Erzeuge DTD als eigenes Dokument**: Die DTD wird als separate Datei generiert (externe DTD). Eine externe DTD steht mehreren Benutzern zur Verfügung und sorgt so für eine einheitliche Strukturierung der XML Dokumente aus unterschiedlichen Quellen.
Hinweis: 4D erlaubt, dass Tabelle und Feld denselben Namen haben. Die XML Sprache dagegen verbietet, dass unterschiedliche Elemente denselben Namen haben. Von daher dürfen 4D Daten, die mit der Option "Erzeuge DTD" exportiert werden, keine gleichnamigen Tabellen und Felder enthalten. Dann ist die XML Datei ungültig und lässt sich nicht mit einem XML Parser öffnen.
- **Existierende DTD**: Diese Option ermöglicht in Verbindung mit der Schaltfläche[...], eine vorhandene externe DTD Datei zuzuweisen. 4D integriert dann in der exportierten Datei eine Referenz auf diese DTD.

XSL

Über XSL (eXtensible Stylesheet Language) lassen sich die im XML Dokument definierten Elemente in ansprechender Form darstellen. Über diese Sprache können Sie Vorlagen (style sheets) zum Bearbeiten und Anzeigen des Inhalts eines XSL Dokuments definieren. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **XSL Transformation**.

- **Auf XSL verweisen**: Über die Schaltfläche [...] können Sie der exportierten Datei eine XSL Datei zuweisen.

Wurde eine XSL Datei gewählt, müssen Sie definieren, wie sie mit exportierten Daten verwendet wird. Es gibt zwei Optionen:

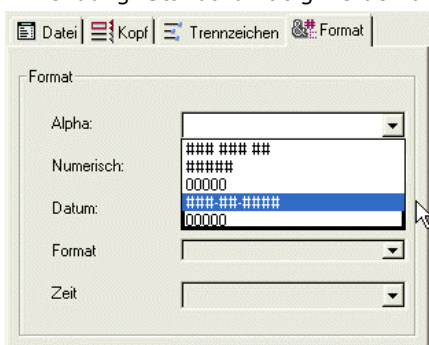
- **XSL Transformation anwenden**: Mit dieser Option führt 4D die Umwandlung der XSL Daten direkt beim Export im XSL Dokument durch.
- **Verknüpfung zu XSL Datei einfügen**: Mit dieser Option fügt 4D nur die Referenz auf das angegebene XSL im exportierten Dokument ein, d.h. die Umwandlung wird nicht ausgeführt. Die exportierte XML Datei enthält deshalb Referenzen auf diese XSL Datei.

XML Optionen

- **Leerzeichen ersetzen**: Diese Option ersetzt in Namensfeldern der generierten XML Datei Leerzeichen durch Unterstrich ("_"). Diese Option ist standardmäßig markiert, da Leerzeichen in Feldern mit XML Wert nicht erlaubt sind. Sie können diese Option für spezifische Fälle deaktivieren. Bedenken Sie jedoch, dass die generierte Datei dann nicht konform mit den allgemeinen Syntaxregeln ist, wie Sie von W3C für XML definiert wurden.
- **Binäre Felder als Base64 codieren**: Diese Option fügt den Kopfteil "data;base64," zu exportierten binären Feldern (Felder vom Typ BLOB bzw. Bild) hinzu. Ist diese Option nicht markiert, werden Felder in Base64 codiert, jedoch ohne den Kopfteil.
- **Mit Einrückungen**: Diese Option wendet auf die exportierten Daten automatisches Einrücken an. Auf diese Weise lässt sich die Hierarchie der XML Elemente anzeigen.
- **Bilder als PNG codieren**: Diese Option codiert die exportierten Bilder automatisch im PNG Format, unabhängig vom Originalformat. Ist diese Option nicht markiert, werden die Bilder in ihrem nativen Format codiert. Bei Bildern, die in SVG vorliegen, sollten Sie diese Option nicht markieren, damit die Bilder ihre Eigenschaften beibehalten.

Seite Format

Diese Seite ermöglicht, Exportformate einzurichten. Sie ist für alle Dateiformate verfügbar, mit Ausnahme von 4D Anwendung. Standardmäßig werden die Standardformate von 4D verwendet:



Hinweis: Das Exportformat für eine Spalte können Sie auch über das Kontext-Menü des Spaltentitels setzen (siehe Schritt 5 unter **Daten für den Export festlegen**).

Sie können ein Exportformat für Felder vom Typ alphanumerisch, Zahl, Datum, Zeit oder Boolean setzen. Wählen Sie in der Vorschau eine Spalte aus, erscheint auf dem Bildschirm eine Combo-Box, die diesem Datentyp entspricht. Wählen Sie ein passendes Format (für Daten vom Typ alphanumerisch, Zahl oder Boolean) oder geben Sie ein eigenes Format ein. Weitere Informationen dazu finden Sie im Abschnitt **Anzeigeformate**.

- **Format ISO Datum Zeit**

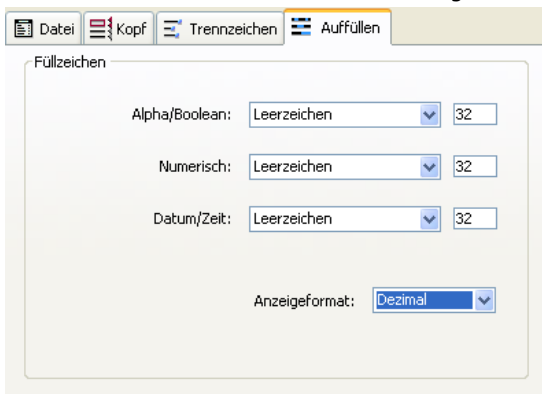
Das Format ISO Datum Zeit ist zum Export von Datum im Typ Datum oder Zeit verfügbar. Es entspricht der Standarddarstellung für Datum/Zeit in XML (ISO8601). Das Datum 31. Mai 2008 um 1:20 p.m. erscheint im ISO Format als 2008-05-31T13:20:00.

4D erlaubt nicht, dass Datum und Zeit zusammen in einem Feld gespeichert werden. Sie können jedoch Daten in

diesem Format exportieren, so dass Datum und Zeit in Übereinstimmung mit den XML Standards gespeichert werden. Beim Export hat das Datum die Form: 2008-05-31T00:00:00; Zeit die Form: 0000-00-00T13:20:55.

Seite Auffüllen

Diese Seite erscheint nur, wenn für das Dateiformat "Text fester Länge" ausgewählt ist. Damit definieren Sie je nach Art der Werte die Füllzeichen. Standardmäßig ist dafür das Leerzeichen definiert.



Weitere Informationen dazu finden Sie im Abschnitt [Dateiformate](#).

Für folgende Datentypen können Sie Füllzeichen verwenden: **Alpha/Boolean** (Felder vom Typ Boolean werden als Alpha importiert), **Numerisch** und **Datum/Zeit**. Das Leerzeichen dient normalerweise zum Füllen von festen Formatfeldern. Die DropDown Menüs bieten weitere Werte an. Im Eingabebereich rechts daneben erscheinen die ausgewählten Zeichen. Wollen Sie eigene Zeichen einsetzen, geben Sie diese direkt im jeweiligen Eingabebereich ein.

Über das DropDown Menü Anzeigeformat lassen sich die Füllzeichen in verschiedenen Formaten anzeigen: **Zeichen**, **Dezimal** (Standardformat) und **Hexadezimal**. Ein nicht-druckbares Füllzeichen erscheint nicht, wenn Sie das Menü **Zeichenformat** verwenden.

Daten mit Formular exportieren

Wollen Sie für den Export ein Formular verwenden, klicken Sie im rechten Teil auf die Registerkarte **Formular**. Nun können Sie für die ausgewählte Tabelle ein Formular aus der Formularliste wählen.

Der Vorteil eines Formulars ist, dass die zugeordneten Formular- oder Objektmethoden bei jedem Datensatz ausgeführt werden, der exportiert wird. So können Sie die Daten beim Export steuern. Felder und Variablen werden in der Eingabereihenfolge des Formulars exportiert. Beachten Sie jedoch, dass die Auswirkung dieser Methoden nicht in der Vorschau angezeigt wird.

Methoden werden im Formularereignis [On Load](#) ausgeführt. Setzen Sie keine Schaltflächen im Formular. Objekte in Unterformularen werden nicht berücksichtigt.

Hinweis für 4D Server: In Serverprozeduren lassen sich keine Daten über Formulare mit Methoden exportieren, da auf dem Server keine Formularereignisse verwaltet werden. Folglich werden die zugewiesenen Methoden nicht aufgerufen.

Import und Export über ODBC Datenquelle

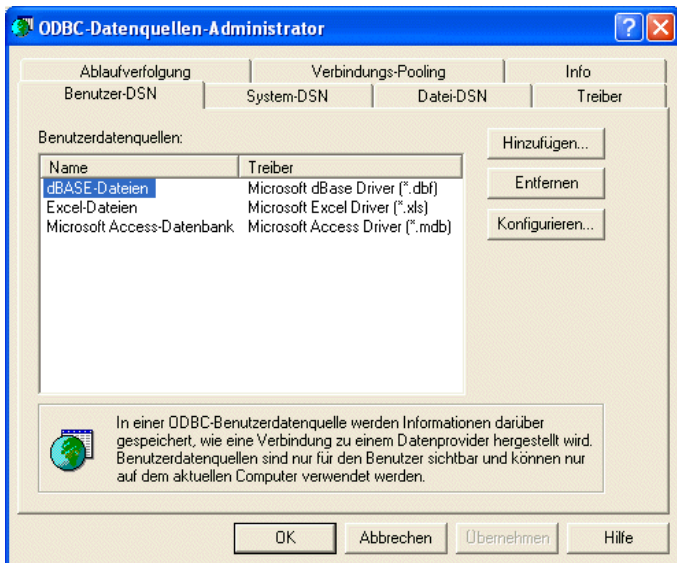
Sie können Daten in eine ODBC Datenquelle importieren oder aus dieser exportieren.

Das ODBC Protokoll (Open DataBase Connectivity) ist eine Schnittstelle, über die unterschiedliche Programme per SQL-Sprache miteinander kommunizieren können. Eine ODBC Datenquelle ist ein ständig geöffnetes "Fenster" zum Dokument eines anderen Programms, selbst wenn dieses nicht geöffnet ist. Viele Programme (Datenbanken, Tabellenkalkulationen, Datei-Manager, o.ä.) ermöglichen über einen spezifischen ODBC-Treiber, Daten als eine ODBC-Quelle zu veröffentlichen.

Jedes Programm, welches das ODBC-Protokoll unterstützt, kann Daten in einer ODBC-Quelle lesen oder schreiben. Die Importfunktion von 4D ermöglicht, Daten aus einer ODBC-Quelle zu erhalten. Die Exportfunktion ermöglicht, Daten in eine ODBC-Quelle einzufügen.

ODBC Quelle definieren

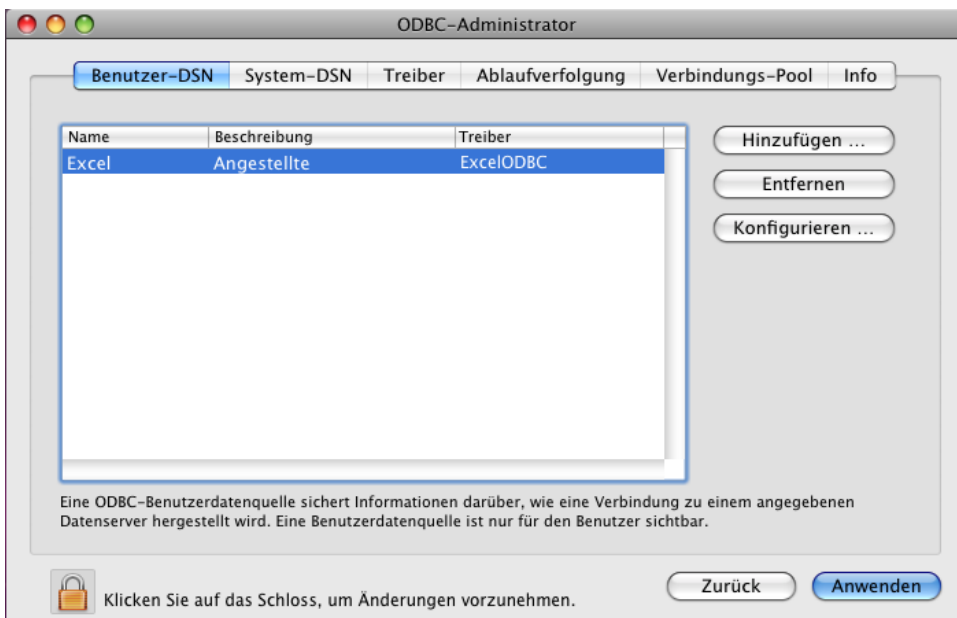
Unter Windows wird die ODBC-Quelle im Dialogfenster **Administrative Tools** und dem Konfigurationsfenster **ODBC Datenquellen Administrator** definiert:



Auf Mac OS X wird die ODBC-Quelle über das Icon **ODBC Administrator** im Ordner **Utilities** definiert:



ODBC Administrator



Weitere Informationen zu diesen Dialogfenstern finden Sie in der Dokumentation des jeweiligen Betriebssystems.

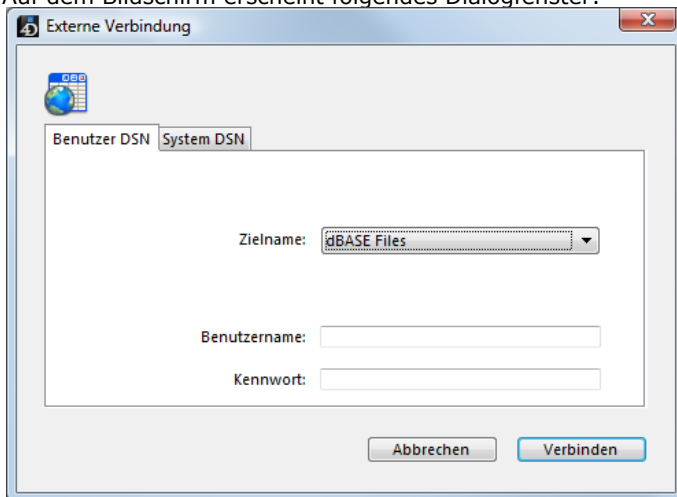
Daten von ODBC Quelle importieren

Sie können Daten aus einer ODBC Quelle auf dieselbe Weise wie andere Datentypen importieren (siehe **Daten aus Datei importieren**). Hier erscheint lediglich noch ein Zwischendialog zum Auswählen der ODBC Quelle.

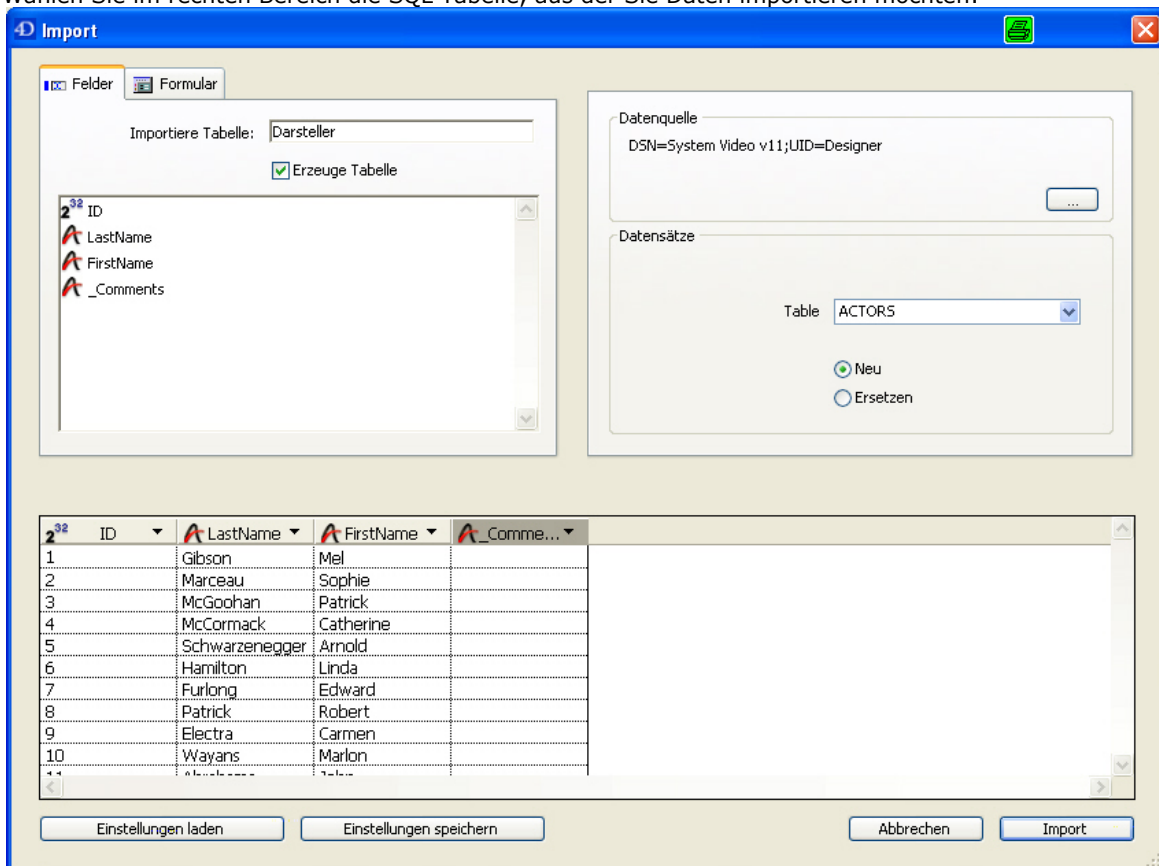
In der Anwendungsumgebung importieren Sie Daten über den Befehl **IMPORT ODBC**, in der Designumgebung über das Menü **Datei/Ablage**.

Um Daten aus einer ODBC Quelle zu importieren:

1. Wählen Sie im Menü **Datei/Ablage** den Befehl **Importieren>Von ODBC-Quelle**. Auf dem Bildschirm erscheint folgendes Dialogfenster:



2. Wählen Sie die zu importierende Datei, tragen bei Bedarf Benutzername und Kennwort ein und klicken auf die Schaltfläche **Verbindung**. Auf dem Bildschirm erscheint der Importeditor.
3. Wählen Sie im rechten Bereich die SQL Tabelle, aus der Sie Daten importieren möchten:



4. Wählen Sie die Tabelle und 4D Felder zum Importieren der Daten oder markieren Sie die Option **Erzeuge Tabelle** (ist für kompilierte Datenbanken nicht verfügbar). In der Vorschau erscheinen die Treffer zwischen importierten Daten und Feldern. Sie können auch auf die Registerkarte **Formular** klicken und Daten über ein Formular importieren. Weitere Informationen dazu finden Sie im Abschnitt **Daten aus Datei importieren**.
5. Wählen Sie im rechten Bereich die Importoptionen:
 - **Datenquelle**: Dieser Bereich beschreibt die Datenquelle, mit der die Datenbank verbunden ist. Um erneut den Auswahldialog für Datenquellen anzuzeigen, klicken Sie auf die Schaltfläche [...].
 - **Datensätze**: Sie können die Datensätze der Importtabelle mit Daten aus der Datenquelle ersetzen oder hinzufügen.
6. Klicken Sie am unteren Rand auf die Schaltfläche **Import**. Sobald die Daten importiert sind, schließt die Verbindung zwischen 4D und der Datenquelle automatisch.

Einstellungen für Import-Export speichern und laden

Im Dialogfenster Import/Export können Sie Ihre Einstellungen in eigenen Dateien auf der Festplatte sichern und erneut laden.

Sie können diese Operation in den Import/Export Dialogfenstern für Dateien und ODBC Quellen ausführen. Jedoch sind im ODBC Import/Exportdialog gesicherte Projekte nicht kompatibel mit dem standardmäßigen Import/Exportdialog und umgekehrt.

Um die Einstellungen für den Import/Export zu sichern bzw. zu laden, klicken Sie unter der Vorschau auf die entsprechende Schaltfläche:





Unter Windows haben die Dateien mit den Einstellungen für Import /Export die Endung ".4SI", auf Macintosh die Endung .4DSI. Diese Dateien speichern alle Einstellungen, die im Import-/ Exporteditor definiert wurden:


- Dateiname und Zugriffspfad der Datei oder ODBC Quelle
- Die ausgewählten Tabellen und Felder sowie Name des Formulars, sofern es ausgewählt wurde
- Die Optionen für Import-/Export, wie Dateityp, Trennzeichen, etc.

Auf diese Weise können Sie den Datenimport bzw. -export automatisieren. Das ist besonders hilfreich mit den 4D Befehlen **IMPORT ODBC** und **EXPORT ODBC**.

Web Services publizieren und einsetzen

 Einführung

 Web Service mit 4D veröffentlichen

 Web Service in 4D abonnieren

Was sind Web Services?

Ein Web Service ist ein Satz von Funktionen, die als Einheit zusammengefasst sind und in einem Netzwerk veröffentlicht werden. Jede Anwendung, die mit Web Services kompatibel ist und an dasselbe Netzwerk angeschlossen ist, kann diese Funktionen aufrufen. Web Services eignen sich hervorragend im Kontext zum Veröffentlichen im Internet.

Web Services können ganz verschiedene Aufgaben durchführen, z.B. Bücher bestellen, E-Commerce abwickeln, Marktwerte überwachen, usw.

Das Programm, welches die Services (Dienste) veröffentlicht, heißt Web-Service Server, SOAP Server oder in Kurzform „Server“. Jede mit Web Services kompatible Anwendung kann eine oder mehrere dieser Funktionen verwenden; das ist das „Client“ Programm (Web Service Client, SOAP Client).

Der Vorteil von Web Services ist ihre Interoperabilität mit verschiedenen Informationssystemen, d.h. Server- und Client-Programme müssen nicht zueinander kompatibel sein, damit das System arbeitet. Von der Client-Seite aus gesehen ist ein Web Service eine „Black Box“, an die sowohl Werte gesendet als auch Werte als Bearbeitungsergebnis zurückgegeben werden.

Die vom Server angebotenen Web Services können öffentlich oder privat sein. Es gibt eine Vielzahl öffentlicher Web Services im Internet, die jede Anwendung kostenlos abrufen kann.

Web Services werden von der W3C (World Wide Web Consortium, die Regulierungsbehörde für das Internet) und großen Unternehmen der Computer-Industrie unterhalten und sind eine zuverlässige, dauerhafte und aktualisierbare Connectivity-Lösung.

Funktionsweise von Web Services – Hauptdefinitionen

Web Services arbeiten hauptsächlich mit dem HTTP Transport Protokoll.

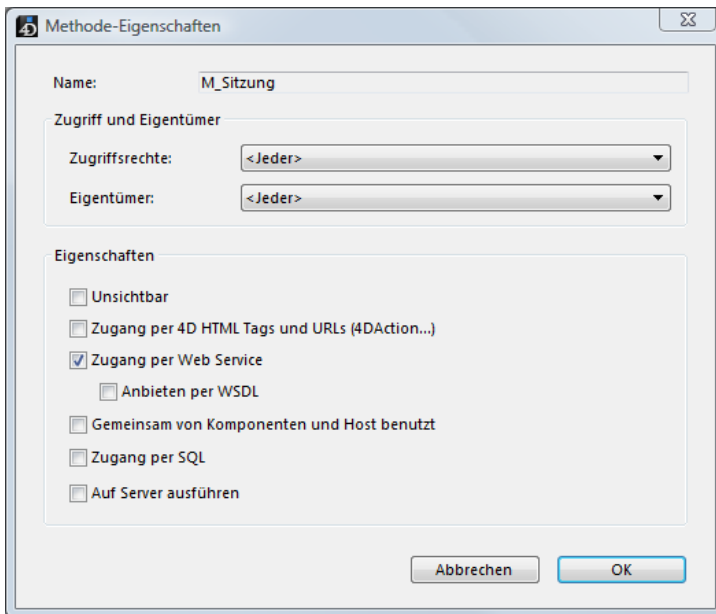
- **SOAP:** Web Services verwenden ein „offenes“ leistungsstarkes Kommunikationsprotokoll mit Namen SOAP (Simple Object Access Protocol). Es basiert vollständig auf der XML Sprache, sowohl auf Ebene der Nachrichtenstruktur (envelope) als auch auf Ebene der ausgetauschten Daten. Die Operation dieses Protokolls wird über RFC (Request for Comment) definiert. Das sind Dokumente, welche die verschiedenen Aspekte des Internet standardisieren. Von daher verfügt SOAP über eine breit gestreute Kompatibilität.
Ein Web Service funktioniert folgendermaßen: Ein Web Service Client sendet über ein SOAP Protokoll eine Anfrage in XML-Format an den Server. Der Server analysiert die Anfrage, führt die angeforderte Operation durch und gibt über dasselbe Protokoll und dieselbe Sprache die Antwort zurück.
- **WSDL:** Die Server von Web Services veröffentlichen in der Regel eine WSDL Datei (Web Service Description Language), um die Zugriffsspezifikationen für die angebotenen Dienste zu definieren. Über die WSDL Datei können Server von Web Services die „Bearbeitungsanweisungen“ der angebotenen Dienste (URLs, Methodenlisten, Parameter, etc.) veröffentlichen. Das geschieht in Form einer XML Datei, die die Server-Anwendung in der Regel selbst erstellt. Diese Datei ist nicht zwingend.
- **UDDI:** (Universal Description Discovery and Integration) ist eine weltweite Datenbank, die alle öffentlichen Web Services inventarisiert. Beachten Sie, dass ein Web Service nicht zwingenderweise öffentlich gemacht werden muss. Das ist in den meisten Fällen auch nicht notwendig.

Integration von Web Services in 4D

4D lässt sich als Web Service Server bzw. Client verwenden. Die Integration in 4D ist einfach und sicher: Verschiedene Konfigurationen ermöglichen, die Bedingungen zum Veröffentlichen und Abonnieren präzise zu überwachen.

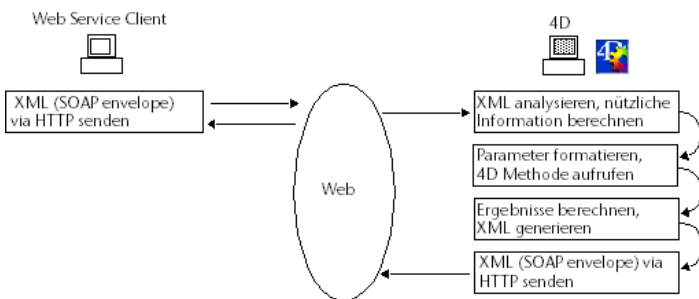
4D als Web Services Server

Sie können jede eigene Projektmethode ohne größere Änderungen als ein Web Service veröffentlichen. Das können Sie als Eigenschaft der Methode festlegen:



Hinweis: Die Veröffentlichung von Web Services mit 4D erfordert eine spezifische Lizenz: die SOAP Lizenz. Der 4D Web Server verwaltet automatisch die Veröffentlichung und Pflege der WSDL Datei. 4D analysiert (parsing) den XML Inhalt der Anfrage, formatiert die Parameter, sendet die Ergebnisse, etc.. Dafür ist keine spezifische Programmierung erforderlich.

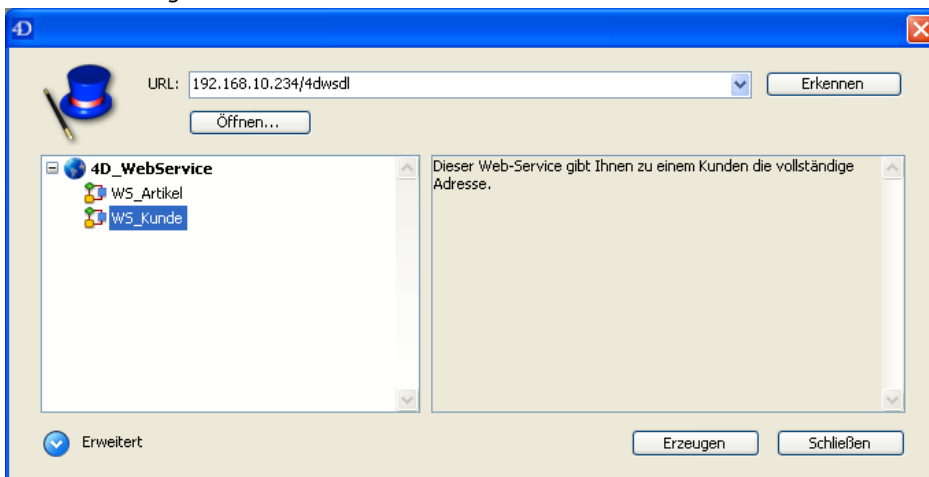
Bearbeitung von SOAP Anfragen mit 4D



Sie können jedoch mit Hilfe spezifischer 4D Befehle die Bearbeitung einer Anfrage an Ihre Bedürfnisse anpassen. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Web Services (Server)**.

4D als Web Services Client

Ihre Datenbanken können jeden Typ von Web Service verwenden, der im Internet oder in Ihrem Netzwerk angeboten wird. In den meisten Fällen sorgt der Assistent für Web Services auf Client-Seite dafür, dass Sie jeden Web Service mit minimalem Programmieraufwand rasch benutzen können:



Einen Web Service in 4D nutzen bedeutet, über Proxy Methoden Parameter über das Netzwerk senden und die Antwort auswerten. Das Erstellen und Ausführen von Proxy Methoden, die Web Services aufrufen, ist vollkommen automatisch und erfordert keine Programmierung. Sie müssen diese Methoden lediglich in Ihrem Code aufrufen.

Sie können diese Methoden, wie auf der Server-Seite, über die Befehle der 4D Programmiersprache an Ihre Bedürfnisse anpassen (siehe **Web Services (Client)**).

Sicherheit von Web Services

Von 4D veröffentlichte Web Services übernehmen die Sicherheitsmechanismen, die für den 4D Web Server gelten. Web Services profitieren also von denselben Konfigurationen wie konventionelle Web Anfragen: Kennwörter, Datenbankmethoden **On Web Authentication** und **On Web Connection**, TLS Protokoll, etc.

Darüberhinaus ermöglichen spezifische 4D Funktionen, wie **SOAP Get info** und **SOAP request** die Veröffentlichung von Web Services präzise zu steuern.

Auf Client-Seite ist die Verbindung an Server mit Web Services über den gesicherten Modus mit TLS möglich. Der 4D Befehl **WEB SERVICE AUTHENTICATE** ermöglicht auch die Verbindung mit Servern bzw. Proxies, für die Authentifizierung erforderlich ist.

Kompatibilität von RPC, DOC und komplexen Typen

Die Kommunikationsschicht von Web Services (Transport gewährleisten, Dienste aufrufen und Sicherheit bei der Übertragung) kann in zwei unterschiedlichen Modi operieren: RPC (Remote Procedure Call) und DOC (Message/Document). Sie unterscheiden sich in der Konstruktionsebene der Anfragen und Antworten für Server und Client. Für bestimmte Client-Anwendungen ist der Modus DOC erforderlich.

4D unterstützt den RPC und den DOC Modus:

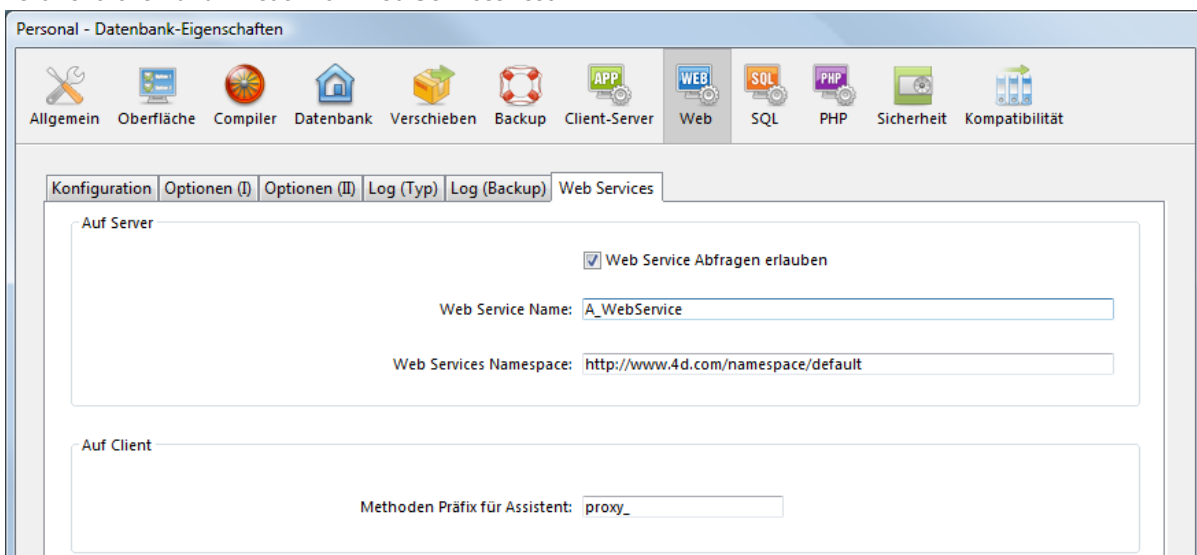
- Client-Seite: Die Unterstützung läuft unsichtbar über den Web Services Assistenten. Der erzeugte Code wird automatisch an den Publikationsmodus angepasst.
- -Seite: Die Unterstützung läuft ebenfalls unsichtbar. Die Methoden werden automatisch als Web Services im RPC- und DOC-Modus veröffentlicht. Die Wahl des Modus richtet sich nach der URL des WSDL. Weitere Informationen siehe Abschnitt **WSDL erzeugen**.
 - Um ein 4D Web Service im RPC Modus zu veröffentlichen, verwenden Sie folgende URL für das WSDL:
http://ServerAddress/4DWSDL
Der 4D Server verwaltet das Bearbeiten der Anfragen und Antworten in RPC. RPC Anfragen werden automatisch an folgende Adresse gesendet: http://ServerAddress/4DSOAP.
 - Um ein 4D Web Service im DOC Modus zu veröffentlichen, verwenden Sie folgende URL für das WSDL:
http://ServerAddress/4DWSDL/DOC
Der 4D Server verwaltet das Bearbeiten der Anfragen und Antworten in DOC. DOC Anfragen werden automatisch an folgende Adresse gesendet: http://ServerAddress/4DSOAP/DOC

Über das SOAP Protokoll werden zwei unterschiedliche Arten von XML Daten ausgetauscht: **einfache** Typen und **komplexe** Typen. Für Daten von Web Services, die im RPC Modus veröffentlicht werden, sind beide Typen möglich. Dagegen sind Daten von Web Services, die im DOC Modus publiziert werden, systematisch im komplexen Typ.

Beachten Sie, dass komplexe Datentypen in XML nicht direkt in einer 4D Datenbank einsetzbar sind. Dazu ist eine spezifische Vorgehensweise notwendig. In den meisten Fällen übernimmt der Assistent für Web Services die Ausführung. In manchen Fällen muss die Bearbeitung jedoch durch 4D XML Programmierbefehle ergänzt werden. Weitere Informationen dazu finden Sie im Abschnitt **Komplexe Typen**.

Konfiguration

Im Dialogfenster Datenbank-Eigenschaften legen Sie auf der Seite Web>Web Services die allgemeinen Parameter zum Veröffentlichen und Einsatz von Web Services fest:



Die Aktion dieser Parameter wird auf den folgenden Seiten beschrieben. Die ausführliche Beschreibung jeder Option finden Sie auf der [Seite Web/ Web Services](#).

Web Service mit 4D veröffentlichen

Die Veröffentlichung eines Web Service in 4D erfolgt in der Regel in drei Stufen:

1. Methode zum Veröffentlichen erstellen
2. Veröffentlichung (WSDL) konfigurieren
3. Publikation

Sie können bei Bedarf weitere eigene Stufen einbauen.

Web Service Methode erstellen

Sie können jede Art von Projektmethode erstellen, die zur Veröffentlichung als Web Service dienen soll. Sie muss Parameter akzeptieren und ein Ergebnis zurückgeben. Diese Parameter müssen im Kopfteil der Methode mit 4D Befehlen aus dem Kapitel **Compiler** deklariert werden.

4D formatiert standardmäßig die Parameter zum Ausführen von Methoden, die als Web Service veröffentlicht werden. Sie können diese Parameter bei Bedarf über den 4D Befehl **SOAP DECLARATION** verändern.

4D erledigt automatisch die Umwandlung der empfangenen und via SOAP gesendeten Daten.

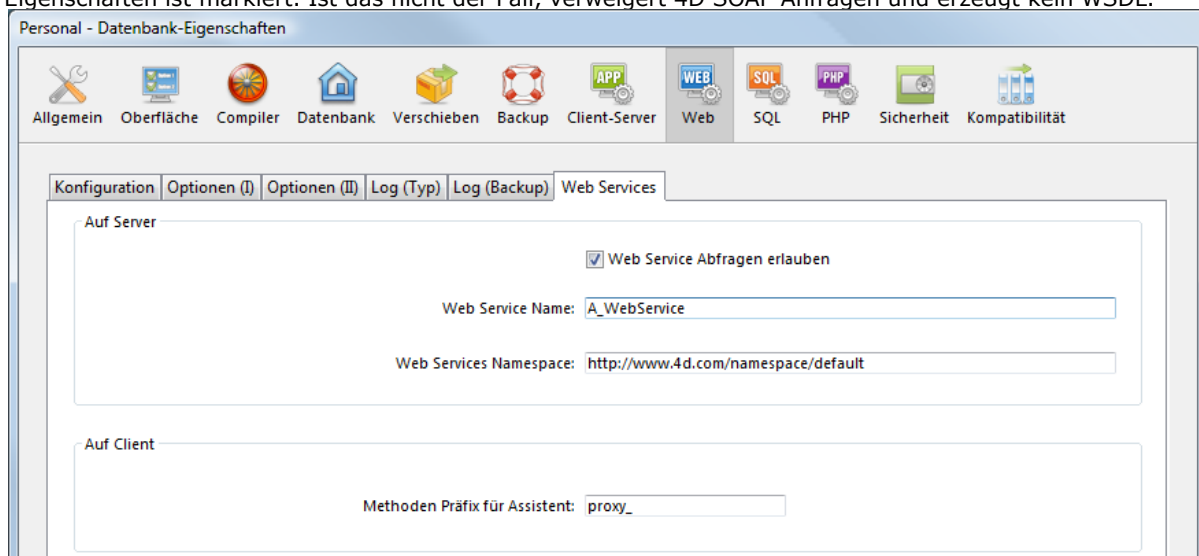
Warnung: Die Methodennamen werden in SOAP Anfragen als XML Tags verwendet. Sie dürfen daher, in Übereinstimmung mit dem XML Standard für Tag-Namen weder Leerzeichen noch Zeichen aus dem erweiterten Ascii-Satz enthalten. Nur der Latin Zeichensatz ist zulässig ([A-Za-z0-9._] | '-!)*.

Zur Entwicklung und Überwachung einer Methode, die als Web Service veröffentlicht wird, verwenden Sie 4D Befehle aus dem Kapitel **Web Services (Server)**.

Methoden publizieren

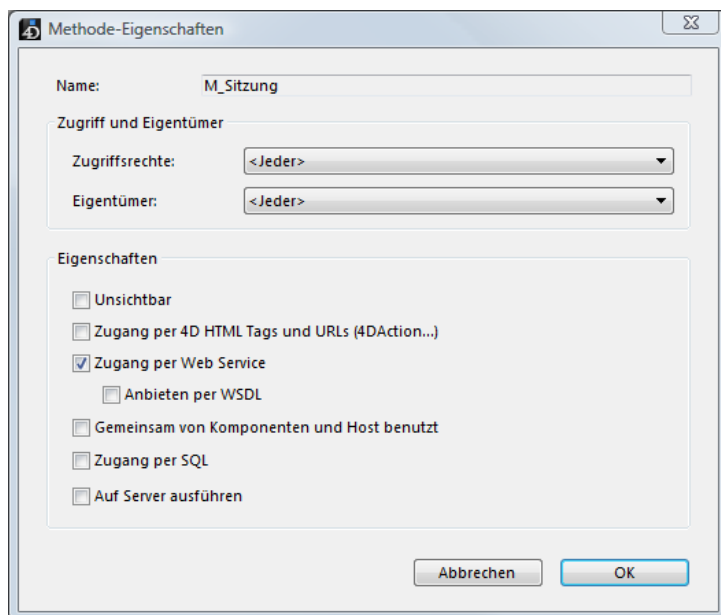
Um Methoden Ihrer Datenbank als Web Service zu publizieren, müssen nachfolgende Voraussetzungen erfüllt sein.

- Der Rechner, welcher als SOAP Server dient (4D Server, 4D Client oder 4D Einzelplatz), verfügt über eine 4D SOAP Lizenz.
- Der 4D Web Server ist gestartet.
- Die Option **Web Service Abfragen erlauben** auf der Seite "Web Services" im Dialogfenster Datenbank-Eigenschaften ist markiert. Ist das nicht der Fall, verweigert 4D SOAP Anfragen und erzeugt kein WSDL.



Ist diese Option markiert, erstellt 4D die WSDL Datei. Weitere Informationen dazu finden Sie im Abschnitt **WSDL erzeugen**.



- Jede Methode, die veröffentlicht werden soll, muss die Eigenschaft **Zugang per Web Service** haben. Das legen Sie im Dialogfenster Methode-Eigenschaften fest:



Hinweis: Weitere Informationen zu diesem Dialogfenster finden Sie im Abschnitt **Eigenschaften für Projektmethoden**.

Mit der Option **Zugang per Web Service** lässt sich die dazugehörige Methode über eine SOAP Anfrage als Web Service aufrufen.

Hinweis: Ist außerdem die Option **Anbieten per WSDL** markiert, erscheint die Methode in der WSDL des Servers (siehe nächster Absatz).

Auf der **Seite Methoden** von 4D Explorer zeigen spezifische Icons an, dass die Methode als Web Service angeboten  und zusätzlich auch in der WSDL Datei veröffentlicht wird .

WSDL erzeugen


WSDL beschreibt in XML Sprache die Syntax und die benötigte Information zum Aufrufen der 4D Methode (Methodenname, URL, Parameter, etc.).

In 4D entspricht WSDL einem einzelnen Web Service. Dieser definiert die Methoden und ihre Parameter und kann an einer bestimmten Stelle konsultiert werden. In 4D ist WSDL keine echte "Datei", sondern eine URL mit Namen **4DWSDL** für Web Services, die im Modus RPC veröffentlicht werden, mit Namen **4DWSDL/DOC** für Web Services, die im Modus DOC veröffentlicht werden. Sie liegt immer im Root des Web Servers. Sie existiert nur im Speicher und wird nicht auf die Festplatte geschrieben.

Lautet die Adresse Ihres Web Server z.B. <http://www.myserver.com>, können Sie WSDL aufrufen über die URL:

- <http://www.myserver.com/4DWSDL> (RPC Modus)
- <http://www.myserver.com/4DWSDL/DOC> (DOC Modus)

Um eine Methode in WSDL hinzufügen oder zu entfernen, müssen Sie lediglich die entsprechende Option in den Methodeigenschaften aktivieren bzw. deaktivieren. 4D aktualisiert sofort den Inhalt von WSDL.

Hinweis: In 4D Explorer zeigt ein spezifisches Icon  an, dass die Methode in WSDL publiziert wird.

Sind Web Services zugelassen, erzeugt 4D automatisch und dynamisch die WSDL des 4D Web Server, wenn mindestens für eine Methode die Eigenschaft **Anbieten per WSDL** markiert ist. Diese Option ist standardmäßig inaktiv.

Name des Web Service anpassen

Jeder im Internet veröffentlichte Web Service hat einen Namen. Er dient zur Unterscheidung der Dienste auf Ebene des SOAP Server (wenn der Server mehrere Web Services veröffentlicht) und in den Verzeichnissen der Web Services.

4D verwendet standardmäßig den Namen *A_WebService*. Sie können ihn in den Datenbank-Eigenschaften auf der Seite **Web>Web Services** verändern.

Warnung: In Übereinstimmung mit dem XML Standard für Tag-Namen dürfen die Zeichenfolgen weder mit einer Zahl beginnen noch Leerzeichen enthalten. Sie sollten auch keine Zeichen aus dem erweiterten Ascii-Satz enthalten, z.B. Akzente.

Namensbereich anpassen

Jeder im Internet veröffentlichte Web Service muss einmalig sein. Das wird über die XML Namensbereiche sichergestellt. Ein Namensbereich ist eine willkürliche Zeichenkette, um einen Satz XML Tags als einmalig zu identifizieren. In der Regel beginnt ein Namensbereich mit der URL der Firma, also <http://mycompany.com/mynamespace>. Die URL muss in diesem Fall weiter nichts besonderes enthalten; wichtig ist, dass die verwendete Zeichenfolge einmalig ist.

4D verwendet standardmäßig folgenden Namen: **<http://www.4d.com/namespace/default>**. Sie können ihn in den Datenbank-Eigenschaften auf der Seite **Web>Web Services** verändern.

Kommentare zu veröffentlichten Methoden

Kommentare zu Methoden, die als Web Service angeboten und in WSDL veröffentlicht werden, erscheinen in dieser Datei automatisch als Feld "Dokumentation".

Diese Kommentare müssen Sie auf der **Seite Methoden** des Explorers eingeben (siehe Abschnitt **Kommentare verwenden**).

Dieser Mechanismus dient zum Beschreiben oder Dokumentieren der veröffentlichten Methoden. Die Interpretation und Verwaltung dieses Felds richtet sich nach der Einbettung des Web Services auf Client-Seite.

Auf mit 4D veröffentlichten Web Service zugreifen

Sobald Ihr Web Service von 4D veröffentlicht wird, kann jede Client Anwendung, die Web Services unterstützt, sich an diesen anmelden. Der Zugriffsmodus und die Abarbeitung der Information, die mit den Web Service Server ausgetauscht wird, richtet sich nach der Client Anwendung, die diese Operation ausführt.

Alle für ein 4D Web Service benötigte Information, z.B die URL des Dienstes, die notwendigen Parameter, etc. werden in der WSDL von 4D veröffentlicht. Im Prinzip sollte die Nutzung eines Web Service mit dem Lesen der WSDL des SOAP Server beginnen, um diese Information ausfindig zu machen. In 4D lautet die URL der WSDL

http://ServerAddress/4DWSDL (RPC Modus) oder **http://ServerAddress/4DWSDL/DOC** (DOC Modus).

Dieser Schritt ist jedoch nicht zwingend. Die Verbindung zum SOAP Server kann auch direkt erfolgen.

Folgende Werte werden zum Erstellen von SOAP Anfragen benötigt:

- **URL Zugriff auf den von 4D veröffentlichten Web Service**
http://ServerAddress/4DSOAP/ (RPC mode) or
http://ServerAddress/4DSOAP/DOC (DOC mode)
(nicht anpassbar)
- **Web Service Name**
Standard: A_WebService
Anpassbar (siehe **Name des Web Service anpassen**).
- **Name der veröffentlichten Methode**
Name der 4D Projektmethode, die der Entwickler definiert hat (siehe **Web Service Methode erstellen**).
- **Parameter der Methode**
Die Parameter in der Methode müssen deklariert sein (definiert vom Entwickler).
Standardnamen für SOAP: FourD_arg0, FourD_arg1... FourD_argn
Anpassbare Namen über den Befehl **SOAP DECLARATION**.
- **Namensraum**
Standard: http://www.4d.com/namespace/default
Anpassbar (siehe **Namensbereich anpassen**).
- **Inhalt des Feldes SOAP Action**
ServiceName#MethodName (nicht anpassbar).

Web Service in 4D abonnieren

Mit 4D können Sie Web Services abonnieren, d.h. Sie können sie aufrufen und innerhalb Ihrer eigenen Anwendung nutzen.

Über Web Services können Sie mühelos verschiedene Funktionalitäten in Ihrer Software hinzufügen, z.B. Zugriff auf Lagerhaltung, Verfolgung von Paketzustellung, Ausführung komplexer Berechnungen. Sehr viele Dienste werden im Internet über Web Services angeboten.

Sie können auch Web Services abonnieren, die Sie selbst in anderen Anwendungen veröffentlicht haben. So können verschiedene 4D Datenbanken miteinander kommunizieren.

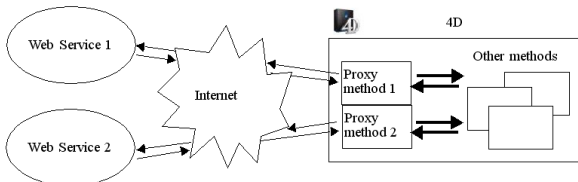
Vorgehensweise

Jede mit 4D erstellte Datenbank kann einen Web Service abonnieren, sie muss lediglich an ein Netzwerk bzw. an das Internet angeschlossen sein.

Um einen Web Service aufrufen zu können, müssen Sie nachfolgende Schritte ausführen:

1. Machen Sie die URL des Web Service ausfindig, den Sie abonnieren wollen.
Dazu können Sie Web Sites einsetzen, die im Internet veröffentlichte Web Services auflisten, z.B. www.xmethods.net oder Verzeichnisse wie UDDI. In den meisten Fällen müssen Sie die URL der WSDL Datei für den Web Service kennen.
Hinweis: kann Web Services verwenden, die im RPC oder DOC Modus veröffentlicht werden (siehe **Kompatibilität von RPC, DOC und komplexen Typen**)
2. Über den Assistenten für Web Services analysieren Sie den Inhalt der WSDL des betreffenden Web Service und generieren die dazugehörige Proxy Methode. Das ist die lokale Projektmethode, die den Web Service abfragt und die zurückgegebenen Werte herausfindet. Dieser Schritt wird im Absatz unten beschrieben.
Hinweise:
 - Im Assistenten für Web Services können Sie auch eine Proxy Methode ohne WSDL Datei erstellen. Dazu geben Sie einfach die betreffenden Parameter manuell ein.
 - Sie können eine Proxy Methode auch im Methodeneditor erstellen, ohne den Assistenten für Web Services zu verwenden (für fortgeschrittene Anwender).
3. Im Code Ihrer 4D Datenbank rufen Sie die Proxy Methode, so oft wie benötigt, durch Übergeben geeigneter Parameter auf.
Dieser Schritt wird im Absatz **Proxy Methode aufrufen**.

Die Proxy Methode verwaltet die Verbindung zum Web Service:



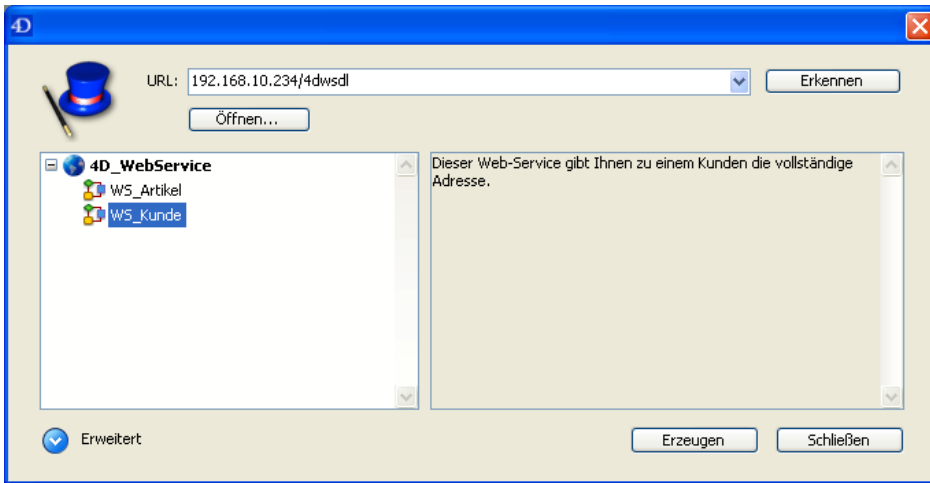
Assistent für Web Services verwenden

Die 4D Anwendung verwendet zum Abonnieren eines Web Service den Assistenten für Web Services. Er führt automatisch folgende Schritte aus:

- Analyse (parsen) der WSDL Dateien für die zu verwendenden Web Services
- Definition der Parameter für die zu erstellenden Proxy Methoden
- Erstellen der Proxy Methoden

Dialogfenster Assistent

Um den Assistenten für Web Services zu öffnen, wählen Sie im Menü **Design>Web Services Assistent**. Auf dem Bildschirm erscheint folgendes Dialogfenster:



Hinweis: Das Fenster erscheint auch, wenn Sie im Explorer auf der Seite **Methoden** am unteren Rand auf das Menü **Optionen** klicken. Weitere Informationen dazu finden Sie im Abschnitt).

Es ist in drei Bereiche gegliedert:

- Im Bereich "URL" geben Sie die URL der WSDL Datei für den gewählten Web Service ein oder wählen eine URL aus. Dieser Bereich ist eine Combo Box, welche die zuvor eingegebenen Werte in Form einer DropDown-Liste speichert.
- Der Hauptbereich zeigt das Ergebnis aus der Analyse des Inhalts in der WSDL Datei: Namen der Dienste und veröffentlichte Methoden.
- Der untere Bereich ist standardmäßig ausgeblendet. Über das Icon **Erweitert** können Sie ihn aufklappen. Er zeigt die Parameter zur Methode, die im Hauptbereich ausgewählt wurde.

Die Schaltfläche **Erkennen** durchsucht die angegebene WSDL Datei und setzt die Informationen in den darunterliegenden Bereich ein.

Die Schaltfläche **Öffnen** zeigt einen Standard-Öffnen Dialog, um eine lokal gespeicherte WSDL auszuwählen. Der Zugriffspfad wird dann im Bereich URL angezeigt. Er beginnt mit "Datei:///". Sie können ihn auch manuell eingeben.

Die Schaltfläche **Erzeugen** generiert die Proxy Methode, die zum gewählten Web Service gehört.

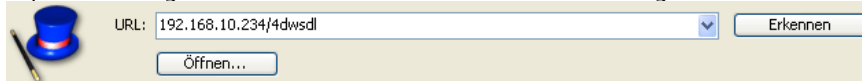
Die Schaltfläche **Schließen** schließt das Dialogfenster wieder.

WSDL und Proxy Methode (Standard Modus)

Die Hauptaufgabe des Assistenten für Web Services ist, eine WSDL Datei zu analysieren und die entsprechende(n) Proxy Methode(n) zu generieren. Diese Operation ist vollkommen automatisch und erfordert vom Benutzer weder Programmierung noch besonderes Know-how.

Um eine WSDL Datei zu analysieren und die Proxy Methode zu generieren:

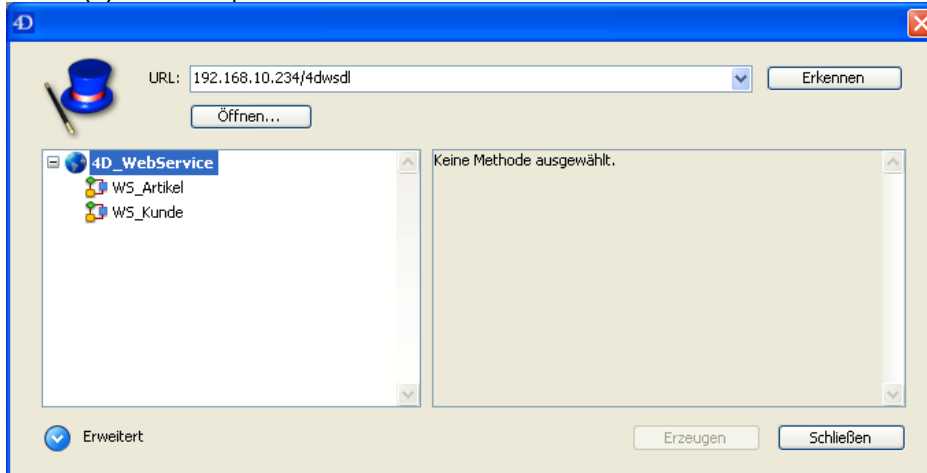
1. Kopieren oder geben Sie im Bereich "URL" die URL für den gewünschten Web Service ein:



Die URL kann z.B. aus einem Verzeichnis mit Web Services stammen oder direkt vom Server des Web Services mitgeteilt werden.

Sie können auch eine lokale URL angeben, z.B. die Adresse einer WSDL Datei, die auf Ihrer Festplatte gespeichert ist. Dazu klicken Sie auf die Schaltfläche **Öffnen** und wählen die lokale WSDL Datei aus oder geben den Zugriffspfad direkt ein. Der Zugriffspfad der lokalen Datei beginnt mit "Datei:/// " und verwendet dann den standardmäßigen Separator des Systemordners. Sie müssen einen absoluten Zugriffspfad übergeben.

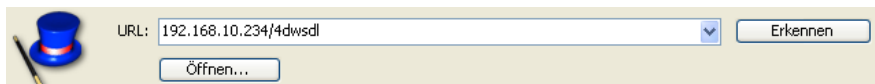
2. Klicken Sie auf die Schaltfläche **Erkennen**, um den Inhalt der WSDL Datei zu analysieren (parsen). Nach wenigen Augenblicken erscheint im Hauptbereich das Ergebnis der Durchsuchung: Der Name des/der Web Service(s) sowie die publizierten Methoden erscheinen in Form einer hierarchischen Liste.



Hinweis: Klicken Sie mit gedrückter **Umschalttaste** auf die Schaltfläche **Erkennen**, erscheint der XML Quellcode der WSDL Datei direkt in Ihrem Standard Web Browser.

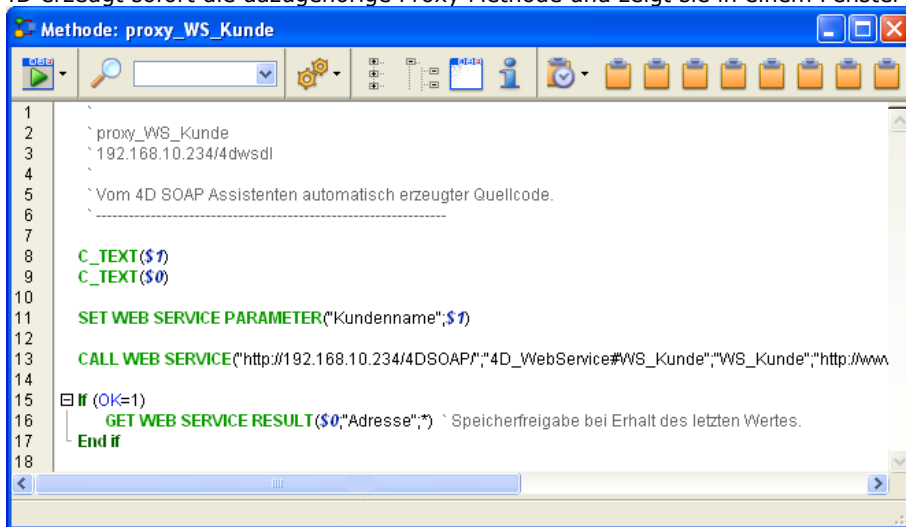
Klicken Sie auf einen Web Service, erscheint – sofern vorhanden – die dazugehörige Beschreibung neben der hierarchischen Liste. Ansonsten erscheint die Angabe "Keine Dokumentation".

Dasselbe gilt, wenn Sie einen Methodennamen wählen und dazu Dokumentation vorhanden ist:



Hinweis: Werden beim Durchsuchen der WSDL Datei Parameter für komplexe Typen festgestellt, zeigt der Assistent vor der betreffenden Methode eine gelbe Flagge .

- Wählen Sie die gewünschte Web Service Methode und klicken auf die Schaltfläche **Erzeugen**. 4D erzeugt sofort die dazugehörige Proxy Methode und zeigt sie in einem Fenster des 4D Methodeneditors an:



Der Name der Proxy Methode beginnt mit der Standardvorsilbe "proxy_" gefolgt vom Namen der Web Service Methode. Sie können die Standardvorsilbe in den Datenbank-Einstellungen auf der [Seite Web/ Web Services](#) ändern. Sie können den Namen auch nach Erstellen der Proxy Methode ändern. Das beeinflusst nicht die Ausführung der Methode.

Zusätzliche Parameter

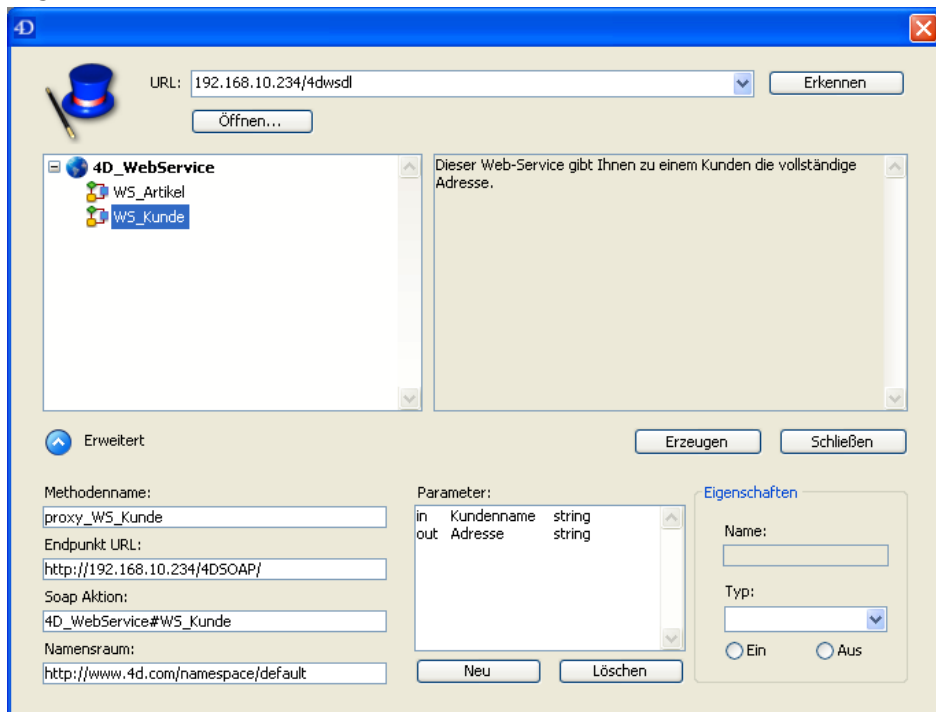
Die Proxy Methoden, die der Assistent für Web Services über Analysieren einer WSDL Datei erzeugt, sind sofort einsatzfähig, ohne irgendeine Änderung (Standardmodus). Sie können die Parameter dieser Proxy Methoden aber auch verändern, z.B. einen anderen Namen zuweisen.

Mit dem Assistenten für Web Services können Sie auch eine Proxy Methode erstellen, in die Sie die Parameter manuell eingegeben haben. In diesem Fall verwenden Sie nicht den WSDL Parser.

Eine Methode wird auch erstellt, wenn Sie nicht alle Parameter eingeben.

Sie können sogar eine Proxy Methode ohne Parameter erstellen, die als Vorlage dient und dann über 4D Programmierung gefüllt wird.

Für diese Operationen müssen Sie die zusätzlichen Parameter des Assistenten für Web Services verwenden. Klicken Sie dazu auf das Icon **Erweitert** unter der Liste, um das Fenster zu erweitern. Ist eine Methode ausgewählt, enthalten die Eingabefelder die aktuellen Parameter:



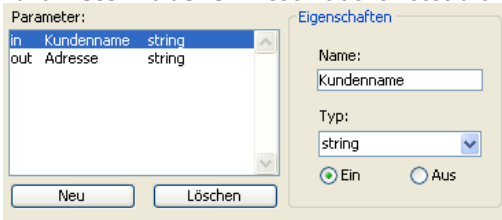
Sie können alle Parameter verändern. Behandeln Sie jedoch Parameter, die sich aus WSDL Parsing ergeben, mit Bedacht, denn das kann die Funktionalität des Web Service beeinflussen. Davon ausgenommen ist der Methodenname.

Es gibt folgende zusätzliche Parameter:

- Methodenname:** Name, den der Assistent der zu erstellenden Proxy Methode gibt. Er beginnt standardmäßig mit der Vorsilbe "proxy_". Das können Sie in den Voreinstellungen der 4D Anwendung verändern. Sie können den

Namen beliebig verändern, z.B. weil er schon in der Datenbank vorhanden ist. Er hat keinen Einfluss auf die Ausführung des Web Service.

- **Endpunkt URL:** URL, an welche die Proxy Methode die SOAP Anfragen sendet.
- **SOAP Aktion:** Inhalt des Feldes SOAPAction. Es enthält in der Regel den Wert "ServiceName#MethodName".
- **Namensraum:** Namensraum des Web Service. Weitere Informationen dazu finden Sie im Abschnitt **Namensbereich anpassen**).
- **Parameter Tabelle:** Diese Tabelle listet die Parameter der veröffentlichten Methode auf.



Jede Zeile steht für einen Parameter:

- Die erste Spalte zeigt an, ob der Parameter hereinkommt ("in") oder hinausgeht ("out"). Dieses Merkmal wird aus Sicht der Proxy Methode, nicht aus Sicht der veröffentlichten Methode bewertet.
- Die zweite Spalte gibt den Namen der Parameter an.
- Die dritte Spalte gibt den SOAP Typ der Parameter an. Verschiedene von 4D anerkannte Typen lassen sich im Menü Typ im Bereich Eigenschaften anzeigen. Der 4D Assistent für Web Services hat die Aufgabe, die SOAP Typen den entsprechenden 4D Typen in der Proxy Methode zuzuweisen.

Nachfolgende Tabelle gibt eine Übersicht:

SOAP Typ	Entsprechender 4D Typ
boolean	Boolean
int	Lange Ganzzahl
time	Zeit
float	Zahl
double	Zahl
date	Datum
string	Text
base64Binary	BLOB
ArrayOfBoolean	Array Boolean
ArrayOfInt	Array Lange Ganzzahl
ArrayOfTime	Array Lange Ganzzahl
ArrayOfFloat	Array Zahl
ArrayOfDate	Array Datum
ArrayOfString	Array Text
AsXML	BLOB

Hinweis: Der Typ AsXML ist streng genommen kein SOAP Typ, er dient zur Unterstützung komplexer XML Typen (siehe **Komplexe Typen**).

Der Bereich Eigenschaften zeigt die Wesensmerkmale der Parameter, die in der Tabelle gewählt wurden. Im Assistenten für 4D Web Services können Sie vorhandene Parameter verändern oder Parameter hinzufügen, z.B. wenn die angegebene WSDL Datei nicht mehr aktuell ist.

- Wollen Sie einen Parameter verändern, wählen Sie ihn aus und nehmen die Änderungen im Bereich Eigenschaften vor.
- Wollen Sie einen Parameter hinzufügen, klicken Sie auf die Schaltfläche **Hinzufügen** und definieren dann die Merkmale im Bereich Eigenschaften.
- Wollen Sie einen Parameter löschen, wählen Sie ihn in der Liste aus und klicken auf die Schaltfläche **Löschen**.

Hinweis: Änderungen in den zusätzlichen Parametern werden nur berücksichtigt, wenn die Proxy Methode über die Schaltfläche **Erzeugen** erstellt wurde.

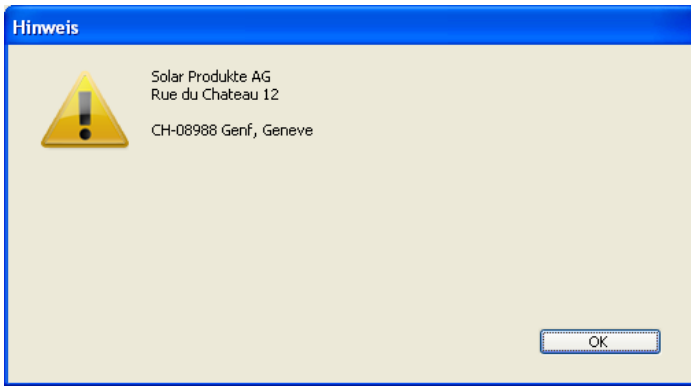
Proxy Methode aufrufen

Um eine Proxy Methode in Ihrem Code aufzurufen, schreiben Sie einfach ihren Namen und übergeben die erforderlichen Parameter. Diese Parameter deklariert der Assistent für Web Services im Header-Bereich der Proxy Methode. In Übereinstimmung mit der standardmäßigen Syntax zum Übergeben von Parametern zwischen Methoden in 4D lauten sie \$0, \$1, \$2... Sie erscheinen auch in den zusätzlichen Parametern der Beschreibung zu den veröffentlichten Methoden (siehe Abschnitt **Zusätzliche Parameter**) und werden manchmal in der dazugehörigen Dokumentation beschrieben.

Beispiel: In einer Methode lässt sich die Proxy Methode folgendermaßen aufrufen:

```
$Adresse:=proxy_WS_Kunde("Solar")
ALERT($Adresse)
```

Nach Ausführung der Methode erscheint folgende Meldung:



Komplexe Typen

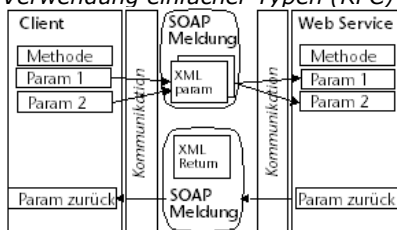
Mit 4D können Sie Web Services verwenden, die im RPC oder DOC Modus veröffentlicht werden und komplexe Typen enthalten (siehe **Kompatibilität von RPC, DOC und komplexen Typen**).

Hinweis: Auch bei komplexen XML Typen behandelt 4D Daten Arrays als einfache Typen.

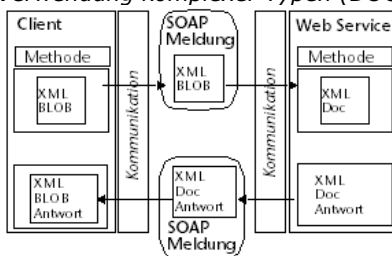
Proxy Methoden mit komplexen Typen, welche der Assistent für Web Services erzeugt, ähneln standardmäßigen Proxy Methoden, z.B. veröffentlicht im RPC Modus mit komplexen Typen oder im DOC Modus. Bei diesen Web Services werden Sie jedoch feststellen, dass der 4D Befehl **WEB SERVICE CALL** als Parameter eine Konstante mit den Wort *manuell* enthält.

Das bedeutet, dass solche Web Services weitere Bearbeitung durch den 4D Entwickler erfordern. Der Hauptgrund dafür ist, dass komplexe Typen in Form von Dokumenten oder XML Elementen ausgetauscht werden. Demzufolge muss zum Entnehmen oder Einfügen von Information in diesen SOAP Parametern zuerst XML parsing ausgeführt werden – im Gegensatz dazu sind in einfachen Typen Werte von Parametern direkt lesbar.

- **Verwendung einfacher Typen (RPC)**











- **Verwendung komplexer Typen (DOC)**



Der Assistent für Web Services kann nur Arrays und komplexe Typen mit einer Ebene verwalten, d.h. eine hierarchische Ebene in der SOAP Anfrage. Werden mehr komplexe Einträge in der Anfrage gefunden, signalisiert der Assistent über eine Flagge neben dem Methodennamen, wenn eine WSDL entdeckt wird. Die Verwaltung dieser Art Web Service erfordert in der Regel durch den Entwickler angepasstes Analysieren.

In 4D werden Parameter komplexer Typen – mit Ausnahme von Arrays – in Form von BLOBS verwaltet. Über die XML-Befehle können Sie deren Inhalt verwalten. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* in den Kapiteln **Web Services (Client)** und **XML DOM**.

Backup und Anwendung wiederherstellen

-  Überblick
-  Backup der Anwendung ausführen
-  Backup Einstellungen
-  Backup Dateien konfigurieren
-  Backup-Strategie konfigurieren
-  Logbuch verwalten
-  Daten wiederherstellen
-  Konfiguration und Dateien verfolgen

4D enthält ein komplettes Modul für Backup und Wiederherstellen.

Damit können Sie von einer gerade laufenden Anwendung ein Backup erstellen, ohne sie zu beenden. Jedes Backup kann die Strukturdatei, die Datendatei sowie einzelne Segmente und zusätzliche Dateien oder Ordner enthalten. Diese Parameter setzen Sie zuerst in den 4D Einstellungen.

Die Backups lassen sich manuell starten oder automatisch nach Zeitplan ohne Aktionen des Benutzers. Über spezifische Programmierbefehle und neue Datenbankmethoden lassen sich Backup-Funktionen in eine selbst gestaltete Oberfläche einbauen.

Datenbanken werden automatisch wiederhergestellt, wenn eine beschädigte Datenbank geöffnet wird.

Das integrierte Backup-Modul kann auch das Logbuch nutzen. Diese Datei schreibt alle mit den Daten ausgeführten Operationen mit und sorgt so für vollkommene Sicherheit zwischen zwei Backups. Treten bei einer laufenden Datenbank Probleme auf, werden Operationen, die in der Datendatei fehlen, beim nächsten Öffnen der Datenbank automatisch nachgetragen. Über ein Fenster können Sie den Inhalt des Logbuchs jederzeit ansehen.

Mit dem integrierten Backup-Modul können Sie:

- Jederzeit ein vollständiges Backup der Datenbankdateien starten (Strukturdatei, Datendatei, Logbuch, angehängte Dateien, etc.)
- In regelmäßigen Abständen automatische Backups einrichten – das kann stündlich, täglich, wöchentlich oder monatlich sein
- Komplexe Backup Parameter setzen, wie Satznummer, Dateikomprimierung, Optionen für Starten nach Wiederherstellen, etc.
- Eine Datenbank mit den dazugehörigen Dateien nach einem Zwischenfall automatisch wiederherstellen
- Fehlende Operationen automatisch aus dem Logbuch in eine wiederhergestellte Anwendung integrieren
- Operationen rückgängig machen, die in den Daten der Datenbank ausgeführt wurden

Hinweis: Sie können auch andere Lösungen zum Wiederherstellen und zur Datensynchronisation einbauen, um identische Versionen der Anwendung für Backup Zwecke beizubehalten. Diese Lösungen basieren auf folgenden Mechanismen und Technologien:

- Einen logischen Spiegel mit 4D Server setzen (über Mechanismen des integrierten Backup Moduls): siehe **Logischen Spiegel einrichten** im *4D Server Handbuch*
- Synchronisation über SQL: siehe **Replication via SQL** im Handbuch *4D SQL Reference*
- Synchronisation über HTTP: siehe im Handbuch *4D Programmiersprache*
- Einen allgemeinen Überblick über die Sicherheitsvorkehrungen in 4D finden Sie im [4D Security guide](#).

Backup der Anwendung ausführen

In 4D gibt es drei Möglichkeiten, ein Backup zu starten:

- Manuell über den Befehl **Datei/Ablage>Backup** oder im Maintenance- und Security-Center (MSC) über die Schaltfläche **Backup** (MSC).
- Automatisch über den Planer, der in den Datenbank-Eigenschaften eingerichtet wird
- Per Programmierung über den 4D Befehl **BACKUP**.

Die Wahl richtet sich nach Ihrer Verwendung der Anwendung und Ihrer Backup Strategie.

Hinweis für 4D Server: Ein Backup lässt sich über eine Methode, die den 4D Befehl **BACKUP** aufruft, auch manuell von einem Remote-Rechner aus starten. In allen Fällen wird er auf dem Server ausgeführt.

Um ein Backup manuell auszuführen:

1. Wählen Sie im Menü **Datei/Ablage>Backup**.
Auf dem Bildschirm erscheint folgendes Fenster:



Über das PopUp-Menü neben "Backup Ziel" sehen Sie den Speicherort des Backup-Ordners. Diesen legen Sie auf der **Seite Backup/ Konfiguration** der Datenbank-Eigenschaften fest.

ODER

Wählen Sie im Menü **Hilfe>Maintenance- und Security-Center** und öffnen die **Seite Backup**.

Weitere Informationen zum MSC finden Sie im Kapitel **Maintenance und Security Center**.

Über die Schaltfläche **Datenbank-Eigenschaften** öffnen Sie die **Seite Backup/ Konfiguration** der Datenbank-Eigenschaften.

2. Klicken Sie auf **Backup**, um das Backup mit den aktuellen Parametern zu starten.

Um ein Backup automatisch nach Plan auszuführen:

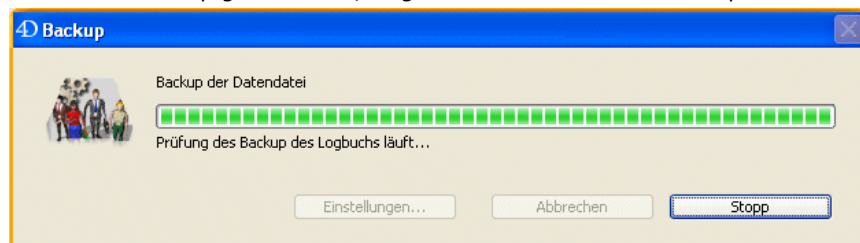
1. Definieren Sie auf der **Seite Backup/ Planer** der Datenbank-Eigenschaften die Häufigkeit des Backup.
Das Backup wird automatisch im hier definierten Intervall ausgeführt, ohne dass der Benutzer irgendetwas tun muss. Weitere Informationen dazu finden Sie im Abschnitt **Backup Einstellungen**.

Um ein Backup nach Plan über Programmierung auszuführen:

1. Führen Sie den 4D Befehl **BACKUP** in einer Methode aus.
Das Backup startet mit den aktuellen Parametern. Zur Abwicklung des Backups können Sie auch die **Datenbankmethode On Backup Startup** und **Datenbankmethode On Backup Shutdown** verwenden. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Backup**.

Backup ausführen

Sobald das Backup gestartet ist, zeigt 4D das Fortschreiten der Operation in einem Dialogfenster an:



Der Ablaufbalken erscheint auch auf der Seite **Backup** des Maintenance- und Security-Centers, wenn Sie dieses Dialogfenster verwendet haben.

Über die Schaltfläche **Stopp** kann der Benutzer das Backup jederzeit unterbrechen. Weitere Informationen dazu finden Sie im Abschnitt **Probleme während einem Backup**.

Das Ergebnis des letzten Backup (erfolgreich oder fehlgeschlagen) wird im Bereich "Letzte Backup Information" auf der **Seite Backup** im MSC oder auf der **Seite Wartung** von 4D Server. Es wird außerdem im Backup-Journal der Datenbank abgelegt. Weitere Informationen dazu finden Sie im Abschnitt **Backup Journal**.

Während Backup auf die Datenbank zugreifen

Während ein Backup läuft, schränkt 4D je nach Kontext den Zugriff auf die Datenbank ein. 4D sperrt alle Prozesse in den im Backup enthaltenen Dateitypen: Wird von der Strukturdatei ein Backup erstellt, ist nur der Zugriff auf die Struktur gesperrt, jedoch nicht auf die Daten.

Wird nur von der Datendatei ein Backup erstellt, ist der Zugriff auf die Struktur weiterhin möglich. Es gibt dann folgende Zugriffsmöglichkeiten:

- In 4D Einzelplatz ist die Datenbank im Lese- und Schreibmodus gesperrt; alle Prozesse sind eingefroren. Es lassen sich keine Aktionen ausführen.
- In 4D Server ist die Datenbank nur im Schreibmodus gesperrt; Client-Rechner können die Daten ansehen. Sendet ein Client eine Anfrage für Hinzufügen, Ändern oder Entfernen an den Server, erscheint eine Meldung, dass der Benutzer bis zum Ende des Backup warten muss.
Wurde die Datenbank gesichert, verschwindet das Fenster und die Aktion wird ausgeführt. Um die laufende Anfrage abzuberechnen und nicht das Ende des Backup abzuwarten, klicken Sie auf die Schaltfläche **Abbrechen**.
Stammt die Aktion in Warteposition jedoch von einer Methode, die vor dem Backup gestartet wurde, sollte sie nicht abgebrochen werden, da nur noch verbleibende offene Operationen abgebrochen werden. Darüberhinaus kann eine nur teilweise ausgeführte Methode zu Inkonsistenzen in der Datenbank führen.
Hinweis: Stammt die Aktion in Warteposition von einer Methode und klickt der Benutzer auf die Schaltfläche **Abbrechen**, gibt 4D Server den Fehler -9976 zurück (Dieser Befehl kann nicht ausgeführt werden, da ein Backup läuft).

Probleme während einem Backup

Es kann passieren, dass ein Backup nicht korrekt ausgeführt wird. Das kann verschiedene Ursachen haben: Unterbrechung durch den Benutzer, angefügte Datei wird nicht gefunden, Probleme auf dem Zieldatenträger, unvollständige Transaktion, etc. 4D bearbeitet den Zwischenfall je nach Ursache.

In allen Fällen wird der Status des letzten Backup (erfolgreich oder fehlgeschlagen) angezeigt, und zwar im MSC auf der **Seite Backup** im Bereich Letztes Backup oder auf der **Seite Wartung** von 4D Server, sowie im **Backup Journal** der Datenbank.

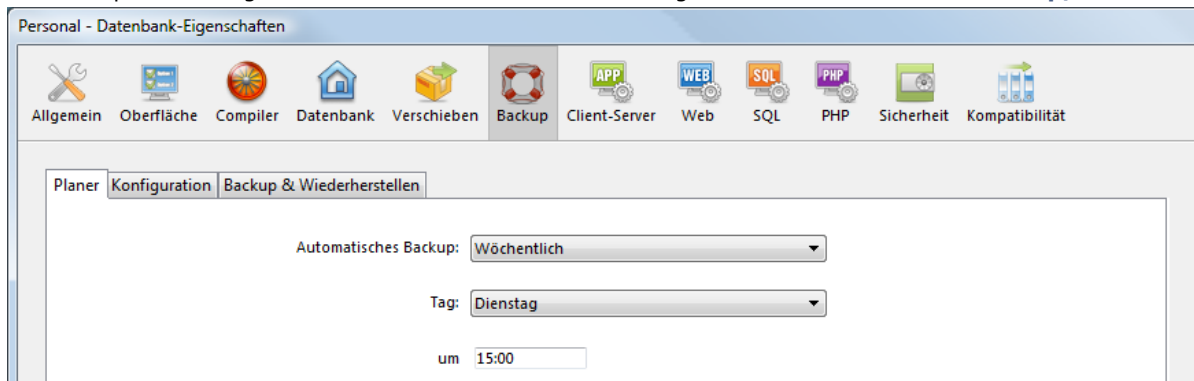
- **Unterbrechung durch den Benutzer:** Der Benutzer kann das Backup im Dialogfenster, das den Verlauf anzeigt, jederzeit über die Schaltfläche **Stopp** abbrechen. Dann stoppt das Kopieren von Elementen und ein Fehler wird angezeigt. Sie können diesen Fehler über die **Datenbankmethode On Backup Shutdown** abfangen.
- **Angefügte Datei wird nicht gefunden:** Ist eine angefügte Datei nicht auffindbar, führt 4D das Backup teilweise durch, d.h. Backup der Dateien der Datenbank und verfügbaren angefügten Dateien, und gibt einen Fehler zurück.
- **Backup ist nicht möglich** (Platte ist voll oder schreibgeschützt, Platte fehlt oder ist fehlerhaft, unvollständige Transaktion, Datenbank zum Zeitpunkt für geplantes automatisches Backup nicht gestartet, etc.):
 - Tritt der Fehler zum ersten Mal auf, versucht 4D erneut, das Backup auszuführen. Die Wartezeit zwischen zwei Versuchen wird auf der **Seite Backup/ Backup & Wiederherstellen** der Datenbank-Eigenschaften definiert.
 - Scheitert auch der zweite Versuch, erscheint ein Meldedialog des Systems und ein Fehler wird angezeigt. Sie können diesen Fehler über die **Datenbankmethode On Backup Shutdown** abfangen.

Backup Einstellungen

Sie können das Backup von Anwendungen, die mit 4D oder 4D Server geöffnet wurden, automatisieren, selbst wenn kein Client-Rechner angemeldet ist. Dazu gehört die Häufigkeit (stündlich, täglich, wöchentlich oder monatlich); 4D startet für jede Sitzung automatisch ein Backup gemäß den aktuellen Backup-Einstellungen.

Wurde die Anwendung zum vorgegebenen Zeitpunkt des Backup nicht gestartet, wertet 4D das Backup beim nächsten Start als fehlgeschlagen und wendet die Konfiguration an, die in den Datenbank-Eigenschaften definiert wurde. Weitere Informationen dazu finden Sie im Abschnitt **Probleme während einem Backup**.

Die Backup-Einstellungen definieren Sie in den Datenbank-Eigenschaften auf der **Seite Backup/ Planer**:



Personal - Datenbank-Eigenschaften

Allgemein Oberfläche Compiler Datenbank Verschieben **Backup** Client-Server Web SQL PHP Sicherheit Kompatibilität

Planer Konfiguration Backup & Wiederherstellen

Automatisches Backup: Wöchentlich

Tag: Dienstag

um 15:00

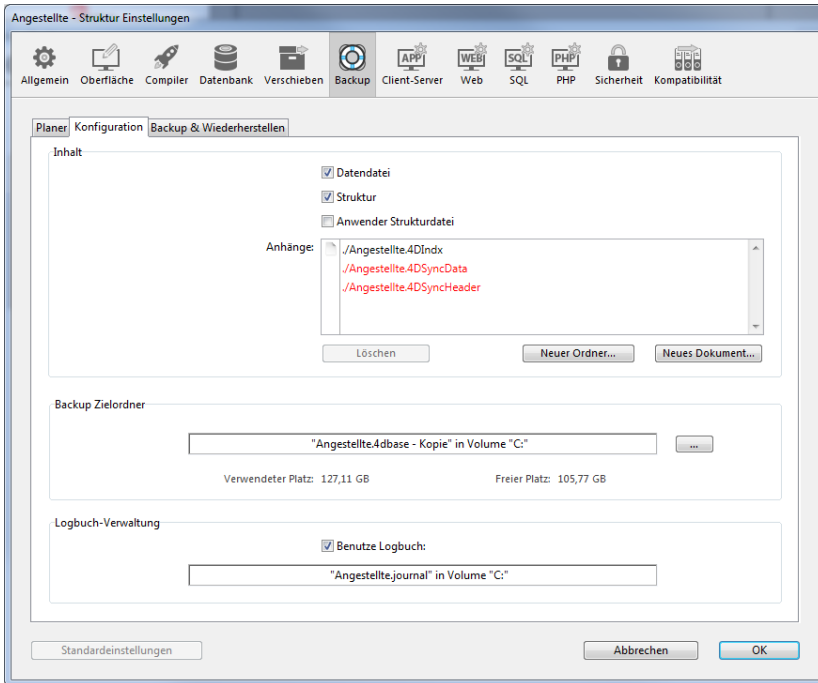
Die Optionen dieser Registerkarte ermöglichen, automatische Backups der Datenbank nach Zeitplan einzurichten. Sie können rasch eine Standardeinstellung wählen oder einen eigenen Zeitplan aufstellen.

Im DropDown-Menü **Automatisches Backup** gibt es folgende Optionen:

- **Nie**: Es gibt keinen Zeitplan für automatisches Backup.
- **Stündlich**: Definiert ein automatisches Backup jede Stunde, beginnend mit der nächsten Stunde.
- **Täglich**: Definiert ein automatisches Backup jeden Tag. Sie können dann die Zeit angeben, wann das Backup starten soll.
- **Wöchentlich**: Definiert ein automatisches Backup jede Woche. Es gibt zwei weitere Bereiche, wo Sie Tag und Uhrzeit zum Starten des Backup eintragen können.
- **Monatlich**: Definiert ein automatisches Backup jeden Monat. Es gibt zwei weitere Bereiche, wo Sie Tag des Monats und Uhrzeit zum Starten des Backup eintragen können.
- **Angepaßt**: Hier können Sie individuell auf Ihre Bedürfnisse ausgerichtete automatische Backups einrichten. Wählen Sie diese Option, erscheinen mehrere zusätzliche Eingabebereiche:
 - **Alle X Stunden**: Damit programmieren Sie Backups nach Stunden. Sie können einen Wert zwischen 1 und 24 eingeben.
 - **Alle X Tage um xx**: Damit programmieren Sie Backups nach Tagen. Geben Sie 1 ein, wenn das Backup jeden Tag durchgeführt werden soll. Ist diese Option markiert, müssen Sie auch die Uhrzeit eintragen, zu der das Backup starten soll.
 - **Alle X Wochen um xx**: Damit programmieren Sie Backups nach Wochen. Geben Sie 1 ein, wenn das Backup jede Woche durchgeführt werden soll. Ist diese Option markiert, müssen Sie auch den Wochentag und die Uhrzeit eintragen, zu der das Backup starten soll. Sie können bei Bedarf auch mehrere Wochentage ankreuzen. Sie können z.B. festlegen, dass das Backup zweimal pro Woche läuft, nämlich am Mittwoch und am Freitag.
 - **Alle X Monate, X.Tag um xx**: Damit programmieren Sie Backups nach Monaten. Geben Sie 1 ein, wenn das Backup jeden Monat durchgeführt werden soll. Ist diese Option markiert, müssen Sie auch den Wochentag und die Uhrzeit eintragen, zu der das Backup starten soll.

Backup Dateien konfigurieren

Auf der **Seite Backup/ Konfiguration** der Datenbank-Eigenschaften definieren Sie die Backup-Dateien, das Logbuch und den jeweiligen Speicherort. Diese Parameter gelten für alle Anwendungen, die mit 4D geöffnet werden.



Hinweis für 4D Server: Im Client-/Server-Betrieb lassen sich die Backup-Einstellungen nur auf dem Rechner mit 4D Server setzen.

Inhalt

Hier definieren Sie, welche Dateien bzw. Ordner beim nächsten Backup kopiert werden.

- **Datendatei:** Datendatei der Anwendung. Ist diese Option markiert, wird das aktuelle Logbuch der Anwendung – sofern vorhanden – zeitgleich mit den Daten gesichert.
- **Strukturdatei:** Strukturdatei der Anwendung. Diese Option ermöglicht bei kompilierten Datenbanken ein Backup der Datei .4dc.
- **Anwender-Strukturdatei** (optional): Anwender-Strukturdatei der Anwendung, die angepasste Benutzerformulare enthält – sofern vorhanden. Weitere Informationen dazu finden Sie im Kapitel **Benutzerformulare**.
- **Anhänge:** Hier können Sie Dateien bzw. Ordner wählen, die zusammen mit der Datenbank als Backup gesichert werden sollen. Das können Dokumente, Vorlagen von Plug-Ins, Etiketten, Berichte, Bilder, o.ä. sein. Sie können eigene Dateien oder Ordner bestimmen, deren Inhalt vollkommen als Backup gesichert wird. Jedes angefügte Element wird mit dem kompletten Zugriffspfad im Bereich „Anhänge“ aufgeführt.
 - **Löschen:** Entfernt die gewählte Datei aus der Liste der Anhänge.
 - **Neuer Ordner:** Öffnet ein Dialogfenster, um Ordner für das Backup auszuwählen. Beim Wiederherstellen wird dieser Ordner mit seiner internen Struktur wiederhergestellt. Sie können jeden Ordner bzw. jedes Verzeichnis wählen, das an den Rechner angeschlossen ist, mit Ausnahme des Ordners mit den Dateien der Anwendung.
 - **Neues Dokument:** Öffnet ein Dialogfenster, um Dateien dem Backup hinzuzufügen.

Weitere Informationen zu Dateien der 4D Datenbank finden Sie im Abschnitt **Beschreibung der 4D Dateien**.

Backup Zielordner

Hier definieren Sie, wo die Backup-Dateien und Logbücher (sofern vorhanden) gespeichert werden sollen.

Um den Speicherort dieser Dateien anzusehen, klicken Sie in das DropDown-Menü, um den Pfadnamen anzuzeigen.

Um den Speicherort dieser Dateien zu ändern, klicken Sie auf die Schaltfläche [...]. Auf dem Bildschirm erscheint ein Auswahldialog, in dem Sie einen Ordner bzw. eine Festplatte zum Ablegen der Backups wählen können. Die Bereiche "Verwendeter Platz" und "Freier Platz" werden automatisch aktualisiert. Sie geben an, wieviel Speicherplatz auf der Festplatte mit dem gewählten Ordner belegt und wieviel frei ist.

Logbuch-Verwaltung

Ist die Option **Benutze Logbuch** markiert, verwendet jede mit 4D erstellte Datenbank ein Logbuch. Der Pfad wird im Bereich darunter angegeben. Die Datenbank lässt sich nicht ohne Logbuch öffnen, wenn diese Option markiert ist.

Standardmäßig verwendet jede mit 4D erstellte Datenbank ein Logbuch (die Option auf der **Seite Allgemein** der 4D Einstellungen ist markiert). Das Logbuch hat den Namen *DatendateiName.journal* und liegt im gleichen Ordner wie die Strukturdatei.

Aktivieren Sie ein neues Logbuch, muss zuvor ein Backup der Daten in der Datenbank erstellt werden. Ist diese Option aktiv, erscheint auf dem Bildschirm die Meldung, dass ein Backup erforderlich ist:

Logbuch-Verwaltung

Benutze Logbuch:

"Angestellte.journal" in Volume "C:"

Das Protokollieren beginnt nach dem nächsten Backup der Datendatei.

Das Erstellen des **Logbuchs** wird verschoben und erst nach dem nächsten Backup der Datenbank erstellt.

Backup-Strategie konfigurieren

Das Ändern der Backup Optionen ist optional. Eine standardmäßige Einstellung ist vordefiniert.

Die Backup Optionen setzen Sie auf der **Seite Backup/ Backup & Wiederherstellen** der Datenbank-Eigenschaften:

The screenshot shows the 'Personal - Datenbank-Eigenschaften' dialog box with the 'Backup & Wiederherstellen' tab selected. The 'Allgemeine Einstellungen' section includes:

- Aufheben der letzten 5 Backup Dateien
- Backup nur wenn Datendatei geändert
- Lösche ältesten Satz: nach Backup
- Wenn Backup fehlschlägt: Erneut nach planmäßigem Datum und Zeit, Erneut nach 60 Sekunden, Abbruch nach 5 Versuchen

The 'Archive' section includes:

- Segment Größe (Mb): Keine
- Komprimierung: Keine
- Interlacing: Keine
- Redundanz: Keine

The 'Automatische Wiederherstellung' section includes:

- Letztes Backup wiederherstellen, wenn Datenbank beschädigt
- Letztes Logbuch integrieren, wenn Datenbank unvollständig

Buttons at the bottom: Standardeinstellungen, Abbrechen, OK.

Allgemeine Einstellungen

Hier bestimmen Sie spezifische Arbeitsweisen beim Ausführen von Sicherungen.

- Aufheben der letzten X Backup-Dateien:** Damit legen Sie fest, dass die älteren Backup-Dateien gelöscht werden und vermeiden so eine Überbelegung der Festplatte.
Das funktioniert folgendermaßen: Ist das aktuelle Backup komplett, löscht 4D das älteste Archiv, wenn es an derselben Stelle liegt und denselben Namen hat. Sie können auch bestimmen, dass das älteste Archiv vor Ausführen des Backup gelöscht wird, um Platz auf der Festplatte zu sparen.
Ist z.B. als Anzahl 3 definiert, erstellen die ersten drei Backups jeweils die Archive MeineDB-0001, MeineDB-0002 und MeineDB-0003. Beim 4. Backup wird das Archiv MeineDB-0004 erstellt und MeineDB-0001 gelöscht.
Standardmäßig ist der Löschmechanismus aktiviert und 4D behält 3 Backup-Sätze.
Wollen Sie diesen Mechanismus deaktivieren, deaktivieren Sie einfach diese Option.
Hinweis: Dieser Parameter betrifft sowohl die Backup-Sätze der Datenbank als auch das Logbuch der Backup-Sätze.
- Backup nur wenn Datendatei geändert:** Ist diese Option markiert, startet 4D geplante Backups nur, wenn seit dem letzten Backup Daten in der Datenbank hinzugefügt, verändert oder gelöscht wurden. Sonst wird das geplante Backup annulliert und auf das nächste geplante Backup gelegt. Es erscheint keine Fehlermeldung; das Logbuch notiert jedoch, dass das Backup verschoben wurde. Damit sparen Sie Rechnerzeit für Backups von Datenbanken, die überwiegend zum Lesen genutzt werden. Beachten Sie jedoch, dass mit dieser Option keine Änderungen an der Struktur oder angefügten Dateien berücksichtigt werden.
Hinweis: Dieser Parameter betrifft sowohl Backups der Datenbank als auch des Logbuchs.
- Lösche ältesten Satz nach/vor Backup:** Diese Option wird nur verwendet, wenn die Option "Aufheben der letzten X Backup Dateien" markiert ist. Damit geben Sie an, ob 4D das älteste Archiv vor bzw. nach Ausführen des Backup löscht. Das funktioniert nur, wenn das älteste Archiv weder umbenannt noch verschoben wurde.
- Wenn Backup fehlschlägt:** Damit legen Sie fest, wie fehlgeschlagene Backups verwaltet werden (Backup nicht möglich). Lässt sich ein Backup nicht ausführen, ermöglicht 4D einen zweiten Versuch.
Hinweis: 4D wertet ein Backup als fehlgeschlagen, wenn die Datenbank zum Zeitpunkt des automatisch geplanten Backups nicht gestartet ist.

- **Erneut nach planmäßigem Datum und Zeit:** Diese Option ist nur sinnvoll für geplante automatische Backups. Das fehlgeschlagene Backup wird abgebrochen und ein Fehler erscheint.
- **Erneut nach X Sekunden, Minuten oder Stunden:** Mit dieser Option wird nach der Wartezeit ein neues Backup versucht. So lassen sich bestimmte Umstände umgehen, die ein Backup blockieren können. Sie können die Wartezeit in Sekunden, Minuten oder Stunden angeben. Scheitert auch der 2. Versuch, erscheint ein Fehler. Das wird auch im Statusbereich des letzten Backup und im Logbuch des Backup vermerkt.
- **Abbruch nach X Versuchen:** Damit legen Sie die maximale Anzahl für gescheiterte Backup-Versuche fest. Ist die max. Anzahl der Versuche erreicht, wird das Backup annulliert und der Fehler 1401 generiert ("Die max. Anzahl der Backup-Versuche ist erreicht; automatisches Backup ist vorübergehend deaktiviert"). In diesem Fall wird ein neues automatisches Backup erst ausgeführt, wenn die Anwendung neu gestartet bzw. ein manuelles Backup erfolgreich ausgeführt wurde.
Dieser Parameter ist hilfreich, um bei einem größeren Problem (das den Eingriff des Administrators erfordert) zu verhindern, dass immer wieder ein Backup-Versuch läuft, der letztendlich die Performance der Datenbank beeinträchtigt. Dieser Parameter ist standardmäßig nicht markiert.

Archive

Hier legen Sie die Einstellungen zum Anlegen des Archivs fest. Sie gelten für die Dateien von Haupt-Backups sowie für Logbücher der Backups.

- **Segmentgröße (Mb)**

Mit 4D können Sie Archive segmentieren, d.h. ein Archiv in mehrere kleinere aufteilen. So können Sie z.B. ein Backup über verschiedene Medien, wie CD, ZIP o.ä. speichern. Beim Wiederherstellen fügt 4D die Segmente automatisch wieder zusammen. Jedes Segment hat den Namen MeineDBxxxx-yyyy].4BK, wobei xxxx die Nummer des Backup ist, yyyy die Nummer des Segments.

Segment Größe ist eine Combobox, in der Sie für jedes Segment des Backup die Größe in Mb festlegen. Sie können eine vordefinierte Größe wählen oder eine eigene Größe zwischen 0 und 2048 angeben. Bei 0 wird nicht segmentiert, das entspricht dem Eintrag **Keine**.

- **Komprimierung**

4D komprimiert die Backups, um Platz zu sparen. Das kann jedoch bei großen Datenmengen den Backup-Prozess beträchtlich verlangsamen.

Es gibt folgende Möglichkeiten:

- **Keine:** Die Dateien werden nicht komprimiert. Das Backup läuft schneller ab, die Archiv-Dateien sind jedoch um einiges größer.
- **Schnell (Standard):** Das ist ein Mittelmaß zwischen Geschwindigkeit des Backup und Größe des Archivs.
- **Kompakt:** Hier wird das Archiv maximal komprimiert. Die Dateien beanspruchen den kleinstmöglichen Platz auf der Festplatte, das Backup läuft jedoch um einiges langsamer ab.

- **Interlacing und Redundanz**

4D erstellt Archive über spezifische Algorithmen, die auf Operationen zur Optimierung (Interlacing) und Sicherheit (Redundanz) beruhen. Dies können Sie gemäß Ihren Anforderungen einstellen. 4D enthält standardmäßig den Wert **Keine**. Weitere Möglichkeiten sind **Niedrig, Mittel** und **Hoch**.

- **Interlacing:** Damit werden Daten in nicht aufeinander folgenden Sektoren gespeichert, um das Risiko bei Beschädigung eines Sektors zu begrenzen. Je höher die Rate, desto besser ist die Sicherheit; Die Datenbearbeitung benötigt jedoch mehr Speicher.
- **Redundanz:** Damit werden die Daten in einer Datei abgesichert, indem dieselbe Information mehrere Male wiederholt wird. Je höher die Redundanzrate, desto besser ist die Sicherheit der Datei. Nachteile sind langsame Speicherung und entsprechend große Dateien.

Automatische Wiederherstellung

Diese Optionen bestimmen die automatischen Mechanismen beim Öffnen einer beschädigten Datenbank.

- **Letztes Backup wiederherstellen, wenn Datenbank beschädigt:** Mit dieser Option startet das Programm automatisch die Wiederherstellung der Daten des letzten gültigen Backup der Datenbank, wenn beim Hochfahren der Datenbank eine Unregelmäßigkeit festgestellt wird (z.B. beschädigte Datei). Der Benutzer selbst muss nichts ausführen; die Operation wird jedoch im Logbuch des Backup vermerkt.
Hinweis: Beim automatischen Wiederherstellen wird nur die Datendatei wiederhergestellt. Soll das auch für die angefügten Dateien oder die Strukturdatei gelten, müssen Sie eine manuelle Wiederherstellung ausführen.
- **Letztes Logbuch integrieren, wenn Datenbank unvollständig:** Ist diese Option markiert, integriert das Programm automatisch das Logbuch, wenn die Datenbank wiederhergestellt oder geöffnet wird.
 - Beim Öffnen einer Datenbank wird das aktuelle Logbuch automatisch integriert, wenn 4D feststellt, dass die im Logbuch gespeicherten Operationen in den Daten nicht vorhanden sind. Das passiert z.B. bei Stromausfall, wenn es Operationen im Daten-Cache gibt, die noch nicht in den Hauptspeicher übertragen wurden.
 - Hat beim Wiederherstellen einer Datenbank das aktuelle Logbuch bzw. Logbuch des Backup dieselbe Nummer wie das Backup, das im gleichen Ordner gespeichert ist, prüft 4D seinen Inhalt. Enthält es Operationen, die nicht in der Datendatei enthalten sind, wird es automatisch integriert.

Auf Benutzerseite erscheint kein Dialogfenster, die Operation läuft vollkommen automatisch ab. Ziel dabei ist, die Verwendung so einfach wie möglich zu machen. Die Operation wird im Backup Journal vermerkt.

Logbuch verwalten

Eine Datenbank im Einsatz speichert ständig Änderungen, es werden Datensätze hinzugefügt oder gelöscht. Die Durchführung von Backups in regelmäßigen Abständen ist wichtig, ermöglicht aber nicht, nach einem Zwischenfall die nach dem letzten Backup eingegebenen Daten wiederherzustellen. Dafür bietet 4D jetzt ein spezifisches Hilfsmittel: das Logbuch. Diese Datei sorgt für permanente Sicherheit der Daten der Datenbank.

Zusätzlich arbeitet 4D kontinuierlich mit dem Daten-Cache im Speicher. Alle Änderungen an den Daten der Datenbank werden im Cache zwischengespeichert, bevor sie auf die Festplatte geschrieben werden. Das beschleunigt die Operationen der Programme, da der Zugriff auf den Speicher schneller ist als auf die Festplatte. Tritt ein Zwischenfall ein, bevor die im Cache gespeicherten Daten auf die Festplatte geschrieben wurden, müssen Sie beim Wiederherstellen der Datenbank auch das aktuelle Logbuch integrieren.

Schließlich hat 4D Funktionen, die den Inhalt des Logbuchs analysieren, so dass ein Rollback der Operationen möglich ist, die in den Daten der Anwendung ausgeführt wurden. Diese Funktionen sind im MSC verfügbar. Weitere Informationen dazu finden Sie auf der [Seite Aktivitätsanalyse](#) und [Seite Zurückfahren](#).

Arbeitsweise

Das von 4D angelegte Journal enthält alle Operationen, die in den Daten von protokollierten Tabellen der Anwendung ausgeführt werden, die sequentiell protokolliert werden. Standardmäßig werden alle Tabellen protokolliert, d.h. in das Journal aufgenommen. Sie können einzelne Tabellen davon ausschließen, wenn Sie im Inspektorfenster die Option **In Journal aufnehmen** deaktivieren.

Somit löst jeder vom Benutzer ausgeführte Schritt zwei Aktionen aus: Einerseits wird die Anweisung des Benutzers in der Anwendung ausgeführt, zur selben Zeit wird die Aktion im Journal mitgeschrieben. Das Journal wird unabhängig erstellt, es beeinträchtigt und verlangsamt nicht die Arbeit des Benutzers. Eine Anwendung kann zur selben Zeit immer nur mit einem Journal arbeiten.

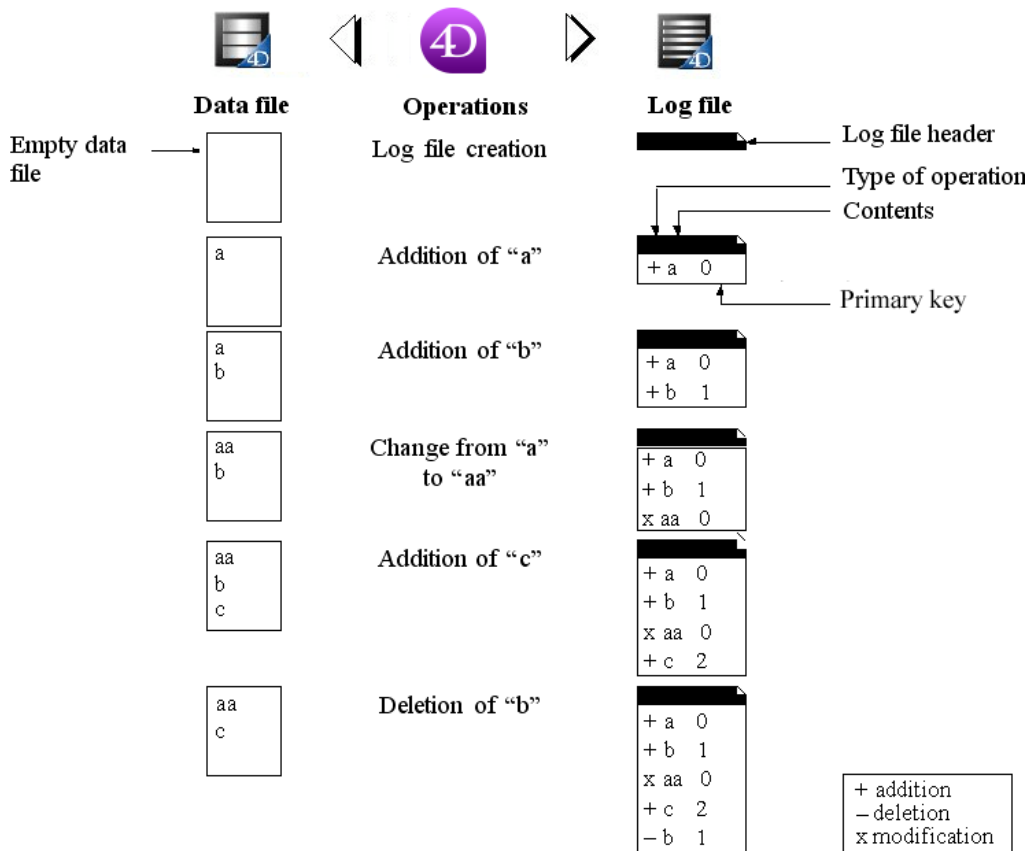
Das Journal registriert folgende Operationen:

- Datendatei öffnen und schließen
- Prozess (Kontexte) öffnen und schließen
- Datensätze oder BLOBs hinzufügen
- Datensätze ändern
- Datensätze löschen
- Transaktionen erstellen und schließen

Weitere Informationen dazu finden Sie auf der [Seite Aktivitätsanalyse](#).

4D verwaltet das Journal. Es schreibt alle Operationen mit, die die Datendatei betreffen, egal welche Aktion vom Benutzer, in 4D Methoden, der SQL Engine, 4D Plug-Ins (4D Write, 4D View, etc.) oder einem Web Browser ausgeführt werden.

Nachfolgende Darstellung zeigt, wie das Journal arbeitet:



Das aktuelle Journal wird automatisch mit der aktuellen Datendatei gesichert. Das hat folgende Vorteile:

- Wichtig ist, dass das Volume, in welchem das Journal gespeichert wird, nicht voll wird. Ohne Backup würde das Journal mit der Zeit beträchtlich anwachsen und u.U. den gesamten Platz im Verzeichnis belegen. 4D bzw. 4D Server schließt für jeden Backup-Satz das aktuelle Journal und startet sofort eine neue leere Datei, so dass auf dem Volume genügend Platz frei bleibt. Das alte Journal wird je nach Backup-Einstellungen archiviert oder entfernt.
- Behalten Sie die Journale zu den entsprechenden Backups, damit Sie eine Datenbank später analysieren oder reparieren können. Das Journal kann nur in die Anwendung integriert werden, zu der es gehört. Sie müssen Backups und Journale zusammen archivieren, damit Sie das Journal bei Bedarf korrekt in das Backup einbinden können.

Logbuch erstellen

Standardmäßig führt jede mit 4D erstellte Datenbank ein Logbuch, d.h. die Option auf der **Seite Allgemein** der 4D Einstellungen ist markiert. Das Logbuch hat den Namen **DatenDateiName.journal** und wird in den Ordner mit der Strukturdatei der Datenbank gelegt.

Sie können jederzeit prüfen, ob Ihre Datenbank ein Logbuch führt. Dazu gehen Sie auf die **Seite Backup/ Konfiguration** der Datenbank-Eigenschaften (siehe **Logbuch-Verwaltung**) und prüfen, ob die Option **Benutze Logbuch** markiert ist. Ist diese Option inaktiv oder verwenden Sie eine Datenbank ohne Logbuch und wollen dann eine Backup-Strategie mit Logbuch einrichten, müssen Sie ein Logbuch anlegen.

Um ein Logbuch anzulegen:

1. Markieren Sie auf der **Seite Backup/ Konfiguration** der Datenbank-Eigenschaften die Option **Benutze Logbuch**. Das Programm zeigt den Standarddialog zum Sichern einer Datei oder zum Anlegen eines neuen Logbuchs. Es hat standardmäßig den Namen **DatenDateiName.journal**.
2. Behalten Sie den Standardnamen oder benennen ihn um und wählen dann den Speicherort der Datei. Wir empfehlen, das Logbuch nicht auf dieselbe Platte wie die Datenbank zu legen. Geht diese Platte kaputt, können Sie in jedem Fall das Logbuch aufrufen.
3. Klicken Sie auf die Schaltfläche **Speichern**. Zugriffspfad und Name des aktuellen Logbuchs erscheinen nun in den Einstellungen der Datenbank im Bereich "Benutze Logbuch". Sie können in diesen Bereich klicken, um ein PopUp-Menü mit den Ordner auf der Festplatte anzuzeigen.
4. Bestätigen Sie dieses Dialogfenster.

Wichtig: Ab 4D v14 gibt es neue interne Mechanismen zum Generieren und Einsetzen eines Logbuchs (Datei .journal), das es noch wirkungsvoller macht. Zum Verwenden eines Logbuchs müssen alle protokollierten Tabellen einen Primärschlüssel haben. Ist das nicht der Fall, z.B. in einer konvertierten Datenbank, zeigt 4D eine Warnung, dass sich das Logbuch nicht aktivieren lässt, solange nicht die Primärschlüssel für alle protokollierten Tabellen korrekt gesetzt sind. Mit dem **Primärschlüssel-Assistent** können Sie Ihre Anwendung korrekt konfigurieren. Er prüft den Status jeder Tabelle und ermöglicht, einen Primärschlüssel zu setzen.

Damit Sie ein Logbuch direkt erstellen können, muss die Datenbank sich in einer der folgenden Situationen befinden:

- Jede protokollierte Tabelle muss einen Primärschlüssel haben (s.o.)
- Die Datendatei ist leer
- Sie haben gerade ein Backup der Datenbank ausgeführt und es hat noch keine Änderungen an den Daten gegeben

In allen anderen Fällen erscheint die Meldung, dass Sie ein Backup durchführen müssen. Klicken Sie auf die Schaltfläche **OK**, startet sofort das Backup, dann wird das Logbuch aktiviert. Klicken Sie auf **Abbrechen**, wird die Anfrage gesichert, das Logbuch wird nicht jetzt, sondern erst nach dem nächsten Backup der Datenbank erstellt.

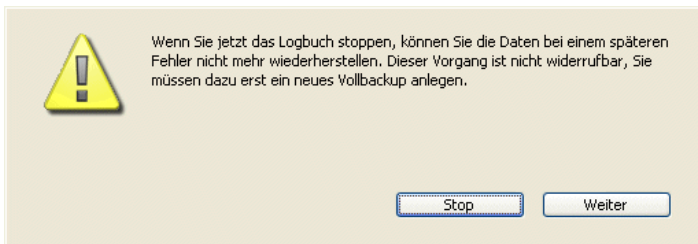
Diese Vorsichtsmaßnahme ist notwendig, denn, damit Sie die Datenbank nach einem Zwischenfall wiederherstellen können, benötigen Sie eine Kopie der Datenbank, in die Sie die im Logbuch gespeicherten Operationen integrieren können. Ab jetzt werden alle Operationen in den Daten im Logbuch mitgeschrieben. Dieses wird in Zukunft immer zusammen mit der Datenbank verwendet

Erstellen Sie eine neue Datendatei, müssen Sie auch ein neues Logbuch anlegen. Sie müssen ein anderes Logbuch einrichten, wenn Sie eine andere Datendatei öffnen, die nicht mit einem Logbuch verknüpft ist oder wenn das Logbuch fehlt.

Logbuch stoppen

Wollen Sie das Mitschreiben der Operationen für das aktuelle Logbuch stoppen, gehen Sie auf die [Seite Backup/Konfiguration](#) der Datenbank-Eigenschaften und deaktivieren die Option **Benutze Logbuch**.

Auf dem Bildschirm erscheint folgende Meldung:



Klicken Sie auf die Schaltfläche **Stop**, wird das aktuelle Logbuch sofort geschlossen, d.h. das Dialogfenster Datenbank-Eigenschaften muss danach nicht bestätigt werden.

Wollen Sie das aktuelle Logbuch schließen, weil es zu groß ist, müssen Sie von der Datendatei ein Backup ausführen, welches dann auch das Logbuch sichert.

Hinweise für 4D Server:

- Über die 4D Funktion **New log file** lässt sich das aktuelle Logbuch automatisch schließen und ein neues starten.
- Ist das Logbuch während einer Arbeitssitzung aus irgendwelchen Gründen nicht verfügbar, wird der Fehler 1274 generiert und 4D Server erlaubt Benutzern nicht mehr, Daten zu schreiben. Ist das Logbuch wieder verfügbar, muss ein Backup gemacht werden.

Daten wiederherstellen

4D ermöglicht, komplette Datenteile der Datenbank bei Zwischenfällen, egal welcher Art, wiederherstellen. Es gibt zwei Hauptkategorien für Zwischenfälle:

• **Unvorhergesehener Abbruch der Datenbank in Betrieb**

Dieser Zwischenfall kann bei Stromausfall, Fehler im Betriebssystem, o.ä. passieren. Je nachdem, welchen Status der Daten-Cache im Augenblick des Zwischenfalls hat, sind zum Wiederherstellen der Datenbank unterschiedliche Operationen erforderlich:

- War der Cache leer, öffnet sich die Datenbank ganz normal. Alle Änderungen in der Datenbank wurden gespeichert. In diesem Fall ist keine besondere Operation erforderlich.
- Enthält der Cache Operationen, ist die Datenbank intakt. Sie müssen jedoch das aktuelle Logbuch integrieren, um den Zustand im Moment des Abbruchs wiederherzustellen.
- Sollte der Cache gerade auf die Festplatte geschrieben werden, ist die Datendatei wahrscheinlich beschädigt. In diesem Fall müssen Sie das letzte Backup wiederherstellen und das aktuelle Logbuch integrieren.

• **Verlust von Dateien der Datenbank**

Dieser Zwischenfall kann bei fehlerhaften Sektoren auf der Festplatte, bei einem Virus, bei Bedienungsfehlern, o.ä. passieren. Hier müssen Sie das letzte Backup wiederherstellen und das aktuelle Logbuch integrieren. Um herauszufinden, ob eine Datenbank bei einem Zwischenfall beschädigt wurde, starten Sie einfach die Datenbank wieder mit 4D. Das Programm führt selbst eine Prüfung durch und listet auf, welche Operationen zum Wiederherstellen ausgeführt werden müssen. Im automatischen Modus werden diese Operationen direkt ausgeführt, ohne Eingreifen durch den Benutzer. Haben Sie eine reguläre Backup-Strategie eingerichtet, finden Sie über die 4D Werkzeuge zum Wiederherstellen in der Regel die Datenbank in genau dem Zustand vor dem Zwischenfall vor.

Hinweise:

- 4D kann automatisch Operationen starten, um Datenbanken nach einem Zwischenfall wiederherzustellen. Diese Mechanismen werden über zwei Optionen auf der **Seite Backup/ Backup & Wiederherstellen** der Datenbank-Eigenschaften verwaltet. Weitere Informationen dazu finden Sie im Abschnitt **Automatische Wiederherstellung**.
- Beruht der Zwischenfall auf einer unpassenden Operation in den Daten, z.B. Datensatz gelöscht, können Sie versuchen, die Datenbank über die "rollback" Funktion im Logbuch in einen früheren korrekten Zustand zurückzusetzen. Diese Funktion ist auf der **Seite Zurückfahren** des MSC verfügbar.

Backup manuell wiederherstellen (Standarddialog)

Sie können den Inhalt eines Archivs, das vom Backup-Modul erstellt wurde, manuell wiederherstellen. Manuelles Wiederherstellen ist z.B. erforderlich, um den Archivinhalt im Ganzen zu reproduzieren (Strukturdateien und/oder angehängte Dateien) oder zur Ausführung von Suchläufen in den Archiven.

Wiederherstellen lässt sich auch mit Integration des aktuellen Logbuchs ausführen.

Sie können die manuelle Wiederherstellung von Backup über den Standard Öffnen-Dialog oder die Seite Wiederherstellen im Maintenance- und Security-Center (MSC) ausführen.

- Über den Standard Öffnen-Dialog können Sie jedes Archiv wiederherstellen. Dies wird nachfolgend beschrieben.
- Über das MSC können Sie weitere Optionen wählen, z.B. den Inhalt des Archivs einsehen. Dagegen können Sie hier nur Archive der geöffneten Datenbank wiederherstellen. Weitere Informationen dazu finden Sie im unteren Abschnitt **Backup manuell wiederherstellen (MSC)**.

Um eine Datenbank über den Standard Öffnen-Dialog manuell wiederherzustellen:

1. Starten Sie 4D und wählen im Menü **Datei/Ablage>Wiederherstellen**.

Für diesen Befehl muss nicht zwingend eine Datenbank geöffnet sein.

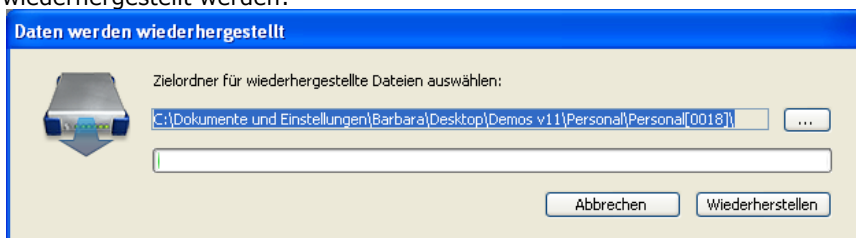
ODER

Führen Sie in einer 4D Methode den 4D Befehl **RESTORE** aus.

Auf dem Bildschirm erscheint ein Standard Öffnen-Dialog, in dem Sie angeben, welche Backup-Datei (.4bk) bzw. welches Logbuch (.journal) wiederhergestellt werden soll.

2. Wählen Sie die gewünschte Datei und klicken auf die Schaltfläche **Öffnen**.

Auf dem Bildschirm erscheint folgendes Dialogfenster. Hier können Sie den Zielordner angeben, in dem die Dateien wiederhergestellt werden:



4D stellt die Dateien standardmäßig im Ordner mit Namen Archivname (ohne Endung) her, die auf derselben Ebene wie das Archiv liegt. Sie können auf die Schaltfläche [...] klicken, um ein anderes Ziel zu wählen.

3. Klicken auf die Schaltfläche **Wiederherstellen**.

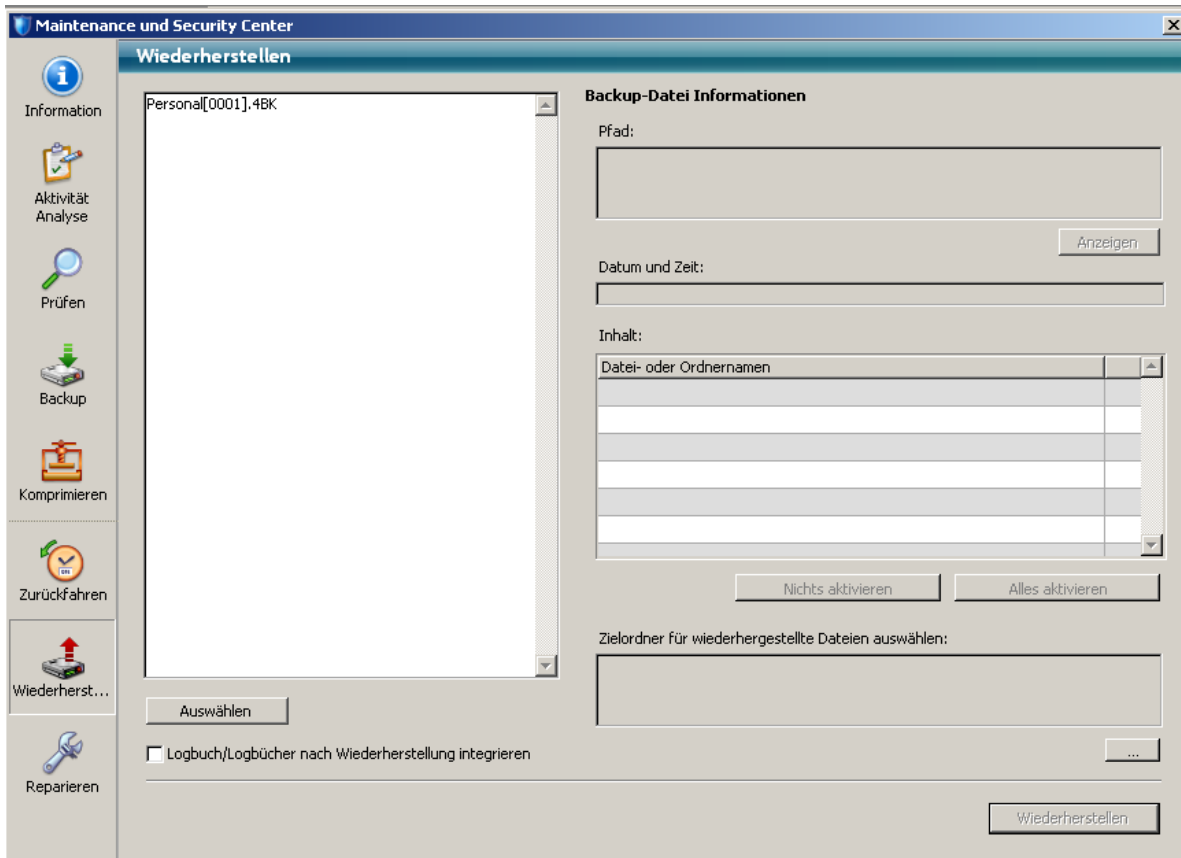
Hat das aktuelle Logbuch bzw. Logbuch des Backup dieselbe Nummer wie das Backup, das im gleichen Ordner

gespeichert ist, prüft 4D seinen Inhalt. Enthält es Operationen, die nicht in der Datendatei enthalten sind, fragt das Programm, ob Sie diese Operationen integrieren wollen. Die Integration erfolgt automatisch, wenn die Option zum Integrieren des Logbuchs markiert ist. Weitere Informationen dazu finden Sie im Abschnitt **Automatische Wiederherstellung**.

- (Optional) Klicken Sie auf die Schaltfläche **OK**, um das Logbuch in die wiederhergestellte Datenbank zu integrieren. Sind Wiederherstellen und Integrieren korrekt abgelaufen, zeigt 4D die Meldung, dass die Operation erfolgreich abgeschlossen wurde.
- Klicken Sie auf die Schaltfläche **OK**.
Der Zielordner wird angezeigt. 4D legt während der Wiederherstellung alle Backup-Dateien in den Zielordner, egal an welcher Stelle auf der Festplatte die Originaldateien beim Starten des Backup liegen. So finden Sie die Dateien leichter.

Backup manuell wiederherstellen (MSC)

Sie können ein Archiv der aktuellen Datenbank manuell über das Maintenance- und Security-Center (MSC) wiederherstellen. Hier können Sie auch verschiedene Optionen zur Durchführung festlegen:



Der linke Bereich enthält die vorhandenen Backups der Datenbank. Sie können auch unter dem linken Bereich auf die Schaltfläche **Auswählen** klicken, um ein Archiv an anderer Stelle zu öffnen. Es wird dann zur Liste der Archive hinzugefügt.

Wählen Sie hier ein Backup aus, erscheinen im rechten Teil des Fensters Informationen über dieses Backup:

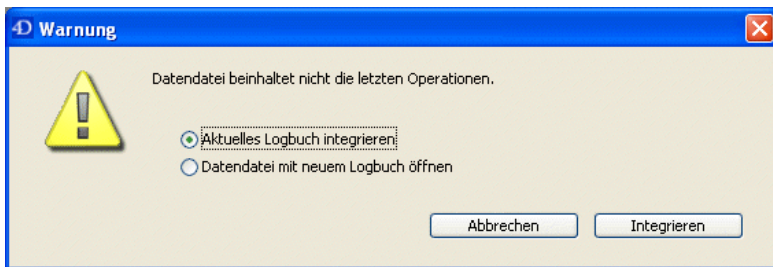
- Pfad:** Vollständiger Pfadname der gewählten Backup-Datei. Mit der Schaltfläche **Anzeigen** öffnen Sie die Backup-Datei in einem Systemfenster.
- Datum und Zeit:** Datum und Uhrzeit des Backup.
- Inhalt:** Inhalt der Backup-Datei. Neben jedem Eintrag in der Liste gibt es ein Ankreuzfeld, über das Sie angeben, ob er wiederhergestellt werden soll. Über die Schaltflächen **Nichts aktivieren** oder **Alles aktivieren** stellen Sie die Liste der Einträge zusammen, die Sie wiederherstellen wollen.
- Zielordner für wiederhergestellte Dateien auswählen:** Ordner, in dem die wiederhergestellten Dateien abgelegt werden. 4D legt die wiederhergestellten Dateien standardmäßig in einem Ordner mit Namen Archivname (ohne Endung) ab, der auf derselben Ebene wie das Archiv liegt. Wollen Sie das Ziel ändern, klicken Sie unter dem Bereich auf die Schaltfläche [...] und geben einen anderen Ordner an.

Mehrere Logbücher nacheinander integrieren

Sie können mehrere Logbücher hintereinander in die Datenbank integrieren. Haben Sie z.B. 4 Logbücher passend zu 4 Backups der Datenbank, können Sie das erste Backup wiederherstellen und dann die Logbücher der Reihe nach integrieren. Auf diese Weise können Sie z.B. eine Datei wiederherstellen, auch wenn die letzten Backup-Dateien fehlen. Ist diese Option markiert, zeigt 4D nach dem Wiederherstellen den Standard Öffnen-Dialog. Hier können Sie das entsprechende Logbuch auswählen. Der Öffnen-Dialog erscheint erneut nach jeder Integration, bis er abgebrochen wird.

Logbuch manuell integrieren

Ist die Option für automatisches Integrieren des Logbuchs nicht aktiv, erscheint beim Öffnen der Datenbank eine Meldung, wenn 4D feststellt, dass das Logbuch mehr Operationen enthält als in der Datenbank ausgeführt wurden (siehe voriger Abschnitt **Mehrere Logbücher nacheinander integrieren**).



Damit dieser Vorgang funktioniert, muss 4D auf das Logbuch an seinem aktuellen Speicherort zugreifen können. Sie entscheiden selbst, ob Sie das aktuelle Logbuch integrieren wollen. Führen Sie diese Aktion nicht aus, können Sie das Reproduzieren fehlerhafter Operationen in den Daten vermeiden.

Backup Journal

Damit das Nachvollziehen und Überprüfen von Backups einfacher ist, schreibt das Backup-Modul einen Abriss jeder ausgeführten Operation in eine spezielle Datei, die eine Art Aktivitätsjournal ist. Hier werden – wie in einem Bordbuch – alle Operationen der Datenbank (Backup, Wiederherstellen, Integration des Logbuchs) eingetragen, und ob sie nach Zeitplan oder manuell ausgeführt wurden. Zusätzlich erscheinen Datum und Uhrzeit der Ausführung.

Das Backup-Journal hat den Namen "Backup Journal.txt" und liegt im Ordner "Logs" der Datenbank.

Das Backup-Journal lässt sich mit einem beliebigen Texteditor öffnen. Zur besseren Übersicht enthält das Journal Einrückungen, jede Informationszeile endet mit einer Zeilenschaltung.

Größe des Journal verwalten

Bei manchen Backup-Strategien, z.B. wenn unzählige Anhänge mitgesichert werden, kann das Backup-Journal rasch zu einem sehr großen Dokument werden.

Der Entwickler hat zwei Möglichkeiten, um die Größe zu steuern:

- **Automatisches Backup:** Vor jedem Backup prüft das Programm die Größe des aktuellen Backup-Journal. Übersteigt es 10 MB, wird die aktuelle Datei archiviert und eine neue angelegt. Archivierte Dateien haben den Namen "Backup Journal[xxx].txt", wobei xxx eine Zahl zwischen 1 und 999 ist. Ist die Dateinummer 999 erreicht, beginnt die Nummerierung wieder bei 1 und die vorhandenen Dateien werden ersetzt.
- **Menge an gespeicherter Information reduzieren:** Dazu müssen Sie lediglich den Wert des Schlüssels `VerboseMode` in der Datei `Backup.xml` der Datenbank verändern. Dieser Schlüssel ist standardmäßig auf `Wahr` gesetzt. Setzen Sie den Wert auf `Falsch`, wird im Backup-Journal nur die Hauptinformation gespeichert. Das sind Datum und die jeweilige Startzeit der Operation sowie gefundene Fehler. Weitere Informationen zu den XML Keys für die Backup Konfiguration finden Sie im Handbuch *XML Keys-Backup* (PDF).

Backup.xml

Die Einstellungen des Backup von 4D sind als XML Datei verfügbar, die der Datenbank zugeordnet ist. 4D verwendet die Daten dieser Datei zum Setzen der Backup-Optionen, wenn ein Backup gestartet wird. Das sind die Optionen in den Datenbank-Eigenschaften. Diese Datei kann auch zum Lesen oder Setzen weiterer Optionen dienen, z.B. die Menge an gespeicherter Information im Backup-Journal.










Die XML Backup Konfigurationsdatei lautet **Backup.xml**. Sie wird automatisch an folgender Stelle angelegt:

`Datenbankordner/Preferences/Backup/Backup.xml`

Datenbankordner ist der Ordner mit der Strukturdatei der Datenbank. Die Unterordner `Preferences/Backup` werden bei Bedarf automatisch angelegt.

Weitere Informationen zu den XML Keys für die Backup Konfiguration finden Sie im Handbuch **4D XML Keys Backup**.

Maintenance und Security Center

-  Überblick
-  Seite Information
-  Seite Aktivitätsanalyse
-  Seite Prüfen
-  Seite Backup
-  Seite Komprimieren
-  Seite Zurückfahren
-  Seite Wiederherstellen
-  Seite Reparieren

Überblick

Das Fenster Maintenance und Security Center (MSC) enthält umfangreiche Werkzeuge zum Prüfen, Warten, Sichern und Komprimieren von Daten- und Strukturdateien. Das MSC ist von allen 4D Programmen aus verfügbar, also 4D Developer, 4D Server oder 4D Desktop.

Es gibt mehrere Wege, das MSC-Fenster zu öffnen. Die Art des Zugriffs bestimmt auch die Art, wie die Datenbank geöffnet wird, entweder im Wartungs- oder im Standardmodus. Im Wartungsmodus wird die Datenbank nicht durch 4D geöffnet, sondern nur ihre Referenz an MSC geliefert. Im Standardmodus wird die Datenbank von 4D geöffnet.

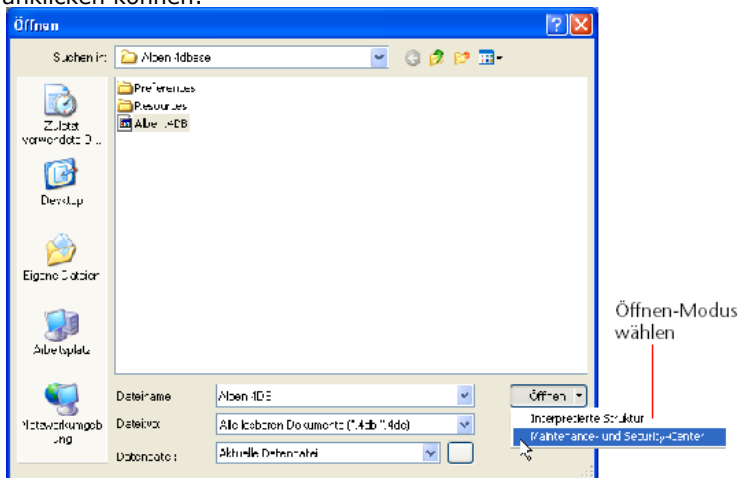
Anzeige im Wartungsmodus

Im Wartungsmodus erscheint nur das MSC-Fenster, d.h. die Datenbank wird nicht vom 4D Programm geöffnet. Das hat zur Folge, dass auch Zugriff auf Datenbanken möglich ist, die sich wegen Beschädigung nicht im Standardmodus öffnen lassen. Für Operationen wie Komprimieren, Reparieren u.ä. müssen Sie die Datenbank auch im Wartungsmodus öffnen. Weitere Informationen dazu finden Sie im Abschnitt **Zugriffsrechte**.

Es gibt zwei Möglichkeiten, das MSC im Wartungsmodus zu öffnen:

- **Über den Standard-Öffnen Dialog**

Haben Sie eine Strukturdatei gewählt, enthält der Standarddialog zum Öffnen der Datenbank in der rechten unteren Ecke die Schaltfläche **Öffnen** mit einem PopUp-Menü, in dem Sie die Option **Maintenance- und Security-Center** anklicken können:



Sie können dann einfach die entsprechende Datenbank wählen und auf die Schaltfläche **Öffnen** klicken.

- **Über das Menü Hilfe oder die Schaltfläche MSC in der Werkzeugleiste** (Die Datenbank ist nicht geöffnet)



Auf dem Bildschirm erscheint der Standard-Öffnen Dialog, in dem Sie die entsprechende Datenbank auswählen können. Dabei wird die Datenbank nicht von 4D geöffnet.

Anzeige im Standardmodus

Im Standardmodus wird eine Datenbank geöffnet. Hier sind nicht alle Wartungsfunktionen verfügbar. Es gibt mehrere Möglichkeiten, den Standardmodus aufzurufen:

- Über das Menü **Hilfe** oder die Schaltfläche **MSC** in der Werkzeugleiste im Designmodus



- Damit öffnen Sie das MSC-Fenster. Diese Funktionalität ist im Anwendungsmodus nicht verfügbar.
- Über die Standardaktion "Öffne MSC", die Sie einem erstellten Menübefehl zuordnen können (siehe **Standardaktionen**).
- Über den 4D Befehl **OPEN SECURITY CENTER**

Zugriffsrechte

Die MSC Funktionen sind je nach Programmart, dem Öffnen-Modus in MSC oder dem Benutzerprofil (bei aktivierten Kennwörtern) unterschiedlich verfügbar:

- Funktionen, welche die Struktur betreffen, wie Prüfen, Reparieren und Komprimieren, sind nur über die 4D im lokalen Modus und 4D Server verfügbar. In remote 4D und 4D Desktop sind die entsprechenden Schaltflächen und

Tabulatoren ausgeblendet.

- Information über den Inhalt von Daten- und Strukturdateien sind nur verfügbar, wenn die Datenbank geöffnet ist, d.h. MSC muss im Standardmodus geöffnet sein.
- Die Funktionen Datenkomprimierung, Zurückfahren, Wiederherstellen und Reparieren sind nur bei nicht-geöffneten Datendateien verwendbar, d.h. MSC muss im Wartungsmodus geöffnet sein. Versuchen Sie, diese Funktionen bei im Standardmodus geöffneter Datenbank aufzurufen, erscheint ein Dialogfenster, um das Programm im Wartungsmodus erneut aufzurufen.
- Sind Kennwörter aktiviert, können nur Administrator und Designer auf die Funktionen Datenkomprimierung, Zurückfahren, Wiederherstellen und Reparieren zugreifen.

Die Seite "Information" liefert Informationen über die 4D Umgebung und die Systemumgebung, die Dateien der Datenbank und der Anwendung. Über die Registerkarten am oberen Rand können Sie die einzelnen Seiten aufrufen.

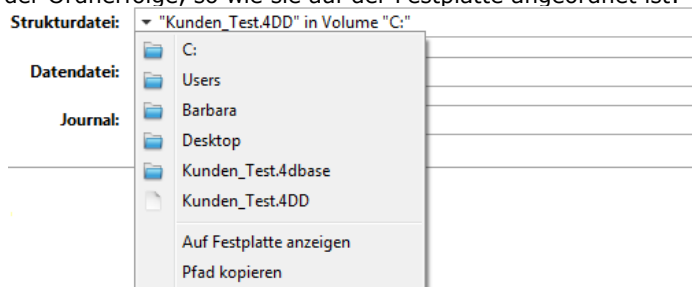
Programm

Der obere Bereich gibt Name, Version und Speicherort des Programms und des aktiven 4D Ordners an. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* unter der Funktion **Get 4D folder**.

Der mittlere Bereich gibt Name und Speicherort der Struktur- und Datendatei der Datenbank und des Logbuchs (wenn vorhanden) an.

Der untere Bereich zeigt den Namen des 4D Lizenzinhabers, die Art der Lizenz, und bei aktivierten Kennwörtern auch den Anwendernamen. Ist das nicht der Fall, erscheint als Anwendername *Designer*.

- **Pfadnamen anzeigen und auswählen:** Auf der Seite Programm erscheinen die Pfadnamen in PopUp-Menüs mit der Ordnerfolge, so wie sie auf der Festplatte angeordnet ist:



Wählen Sie einen Eintrag (Festplatte oder Ordner), erscheint er in einem neuen Systemfenster. Der Eintrag *Pfad kopieren* kopiert den vollständigen Pfadnamen als Text mit den Trennern der jeweiligen Plattform in die Zwischenablage.

- **"Licenses" Ordner**

Diese Schaltfläche zeigt den Inhalt des aktiven **Licenses** Ordner in einem neuen Systemfenster an.

Alle Lizenzdateien, die in Ihrer 4D Umgebung installiert sind, sind in diesem Ordner auf Ihrer Festplatte zusammengefasst. Beim Öffnen über einen Web Browser erscheinen die Informationen über die enthaltenen Lizenzen und ihre Merkmale.

Der Speicherort des Ordners Licenses variiert je nach Version oder Sprache Ihres Betriebssystems. Weitere Informationen dazu finden Sie unter der 4D Funktion **Get 4D folder**.

Hinweis: Sie können auf diesen Ordner auch zugreifen, wenn Sie im Menü **Hilfe>Lizenz aktualisieren** aufrufen. Klicken Sie in Dialogfenster Lizenz aktualisieren auf die Schaltfläche Lizenzordner, öffnet 4D den Ordner Licenses in einem standardmäßigen Systemfenster. Ist Ihr 4D Programm aktiviert, muss mindestens eine Lizenzdatei darin liegen (Datei im HTML-Format). Haben Sie mehrere Produkte, Plug-Ins oder Extensions aktiviert, erscheinen mehrere Lizenzdateien.

Tabellen

Diese Seite gibt einen Überblick über die Tabellen in Ihrer Anwendung.

The screenshot shows the 'Informationen' page in the Maintenance and Security Center. The table lists the following data:

ID	Tabellen	Datensätze	Felder	Indizes	Größe der Adresstab...
1	Kunden	1	3	2	1
2	Rechnungen	1	4	2	1
3	Positionen	1	4	1	1
4	Artikel	1	3	2	1
Gesamt					
	4	4	14	7	4

Die Seite listet alle Tabellen der Datenbank - auch die ausgeblendeten - mit den dazugehörigen Merkmalen:

- **ID:** Interne Nummer der Tabelle
 - **Tabellen:** Name der Tabelle. Namen von gelöschten Tabellen erscheinen in Klammern (wenn sie noch im Papierkorb sind).
 - **Datensätze:** Gesamtanzahl der Datensätze für die Tabelle. Ist ein Datensatz beschädigt oder nicht lesbar, erscheint anstelle der Zahl ein Fehler. Dann können Sie abwägen, ob Sie die Tools zum Prüfen und Reparieren einsetzen.
 - **Felder:** Anzahl der Felder in der Tabelle. Ausgeblendete Felder werden mitgezählt, gelöschte Felder dagegen nicht.
 - **Indizes:** Anzahl der verschiedenen Indizes in der Tabelle
 - **Größe der Adresstabelle:** Größe der Adresstabelle für jede Tabelle. In der Adresstabelle wird pro angelegtem Datensatz in der Tabelle intern ein Element gespeichert. Es verbindet Datensätze mit ihrer physikalischen Adresse. Aus Performance Gründen wird sie beim Löschen von Datensätzen nicht angepasst, so dass ihre Größe von der aktuellen Anzahl der Datensätze in der Tabelle abweichen kann. Bei einem deutlichen Unterschied können Sie die Größe der Adresstabelle über die Option "Komprimiere Adresstabelle" optimieren (siehe [Seite Komprimieren](#)).
- Hinweis:** Der Unterschied zwischen Größe der Adresstabelle und Anzahl der Datensätze kann auch bei einer Störung während dem Leeren des Cache entstehen.

Hinweis: Information auf dieser Seite ist im Standardmodus und im Wartungsmodus verfügbar.

Daten und Struktur

Die Seiten **Daten** und **Struktur** zeigen Informationen über die Füll- und Auslastungsrate der Struktur- und Datendatei der Datenbank.

Hinweise:

- Diese Seiten sind im Wartungsmodus nicht verfügbar
- Die Seite **Daten** berücksichtigt keine Daten, die außerhalb der Datendatei gespeichert sind (siehe [Daten extern speichern](#)).
- Die Seite **Struktur** ist nur in den Programmen 4D und 4D Server verfügbar

Diese Angaben werden grafisch dargestellt:

Maintenance und Security Center

Information

Information

Programm Tabellen Daten Struktur

Struktur:

Freier Speicher (0,97 MB)	1,31 MB
Benutzter Speicher (0,33 MB)	

Struktur-Indizes:

Freier Speicher (0,06 MB)	0,18 MB
Benutzter Speicher (0,12 MB)	

Die Gesamtgröße dieser Dateien (Struktur + Index) ist 1,50 MB.
Benutzt wird 31%.

Die Struktur muß nicht komprimiert werden.

Aktivität Analyse

Prüfen

Backup

Komprimieren

Zurückfahren

Wiederherst...

Reparieren

Zu stark fragmentierte Dateien senken die Performance der Festplatte und somit der Datenbank. Bei zu niedriger Auslastungsrate zeigt 4D ein Icon als gelbes Warndreieck und meldet, dass Komprimieren notwendig ist. Das Icon erscheint in der Schaltfläche **Information** und in der Registerkarte der entsprechenden Datei:



Das Warndreieck erscheint auch in der Schaltfläche **Komprimieren** auf der **Seite Komprimieren**:



Auf der Seite **Aktivitätsanalyse** des MSC können Sie den Inhalt des aktuellen Logbuchs betrachten. Diese Funktion ist hilfreich, um die Verwendung der Datenbank zu analysieren oder die Operationen herauszufinden, die Fehler verursachen oder die Funktionsweise beeinträchtigen. Läuft die Anwendung im Client-/Server-Modus, können Sie auch die Operationen auf den Arbeitsstationen überprüfen.

Hinweis: Sie können die in den Daten der Anwendung ausgeführten Operationen auch zurückfahren (rollback). Weitere Informationen dazu finden Sie auf der [Seite Zurückfahren](#).

The list below shows all the performed operations recorded in the log file since the last backup.

Operation	Action	Table	Primary Key/BL...	Process	Size	Date	Hour	User	Values	Recc
1	Sequence...	Employees		23		1/9/2014	5:00:34		12	
2	Addition	Employees		23	68	1/9/2014	5:00:34	paul ; ; ...	0	
3	Sequence...	Departme...		23		1/9/2014	5:02:02		2	
4	Addition	Departme...	HR	23	23	1/9/2014	5:02:02	HR ; Jerry ...	0	
5	Sequence...	Departme...		23		1/9/2014	5:02:11		3	
6	Addition	Departme...	Accounting	23	37	1/9/2014	5:02:11	Accounti...	1	
7	Sequence...	Departme...		23		1/9/2014	5:02:26		4	
8	Addition	Departme...	IT	23	21	1/9/2014	5:02:26	IT ; Fred ; ...	2	
9	Sequence...	Postal Rat...		23		1/9/2014	5:06:03		3	
10	Addition	Postal Rat...	2	23	36	1/9/2014	5:06:03	France ; 1....	0	
11	Sequence...	Postal Rat...		23		1/9/2014	5:06:14		4	
12	Addition	Postal Rat...	3	23	40	1/9/2014	5:06:14	Ger,many...	1	
13	Sequence...	Postal Rat...		23		1/9/2014	5:06:28		5	
14	Addition	Postal Rat...	4	23	46	1/9/2014	5:06:28	Switzerlan...	2	
15	Sequence...	Postal Rat...		23		1/9/2014	5:06:38		6	
16	Addition	Postal Rat...	5	23	34	1/9/2014	5:06:38	Spain ; 1.1...	3	
17	Sequence...	Postal Rat...		23		1/9/2014	5:06:48		7	
18	Addition	Postal Rat...	6	23	40	1/9/2014	5:06:48	Portugal ; ...	4	
19	Sequence...	Postal Rat...		23		1/9/2014	5:07:27		8	
20	Addition	Postal Rat...	7	23	22	1/9/2014	5:07:27	; 0 ; ; ; 7	5	
21	Sequence...	Departme...		23		1/9/2014	5:07:40		5	

Hinweis: Nur Administrator und Designer der Anwendung können auf die Überprüfungsfunktion des Logbuchs zugreifen. Jede im Logbuch gespeicherte Operation erscheint in einer Zeile. Die Spalten liefern verschiedene Information zur Operation. Sie können die standardmäßige Anzeige der Spalte ändern — zum Beispiel die Werte der Felder anzeigen. Dazu wählen Sie das Kontextmenü des Fensters window (siehe unten). Außerdem wird mit Datum und Uhrzeit angezeigt, wann die Datendatei geöffnet bzw. geschlossen wurde.

Über diese Informationen können Sie die Quelle und den Kontext jeder Operation identifizieren

- **Operation#:** Sequenznummer der Operation im Logbuch
- **Aktion:** Art der in den Daten ausgeführten Operation. Diese Spalte kann eine der nachfolgenden Operationen anzeigen:
 - *Öffnen der Datendatei:* Datendatei öffnen
 - *Schließen der Datendatei:* Eine geöffnete Datendatei wird geschlossen
 - *Erstellen eines Kontexts:* Einen Prozess erstellen, der einen Ausführungskontext angibt.
 - *Schließen eines Kontexts:* Prozess geschlossen
 - *Hinzufügen:* Creation and storage of a record.
 - *BLOB hinzufügen:* Storage of a BLOB in a BLOB field.
 - *Löschen:* Datensatz gelöscht
 - *Ändern:* Datensatz geändert
 - *Starten der Transaktion:* Transaktion gestartet
 - *Bestätigen der Transaktion:* Transaktion bestätigt
 - *Abbrechen der Transaktion:* Transaktion annulliert
- **Tabelle:** Tabelle, zu welcher der hinzugefügte/gelöschte/geänderte Datensatz bzw. das BLOB gehört.
- **Primärschlüssel/BLOB:** Inhalt des Primärschlüssels für jeden Datensatz (besteht der Primärschlüssel aus mehreren Feldern, sind die einzelnen Werte durch Strichpunkte voneinander getrennt) oder Sequenznummer des BLOB, welche an der Operation beteiligt ist.
- **Prozess:** Interne Nummer des Prozesses, in dem die Operation ausgeführt wurde. Diese interne Nummer entspricht dem Kontext der Operation.

- **Größe:** Größe (in Bytes) der Daten, die durch die Operation bearbeitet werden.
- **Datum und Uhrzeit:** Datum und Uhrzeit, zu der die Operation ausgeführt wurde.
- **Anwender:** Name des Anwenders, der die Operation ausgeführt hat. Im Client-/Server-Modus erscheint der Name der Arbeitsstation; im Einzelplatz die Kennung des Anwenders.
Sind die 4D Kennwörter nicht aktiviert, ist diese Spalte leer.
- **Werte** (diese Spalte erscheint standardmäßig nicht): Werte der Felder für den Datensatz bei Hinzufügen oder Änderung. Die Werte werden durch ";" voneinander getrennt. Nur Werte in alphanumerischer Form erscheinen.
- **Datensätze:** Datensatznummern

Klicken Sie am unteren Rand auf die Schaltfläche **Analyse**, um den Inhalt des aktuellen Logbuchs der ausgewählten Datenbank anzuzeigen. Es hat in 4D standardmäßig den Namen *Datenbankname.journal*. Mit der Schaltfläche **Auswählen** wählen und öffnen Sie ein anderes Logbuch der Datenbank. Mit der Schaltfläche **Exportieren** exportieren Sie den Inhalt der Datei als Text.

Auf dieser Seite können Sie die Daten- und Strukturintegrität überprüfen. Die Überprüfung lässt sich in Datensätzen und/oder Indizes durchführen, sowie in Designobjekten, wie Methoden, Formulare, u.ä.. Diese Funktionalität prüft nur die Datenintegrität. Bei Fehlern oder notwendigen Reparaturen erhalten Sie eine Meldung, die **Seite Reparieren** zu verwenden.

Aktionen

Die Seite enthält vier Schaltflächen für Aktionen, über die Sie direkt auf die Überprüfungsfunktionen zugreifen können.

Datensätze und/oder Indizes prüfen

- **Datensätze und Indizes prüfen:** Startet den kompletten Prozess zum Überprüfen der Daten.
- **Nur Datensätze prüfen:** Startet nur den Prozess zum Überprüfen der Datensätze, d.h. Indizes werden nicht geprüft.
- **Nur Indizes prüfen:** Startet nur den Prozess zum Überprüfen der Indizes, d.h. Datensätze werden nicht geprüft.

Hinweis: Sie können Datensätze und Indizes auch im Detail Tabelle für Tabelle überprüfen. Weitere Informationen dazu finden Sie im unteren Abschnitt.

Anwendung prüfen

Die Aktion **Anwendung prüfen** startet den Prozess zum Überprüfen aller Objekte im Designmodus. Das sind Tabellen, Methoden, Formulare, usw.

Überholte Bilder

Ab Version v16 zeigt diese Aktion auch Warnungen zu Bildern, die das veraltete PICT Format benutzen oder enthalten. Weitere Informationen dazu finden Sie im Abschnitt **Bilder in PICT Format** des Handbuchs *Überholte oder entfernte Features*. Diese Warnungen können sowohl statische Bilder als auch Bilder in der Bildbibliothek oder in Formularobjekten betreffen.

Hinweis: Die Operation **Strukturdatei reparieren** hat keine Auswirkung auf Bilder mit veralteten Formaten und die gleichen Warnungen erscheinen im erzeugten Logbuch. Es liegt bei Ihnen, diese Bilder bei Bedarf zu entfernen oder zu ersetzen.

Überholte Zeichen in Namen (".", "[", und "]")

Ab 4D v16 R4 ist die Verwendung von Punkten (.) bzw. eckigen Klammern ([]) in folgenden Elementen nicht mehr zulässig:

- Variablennamen
- Tabellennamen
- Feldnamen
- Namen von Projektmethoden

Diese Regel ist erforderlich, da es sonst Überschneidungen mit der Objekt Notation gibt. Weitere Informationen dazu finden Sie im Abschnitt **Objektnotation verwenden**.

Um Entwickler beim Anpassen Ihrer Anwendung zu unterstützen, überprüft die Aktion **Anwendung prüfen** automatisch solche Zeichen in Namen für Variablen, Tabellen, Felder und Methoden. Werden solche Zeichen gefunden, protokolliert das MSC dies und das Logbuch enthält relevante Warnungen:

6. Check Project Methods [1 warning]

The method name "tes.t.test" contains dots or square brackets

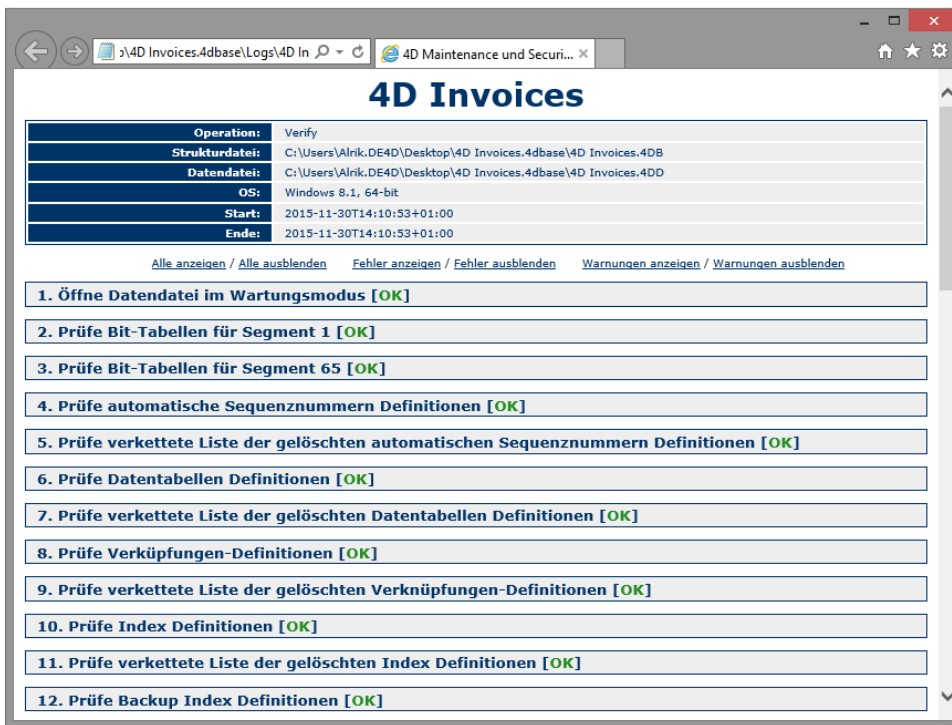
In diesem Fall raten wir, diese Elemente in Ihrer Anwendung umzubenennen.

Logbuch öffnen

Bei allen Überprüfungsarten generiert 4D ein Logbuch im Ordner **Logs** der Datenbank. Hier erscheinen alle durchgeführten Überprüfungen und evtl. gefundene Fehler. Gibt es keine Fehler, wird das durch [OK] angezeigt. Das Logbuch wird im XML- und HTML-Format angelegt und hat den Namen *DatabaseName_Verify_log <yyyy-mm-dd hh-mm-ss>.xml*:

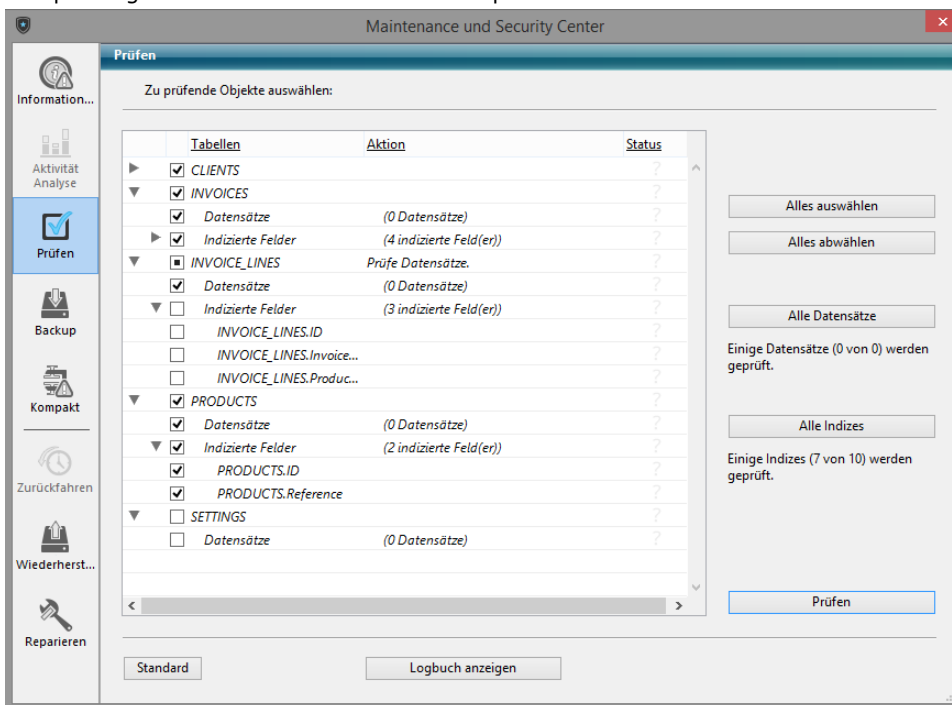
- *<DatabaseName>* ist der Name der Strukturdatei ohne Endung, zum Beispiel "Rechnungen"
- *<yyyy-mm-dd hh-mm-ss>* ist der Zeitstempel der Datei. Er basiert auf der lokalen Systemzeit, wenn die Wartungsoperation gestartet ist, zum Beispiel "2016-01-22 15-20-45".

Klicken Sie auf die Schaltfläche **Logbuch öffnen**, zeigt 4D das aktuellste Logbuch im standardmäßigen Browser des Rechners:



Tabellenliste

Klicken Sie auf die Schaltfläche **Tabellenliste** am unteren Rand, erscheint die Seite mit ausführlichen Angaben zur Überprüfung. Hier können Sie die aktuell zu prüfenden Datensätze und Indizes ansehen und auswählen:



Sie können die Prüfung auf bestimmte Objekte oder Teile einer Tabelle beschränken, um so Zeit bei der Überprüfung zu sparen.

Die Hauptliste zeigt alle Tabellen der Datenbank. Sie können für jede Tabelle die Überprüfung auf Datensätze bzw. Indizes eingrenzen. Klicken Sie auf den Pfeil vor der Tabelle, um die Tabelle oder indizierte Datenfelder aufzuklappen und markieren Sie die Ankreuzfelder je nach gewünschter Aktion. Standardmäßig ist alles ausgewählt.

Sie können auch am rechten Rand die Schaltflächen **Alles auswählen**, **Alles abwählen**, **Alle Datensätze** und **Alle Indizes** dafür verwenden.

Die Spalte "Aktion" gibt für jede Zeile die auszuführenden Operationen an. Bei aufgeklappter Tabelle zeigen die Zeilen "Datensätze" und "Indizierte Felder" die Anzahl der betroffenen Einträge an.

Die Spalte „Status“ zeigt den Überprüfungsstatus für jeden Eintrag mit dem entsprechenden Symbol an:

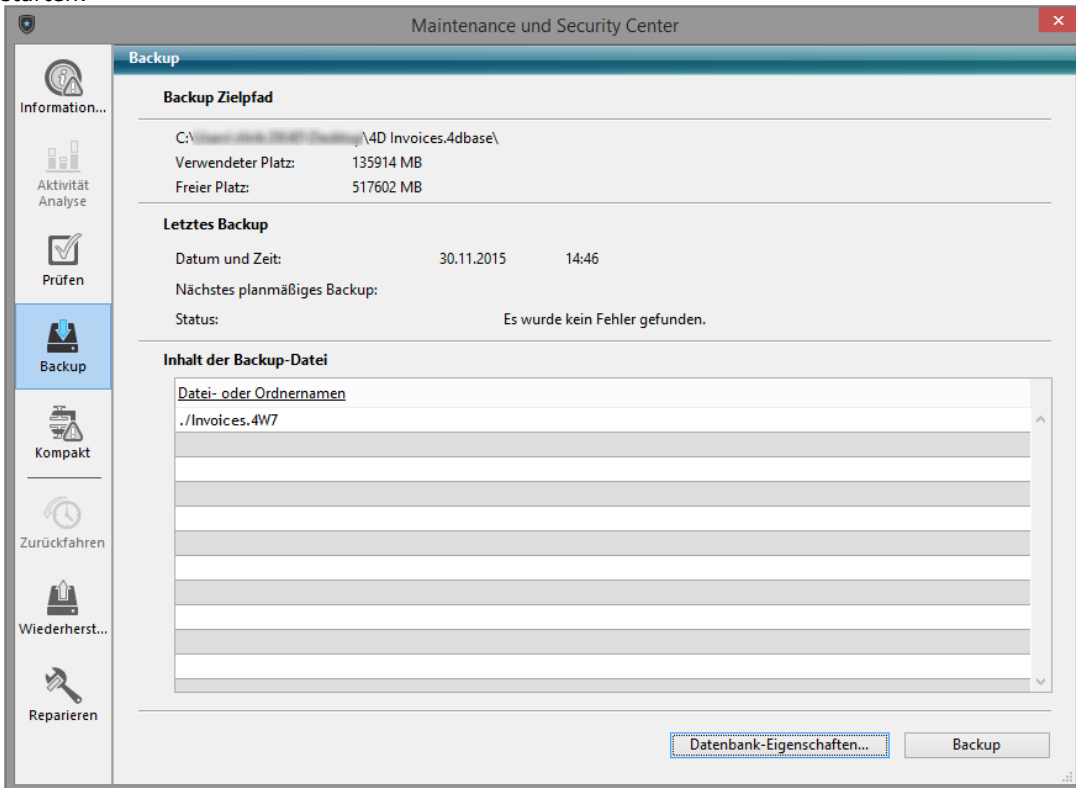
- Überprüfung ausgeführt, keine Probleme
- Überprüfung ausgeführt, Probleme gefunden
- Überprüfung zum Teil ausgeführt
- Überprüfung nicht ausgeführt

Klicken Sie am unteren Rand rechts auf die Schaltfläche **Prüfen**, um die Überprüfung zu starten oder links auf die Schaltfläche **Standard**, um zur Standardseite zurückzukehren

Klicken Sie auf die Schaltfläche **Logbuch anzeigen**, zeigt 4D das Logbuch im standardmäßigen Browser des Rechners (siehe oberen Abschnitt **Logbuch öffnen**).

Hinweis: Die Standardseite berücksichtigt keine Änderungen, die auf der Detailseite gemacht wurden: Klicken Sie auf der Standardseite auf eine Schaltfläche zum Überprüfen, werden alle Einträge überprüft. Die auf der Detailseite definierten Einstellungen bleiben jedoch von einer Sitzung zur nächsten erhalten.

Die Seite **Backup** des MSC zeigt die Backup-Einstellungen für die Datenbank. Hier können Sie auch ein manuelles Backup starten:



Diese Seite ist in drei Bereiche unterteilt:

- **Backup Zielpfad:** Dieser Bereich gibt den Speicherort der Backup-Datei der Datenbank an, sowie den verwendeten und den freien Platz auf der Backup-Festplatte.
- **Letztes Backup:** Hier erscheinen Datum und Uhrzeit des zuletzt für die Datenbank ausgeführten Backup, egal ob automatisch oder manuell.
- **Inhalt der Backup-Datei:** Dieser Bereich zeigt die in der Backup-Datei enthaltenen Dateien und Ordner

Mit der Schaltfläche **Backup** starten Sie ein manuelles Backup. Weitere Informationen über Backups in 4D finden Sie im Abschnitt **Backup der Anwendung ausführen**.

Auf dieser Seite können Sie keine Backup-Parameter verändern. Dazu müssen Sie am unteren Rand auf die Schaltfläche **Einstellungen** klicken.

Auf dieser Seite können Sie die Funktionen zum Komprimieren von Daten- und Strukturdatei verwenden (siehe [Beschreibung der 4D Dateien](#)).

Warum Dateien komprimieren?

Komprimieren der Dateien erfüllt folgende Anforderungen:

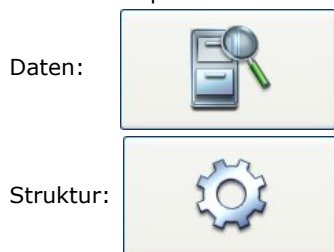
- Größe verringern und Dateien optimieren: Die Dateien können nicht-verwendete Leerräume enthalten, sog. Löcher. Sie entstehen, wenn Sie Datensätze, Formulare, o.ä. löschen. In der Regel verwendet 4D diese Leerräume soweit wie möglich erneut. Da jedoch die Datengröße unterschiedlich ist, entstehen durch sukzessives Löschen oder Ändern unweigerlich nicht-verwendete Leerräume. Dasselbe passiert, wenn eine große Menge Daten gerade gelöscht wurde: die leeren Stellen bleiben in der Datei unausgefüllt. Das Verhältnis zwischen Größe der Datendatei und derzeit für die Daten genutztem Platz ist die Auslastungsrate der Daten. Eine zu geringe Rate ist einerseits Platzverschwendung, und kann andererseits die Performance der Datenbank beeinträchtigen. Hier schafft Komprimieren Abhilfe, denn dadurch wird das Speichern der Daten neu organisiert und optimiert, d.h. die Löcher werden entfernt. Der Bereich "Information" fasst die Daten im Hinblick auf Fragmentierung zusammen und gibt die notwendigen Operationen an. Die Unterseiten **Daten und Struktur** zur Seite Information des MSC zeigen die aktuelle Fragmentierung der Dateien der Datenbank im Detail.
- Daten vollständig aktualisieren mit der aktuellen Formatierung, die in der Strukturdatei festgelegt ist. Das ist hilfreich, wenn Daten aus der gleichen Tabelle in unterschiedlichen Formaten gespeichert wurden, z.B. nach einer Änderungen in der Struktur der Anwendung.

Hinweis: Komprimieren ist nur im Wartungsmodus verfügbar. Versuchen Sie, diese Operation im Standardmodus auszuführen, erhalten Sie die Meldung, dass die Datenbank geschlossen und im Wartungsmodus neu gestartet wird. Sie können jedoch eine Datendatei komprimieren, die nicht von der Datenbank geöffnet ist (siehe Abschnitt [Komprimiere Datensätze und Indizes](#)).

Daten oder Strukturdatei komprimieren

Die standardmäßige Komprimierung ist für die Daten- und Strukturdatei gleich.

Um das Komprimieren direkt zu starten, klicken Sie im MSC-Fenster auf die entsprechende Schaltfläche:



Hinweis: Da beim Komprimieren die Originaldatei dupliziert wird, ist die Schaltfläche inaktiv, wenn auf der Festplatte nicht genügend Speicherplatz vorhanden ist.

Der Vorgang komprimiert die Hauptdatei sowie alle evtl. vorhandenen Index-Dateien.

4D kopiert die Originaldateien und setzt sie in einen Ordner mit Namen "Ersetzte Dateien (Komprimierung)", der neben der Originaldatei angelegt wird. Ist die Operation abgeschlossen, ersetzt die komprimierte Datei automatisch die Originaldatei. Die Datenbank ist sofort einsatzbereit.

Hinweise:

- Sie können dieses Standardverhalten für Datendateien ändern. Dazu verwenden Sie den erweiterten Modus.
- Führen Sie mehrere Komprimierungen durch, wird jedes Mal ein neuer Ordner angelegt. Er lautet beispielsweise "Ersetzte Dateien (Komprimierung)_1", "Ersetzte Dateien (Komprimierung)_2", etc.

Warnung: Die Originaldatei wird bei jedem Komprimierungsvorgang dupliziert, was den Anwendungsordner vergrößert. Sie sollten darauf achten, besonders auf OS X, wo 4D Anwendungsordner als Paket erscheinen, dass die Größe der Anwendungsdatei nicht exzessiv ansteigt. In diesem Fall sollten Sie die Kopien der Originaldatei im Paket per Hand entfernen, damit die Größe des Pakets im Rahmen bleibt.

Ist das Komprimieren abgeschlossen, erzeugt 4D ein Logbuch im Ordner **Logs** der Datenbank. Es wird im XML- und HTML-Format erstellt und hat den Namen "`<DatabaseName>_Compact_Log_<yyyy-mm-dd hh-mm-ss>.xml`":

- `<DatabaseName>` ist der Name der Strukturdatei ohne Endung, zum Beispiel "Rechnungen"
- `<yyyy-mm-dd hh-mm-ss>` ist der Zeitstempel der Datei. Er basiert auf der lokalen Systemzeit, wenn die Wartungsoperation gestartet ist, zum Beispiel "2016-01-22 15-20-45".

Hier können Sie alle ausgeführten Operationen ansehen. Klicken Sie auf die Schaltfläche **Logbuch öffnen**, zeigt 4D das Logbuch im standardmäßigen Browser des Rechners (siehe [Logbuch öffnen](#)).

Erweiterter Modus

Die Seite Komprimieren enthält am unteren Rand die Schaltfläche **Erweitert>**. Damit öffnen Sie eine Seite mit Optionen zum Komprimieren der Daten- und Strukturdatei.

Komprimiere Datensätze und Indizes

Der obere Bereich enthält den Pfadnamen der aktuellen Datendatei und die Schaltfläche [...] zum Auswählen einer anderen Datendatei. Klicken Sie auf diese Schaltfläche, erscheint ein Standard-Öffnen Dialog, um die gewünschte Datendatei zum Komprimieren auszuwählen. Sie muss zur geöffneten Strukturdatei passen. Bestätigen Sie das Dialogfenster, erscheint der Pfadname der zu komprimierenden Datei im Fenster Komprimieren.

Über die darunterliegende Schaltfläche [...] können Sie vor dem Komprimieren einen anderen Ort zum Speichern der Originaldateien angeben. Diese Möglichkeit ist insbesondere beim Komprimieren umfangreicher Dateien auf verschiedenen Festplatten hilfreich.

Erzwingen Aktualisierung aller Datensätze

Ist diese Option markiert, schreibt 4D während der Komprimierung jeden Datensatz für jede Tabelle gemäß seiner Beschreibung in der Struktur erneut. Ist diese Option nicht markiert, organisiert 4D lediglich das Speichern der Daten auf der Festplatte neu. Diese Option ist hilfreich in folgenden Fällen:

- Wenn Feldtypen in der Struktur der Anwendung nach der Dateneingabe geändert wurden. Sie haben z.B. ein Feld vom Typ Lange Ganzzahl in den Typ Zahl geändert. 4D erlaubt sogar Änderungen zwischen unterschiedlichen Typen (kann aber zu Datenverlust führen), z.B. lässt sich ein Feld vom Typ Zahl in den Typ Text ändern und umgekehrt. In diesem Fall konvertiert 4D bereits eingegebene Daten nicht rückwirkend; sie werden nur beim Laden und Sichern von Datensätzen konvertiert. Diese Option erzwingt die Konvertierung aller Daten.
- Wenn die Option externes Speichern für Daten vom Typ Text, Bild oder BLOB nach der Dateneingabe geändert wurde. Das kann insbesondere beim Konvertieren der Datenbanken in v13 passieren, da neue Optionen verfügbar sind (siehe **Daten extern speichern**). Auch hier konvertiert 4D bereits eingegebene Daten nicht rückwirkend. Mit dieser Option können Sie erzwingen, dass alle Datensätze aktualisiert werden, damit der neue Speicherungsmodus auch auf bereits eingegebene Datensätze angewandt wird.
- Wenn Tabellen oder Felder gelöscht wurden. In diesem Fall wird der Platz der entfernten Daten beim Komprimieren mit Aktualisieren aller Datensätze wieder verwendet und so die Dateigröße verringert.

Hinweis: Ist diese Option aktiv, werden alle Indizes aktualisiert.

Komprimiere Adresstabelle

(nur aktiv, wenn die vorige Option markiert ist): Diese Option baut beim Komprimieren die Adresstabelle für die Datensätze komplett neu auf. Das optimiert die Größe der Adresstabelle. Diese Option ist sinnvoll, wenn umfangreiches Datenvolumen in der Datenbank erstellt und dann wieder gelöscht wurde. In anderen Fällen ist die Optimierung nicht signifikant.

Beachten Sie, dass diese Option die Komprimierung beträchtlich verlangsamt und alle Mengen, die über den Befehl **SAVE SET** gesichert wurden, ungültig werden. Wir empfehlen außerdem dringend, in diesem Fall gesicherte Mengen zu löschen, da ihre Verwendung zur Auswahl nicht-korrekturer Daten führen kann.

Hinweise:

- Beim Komprimieren werden auch die Datensätze von Tabellen berücksichtigt, die im Papierkorb liegen. Gibt es dort eine große Anzahl Datensätze, kann das ein weiterer Faktor sein, der die Operation langsamer macht.
- Durch diese Option wird die Adresstabelle und folglich auch die Anwendung inkompatibel zum aktuellen Logbuch - sofern vorhanden. Es wird automatisch gesichert und beim nächsten Start der Anwendung muss ein neues Logbuch angelegt werden.
- Um zu sehen, ob die Adresstabelle komprimiert werden muss, gehen Sie auf die **Seite Information** des MSC und vergleichen die Gesamtanzahl der Datensätze mit der Größe der Adresstabelle.

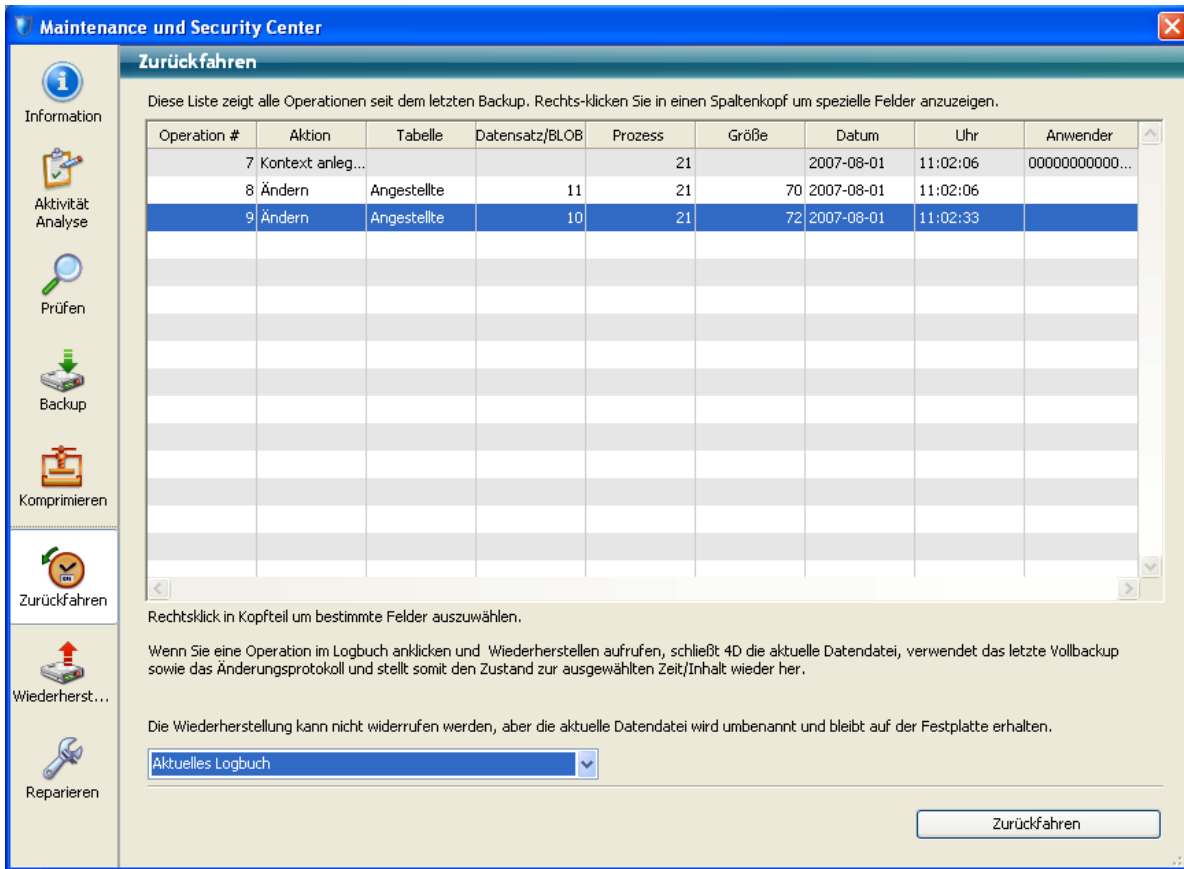
Strukturdatei komprimieren

Der untere Bereich zeigt den aktuellen Pfadnamen der Strukturdatei und die Schaltfläche [...], um vor dem Komprimieren einen anderen Ort zum Speichern der Originaldateien anzugeben.

Klicken Sie jeweils auf die Schaltfläche **Komprimieren**, startet sofort der Komprimierungsvorgang. 4D erstellt einen Bericht im Datenbankordner. Klicken Sie auf die Schaltfläche **Logbuch öffnen**, zeigt 4D das aktuellste Logbuch im standardmäßigen Browser des Rechners (siehe **Logbuch öffnen**).

Auf dieser Seite können Sie auf die Funktion **Rollback** für die im Logbuch ausgeführten Operationen zugreifen. Sie arbeitet ähnlich wie die Operation **Rückgängig** auf mehreren Ebenen. Das ist besonders hilfreich, wenn ein Datensatz versehentlich in der Datenbank gelöscht wurde.

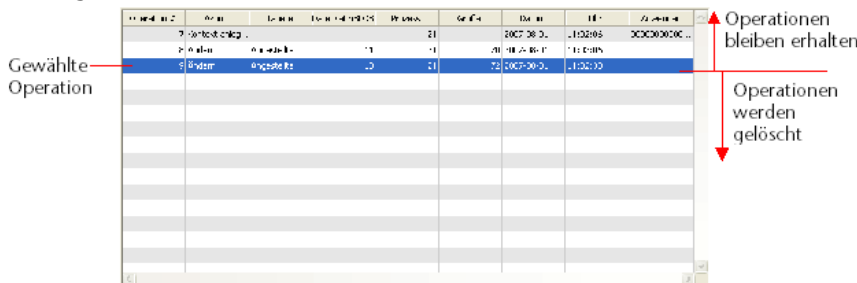
Diese Funktion ist nur verfügbar, wenn die Datenbank mit einem Logbuch arbeitet.



Hinweis: Nur Administrator und Designer der Datenbank sind berechtigt, das Logbuch zu überprüfen.

Die Liste ist genauso aufgebaut wie auf der Seite **Aktivitätsanalyse**. Weitere Informationen dazu finden Sie auf der **Seite Aktivitätsanalyse**.

Um ein Rollback in den Operationen auszuführen, wählen Sie die Zeile, nach der alle Operationen annulliert werden sollen. Die Operation der gewählten Zeile bleibt als letzte erhalten. Wollen Sie z.B. einen Löschvorgang aufheben, wählen Sie die direkt davor liegende Operation. Diese bleibt dann erhalten, alle nachfolgenden Operationen werden zurückgefahren.



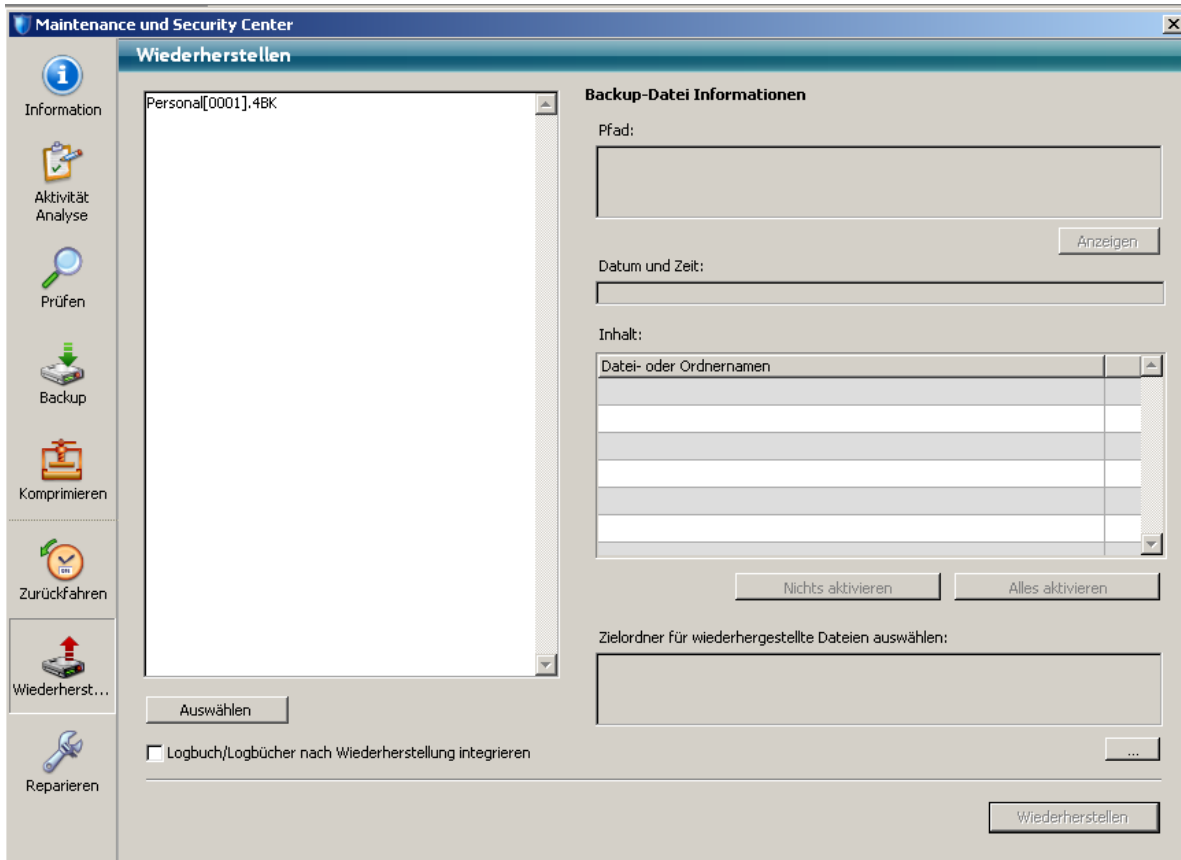
Klicken Sie auf die Schaltfläche **Zurückfahren**. Auf dem Bildschirm erscheint ein Dialogfenster zum Bestätigen.

Klicken Sie auf die Schaltfläche **OK**, werden die Daten in dem Stadium wiederhergestellt, das zum Zeitpunkt der gewählten Operation galt. Über das PopUp-Menü am unteren Rand wählen Sie das passende Logbuch zum Ausführen der Rollback-Funktion in einer wiederhergestellten Datenbank.

Rollback funktioniert folgendermaßen: Klickt der Benutzer auf die Schaltfläche **Zurückfahren**, schließt 4D die aktuelle Datenbank und stellt das letzte Backup der Daten der Datenbank wieder her. Die wiederhergestellte Datenbank wird dann geöffnet. 4D integriert die Operationen des Logbuchs bis zur gewählten Operation. Wurde die Datenbank noch nicht gesichert, startet 4D mit einer leeren Datendatei

Backup manuell wiederherstellen (MSC)

Sie können ein Archiv der aktuellen Datenbank manuell über das Maintenance- und Security-Center (MSC) wiederherstellen. Hier können Sie auch verschiedene Optionen zur Durchführung festlegen:



Der linke Bereich enthält die vorhandenen Backups der Datenbank. Sie können auch unter dem linken Bereich auf die Schaltfläche **Auswählen** klicken, um ein Archiv an anderer Stelle zu öffnen. Es wird dann zur Liste der Archive hinzugefügt.

Wählen Sie hier ein Backup aus, erscheinen im rechten Teil des Fensters Informationen über dieses Backup:

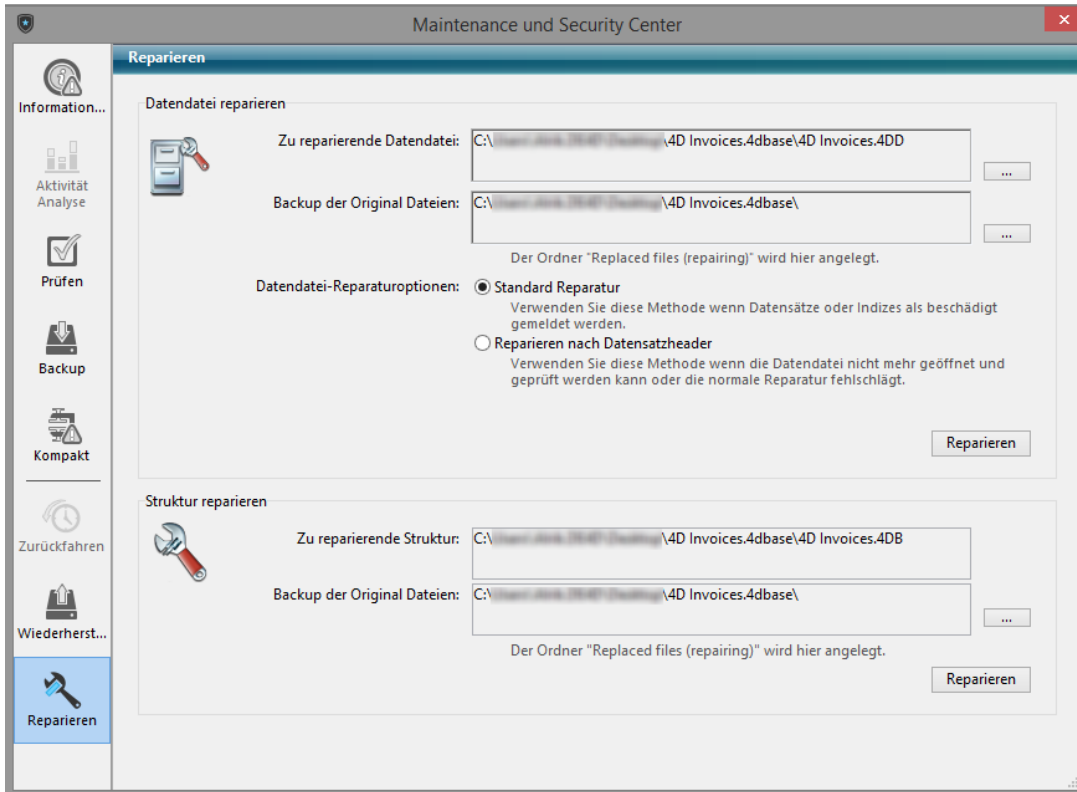
- **Pfad:** Vollständiger Pfadname der gewählten Backup-Datei. Mit der Schaltfläche **Anzeigen** öffnen Sie die Backup-Datei in einem Systemfenster.
- **Datum und Zeit:** Datum und Uhrzeit des Backup.
- **Inhalt:** Inhalt der Backup-Datei. Neben jedem Eintrag in der Liste gibt es ein Ankreuzfeld, über das Sie angeben, ob er wiederhergestellt werden soll. Über die Schaltflächen **Nichts aktivieren** oder **Alles aktivieren** stellen Sie die Liste der Einträge zusammen, die Sie wiederherstellen wollen.
- **Zielordner für wiederhergestellte Dateien auswählen:** Ordner, in dem die wiederhergestellten Dateien abgelegt werden. 4D legt die wiederhergestellten Dateien standardmäßig in einem Ordner mit Namen Archivname (ohne Endung) ab, der auf derselben Ebene wie das Archiv liegt. Wollen Sie das Ziel ändern, klicken Sie unter dem Bereich auf die Schaltfläche [...] und geben einen anderen Ordner an.

Mehrere Logbücher nacheinander integrieren

Sie können mehrere Logbücher hintereinander in die Datenbank integrieren. Haben Sie z.B. 4 Logbücher passend zu 4 Backups der Datenbank, können Sie das erste Backup wiederherstellen und dann die Logbücher der Reihe nach integrieren. Auf diese Weise können Sie z.B. eine Datei wiederherstellen, auch wenn die letzten Backup-Dateien fehlen. Ist diese Option markiert, zeigt 4D nach dem Wiederherstellen den Standard Öffnen-Dialog. Hier können Sie das entsprechende Logbuch auswählen. Der Öffnen-Dialog erscheint erneut nach jeder Integration, bis er abgebrochen wird.

Auf dieser Seite wählen Sie Optionen zum Reparieren der Daten- bzw. Strukturdatei bei Beschädigung. Diese Seite verwenden Sie in der Regel nur auf Anforderung von 4D, wenn beim Öffnen der Datenbank Unregelmäßigkeiten auftreten oder nach einer Überprüfung (siehe [Seite Prüfen](#)).

Warnung: Die Originaldatei wird bei jedem Komprimierungsvorgang dupliziert, was den Anwendungsordner vergrößert. Sie sollten darauf achten, besonders auf OS X, wo 4D Anwendungsordner als Paket erscheinen, dass die Größe der Anwendungsdatei nicht extrem ansteigt. In diesem Fall ist es hilfreich, die Kopien der Originaldatei im Paket per Hand zu entfernen, damit die Größe des Pakets im Rahmen bleibt.



Hinweis: Reparieren ist nur im Wartungsmodus möglich. Wenn Sie versuchen, diese Operation im Standardmodus auszuführen, erscheint eine Meldung, dass die Datenbank geschlossen und im Wartungsmodus neu gestartet wird.

Datendatei reparieren

Der obere Bereich enthält den Pfadnamen der aktuellen Datendatei und die Schaltfläche [...] zum Auswählen einer anderen Datendatei. Klicken Sie auf diese Schaltfläche, erscheint ein Standard-Öffnen Dialog, um die gewünschte Datendatei zum Reparieren auszuwählen. Sie muss zur geöffneten Strukturdatei passen. Bestätigen Sie das Dialogfenster, erscheint der Pfadname der zu reparierenden Datei im Fenster **Komprimieren**.

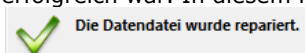
Standardmäßig wird die Originaldatei vor dem Reparieren dupliziert und im Unterordner Ersetzte Dateien (Reparatur) abgelegt. Über die zweite Schaltfläche [...] können Sie vor dem Reparieren einen anderen Ort zum Speichern der Originaldateien angeben. Diese Möglichkeit ist insbesondere beim Reparieren umfangreicher Dateien auf verschiedenen Festplatten hilfreich.

Es gibt zwei Optionen zum Reparieren: **Standard Reparatur** und **Reparieren nach Datensatzheader**. Beide Optionen werden im folgenden beschrieben.

Haben Sie die Einträge zum Wiederherstellen festgelegt, klicken Sie auf die Schaltfläche **Reparieren** oder **Scannen und reparieren** (je nach der markierten Option), um die Reparatur zu starten. Haben Sie die zweite Option gewählt, erscheint ein Zwischendialog, um die Einträge zum Wiederherstellen auszuwählen (siehe unterer Abschnitt [Reparieren nach Datensatzheader](#)).

4D erstellt eine neue leere Datendatei an der Stelle der Originaldatei. Die Originaldatei wird in den Ordner mit Namen "\Ersetzte Dateien (Reparatur) Datum Zeit" geschoben. Seine Position wird im Bereich "Backup der Original Dateien" definiert. Standardmäßig ist der Ordner der Datenbank angegeben. Die leere Datei wird mit den wiederhergestellten Daten gefüllt.

Ist die Reparatur abgeschlossen, erscheint die Seite "Reparatur" des MSC. Eine Meldung gibt an, ob die Reparatur erfolgreich war. In diesem Fall können Sie die Datenbank sofort öffnen.



Die Schaltfläche **Öffne Logbuch** zeigt in Ihrem Browser eine Web Seite mit den Ergebnissen der durchgeführten Operation. Sie zeigt alle ausgeführten Überprüfungen oder Reparaturen an, sowie aufgetretene Fehler ([OK] erscheint,

wenn die Überprüfung korrekt war). Die Datei wird im Ordner **Logs** der Datenbank generiert. Sie wird im XML und HTML Format angelegt und hat den Namen "DatenbankName_Repair_log".

- `<DatabaseName>` ist der Name der Strukturdatei ohne Endung, z.B. "Rechnungen"
- `<yyyy-mm-dd hh-mm-ss>` ist der Zeitstempel der Datei. Er basiert auf der lokalen Systemzeit, wenn die Wartungsoperation gestartet ist, zum Beispiel "2016-01-22 15-20-45".

In dieser Datei können Sie alle ausgeführten Operationen sehen. Klicken Sie auf die Schaltfläche **Öffne Logbuch**, zeigt 4D das aktuellste Logbuch im Standardbrowser des Rechners. Weitere Informationen dazu finden Sie im Abschnitt **Logbuch öffnen**.

Standard Reparatur

Wählen Sie diese Option, wenn nur wenige Datensätze oder Indizes beschädigt sind (Adresstabellen sind intakt). Die Daten werden komprimiert und repariert. Diese Art der Reparatur lässt sich nur ausführen, wenn Datendatei und Strukturdatei zueinander passen.

Reparieren nach Datensatzheader

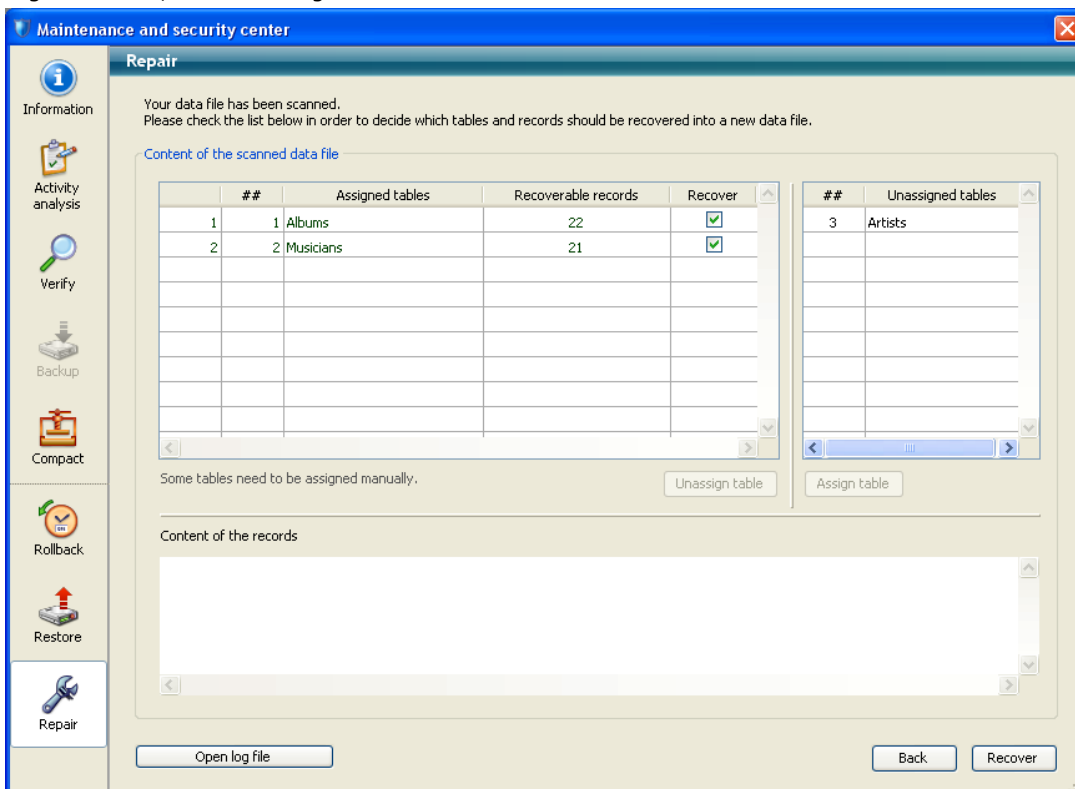
Verwenden Sie diese Reparatur auf niedriger Ebene nur, wenn die Datendatei ernsthaft beschädigt ist und alle anderen Lösungen, wie Wiederherstellen über ein Backup, Standard Reparatur wirkungslos geblieben sind.

4D Datensätze sind unterschiedlich groß. Deshalb muss die Stelle, wo sie auf der Festplatte in einer spezifischen Tabelle, genannt *Adresstabelle*, gespeichert sind, beibehalten werden, um sie wieder zu finden. Das Programm greift deshalb auf die Adresse des Datensatzes über einen Index und eine Adresstabelle zu. Sind nur Datensätze oder Indizes beschädigt, reicht die Standardreparatur in der Regel aus, um das Problem zu lösen. Ist dagegen die Adresstabelle selbst betroffen, ist ein komplexeres Wiederherstellen erforderlich, da diese Tabelle wiederhergestellt werden muss. Dazu verwendet das MSC die Marker, die im Kopfteil jedes Datensatzes angelegt sind. Sie sind vergleichbar mit einem Inhaltsverzeichnis des Datensatzes, inkl. aller wichtigen Informationen, über die sich die Adresstabelle rekonstruieren lässt.

Hinweise:

- Haben Sie in der Datenbankstruktur in den Tabelleneigenschaften die Option **Datensätze definitiv löschen** deaktiviert, können nach dem Wiederherstellen nach Datensatzheader zuvor gelöschte Datensätze wieder erscheinen.
- Wiederherstellen nach Kopfteil berücksichtigt keine Einschränkungen zur Datenintegrität. So erhalten Sie nach Ausführen dieser Operation u.U. doppelte Werte in einmaligen Felder oder NULL Werte mit Felder, für die Nie Null definiert wurde.

Klicken Sie auf die Schaltfläche **Scannen und Reparieren**, scannt 4D die Datendatei komplett. Ist die Operation abgeschlossen, erscheint folgendes Fenster:



Hinweis: Ließen sich alle Datensätze und alle Tabellen zuordnen, erscheint nur der Hauptbereich.

Der Bereich "Datensätze in Datendatei gefunden" besteht aus zwei Tabellen mit den Informationen aus dem Scan-Vorgang.

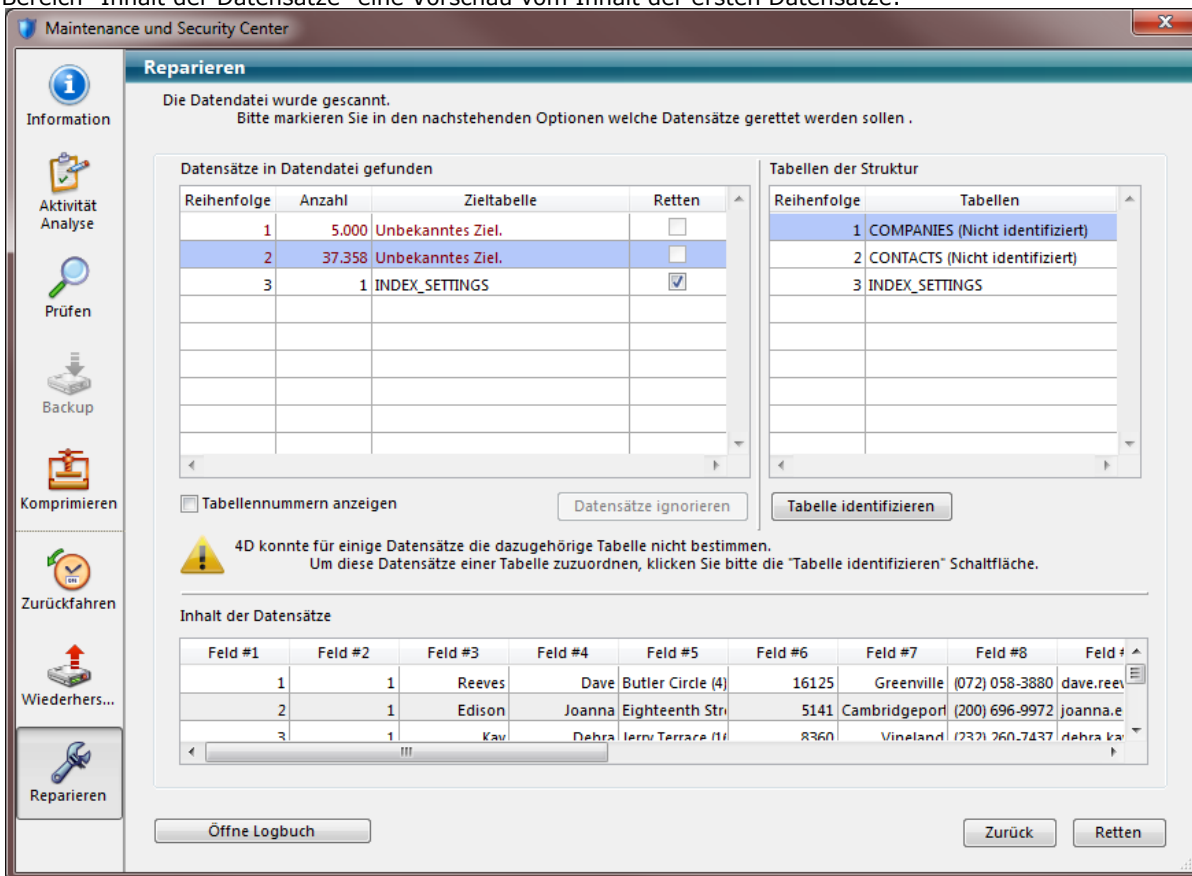
- Die erste Tabelle zeigt die Information aus dem Scannen der Datendatei. Jede Zeile enthält eine Gruppe der wiederherstellbaren Datensätze in der Datendatei:
 - Die erste Spalte gibt die Reihenfolge der Wiederherstellung für die Gruppe an.
 - Die Spalte **Anzahl** zeigt die Anzahl der Tabellen, die der Gruppe zugewiesen wurden - sofern vorhanden.
 - Die Spalte **Zieltabelle** gibt die Namen der Tabellen an, die automatisch den Gruppen der identifizierten Datensätze zugewiesen wurden. Die Tabellennamen erscheinen automatisch in grün.
 - Die Spalte **Wiederherstellbare Datensätze** gibt die Anzahl der wiederherstellbaren Datensätze in der Gruppe an.

- Über die Spalte **Retten** können Sie für jede Gruppe angeben, ob Sie die Datensätze wiederherstellen wollen. Diese Option ist standardmäßig für alle Gruppen markiert.
- Die zweite Tabelle zeigt die nicht zugewiesenen Tabellen an, z.B. Tabellen, die keinen Datensätzen zugeordnet werden konnten.

Manuell zuweisen

Ließen sich einige Datensatzgruppen aufgrund einer beschädigten Adresstabelle nicht zuweisen, können Sie diese manuell zuweisen.

Dazu wählen Sie zuerst im ersten Bereich eine nicht zugewiesene Gruppe aus. Zur leichteren Zuordnung zeigt der untere Bereich "Inhalt der Datensätze" eine Vorschau vom Inhalt der ersten Datensätze:



Als nächstes wählen Sie im rechten Bereich **Tabellen** die passende Tabelle für die Gruppe aus und klicken auf die Schaltfläche **Tabelle identifizieren**. Sie können eine Tabelle auch per Drag-and-Drop zuweisen.

Die Datensatzgruppe wird dann der Tabelle zugewiesen und in dieser Tabelle wiederhergestellt. Die per Hand zugewiesenen Tabellennamen erscheinen in schwarz.

Über die Schaltfläche **Datensätze ignorieren** können Sie die manuell ausgeführte Zuweisung zwischen Tabelle und Datensatzgruppe wieder entfernen.

Strukturdatei reparieren

Der untere Bereich zeigt den aktuellen Pfadnamen der Strukturdatei und die Schaltfläche [...], um vor dem Reparieren einen anderen Ort zum Speichern der Originaldateien anzugeben.





Klicken Sie jeweils auf die Schaltfläche **Reparieren**, wird bei Bedarf die Datenbank geschlossen und im Wartungsmodus neu geöffnet, bevor die Operation startet

Ist der Reparaturvorgang abgeschlossen, erstellt 4D ein Logbuch im Ordner **Logs** der Datenbank. Es wird im XML- und HTML-Format erstellt und hat den Namen "`<DatabaseName>_Repair_Log_<yyyy-mm-dd hh-mm-ss>.xml`":

- `<DatabaseName>` ist der Name der Strukturdatei ohne Endung, zum Beispiel "Rechnungen"
- `<yyyy-mm-dd hh-mm-ss>` ist der Zeitstempel der Datei. Er basiert auf der lokalen Systemzeit, wenn die Wartungsoperation gestartet ist, zum Beispiel "2016-01-22 15-20-45".

Hier können Sie alle ausgeführten Operationen ansehen. Klicken Sie auf die Schaltfläche **Öffne Logbuch**, zeigt 4D das aktuellste Logbuch im standardmäßigen Browser des Rechners (siehe Abschnitt **Logbuch öffnen**).

Kompilieren

-  Überblick
-  Compiler Fenster
-  Einstellungen zum Kompilieren
-  Diagnosehilfen beim Kompilieren

Sie können Ihre Datenbank kompilieren, d.h. alle Methoden in Maschinensprache übersetzen. Durch Kompilieren der Datenbank überprüfen Sie die Konsistenz des Code und beschleunigen seine Ausführung. Außerdem können Sie den Code insgesamt schützen. Kompilieren ist ein unumgänglicher Schritt zwischen der Entwicklung von Datenbanken mit 4D und deren Einsatz als eigenständige Anwendung (stand-alone).

Das Kompilieren ist vollkommen automatisch; es erfordert jedoch mehr Striktheit beim Schreiben von 4D Code. Das Kapitel **Compiler** des Handbuchs *4D Programmiersprache* enthält Empfehlungen und spezifische Informationen zum Programmieren im Hinblick auf Kompilieren. Der Compiler dient auch dazu, jegliche Programmierfehler anzuzeigen und in ihren Kontext zu setzen.

Was ist ein Compiler?

Auf einem Computer werden Befehle nur unter Verwendung von „0“ und „1“ geschrieben. Diese binäre Sprache heißt Maschinensprache. Das Herz der Maschine, der Mikroprozessor, versteht nur diese Sprache. Ein Programm, das mit einer komplexen Computersprache geschrieben wird, z.B. C, C++, SQL, Java wird zuerst in Maschinensprache übersetzt, so dass es für den Mikroprozessor des Rechners verständlich wird.

Dafür gibt es zwei Möglichkeiten:

- Die Anweisungen werden während der Ausführung übersetzt; das bedeutet, das Programm wird interpretiert
- Die Anweisungen werden vor Ausführung des Programms als Ganzes übersetzt, das bedeutet, das Programm wird kompiliert

Modus Interpretiert

Wird eine Reihe von Anweisungen über einen Interpreter ausgeführt, passiert folgendes:

- Das Programm liest die Anweisung in der zum Programm gehörenden Sprache
- Es übersetzt die Anweisung in Maschinensprache
- Es führt die Anweisung aus

Dieser Ablauf wird für jede Anweisung im Programm durchgeführt. Das Programm, welches diesen Ablauf steuert, heißt Interpreter. Für eine Datenbank in der Entwicklungsphase werden 4D Methoden interpretiert.

Modus kompiliert

Ein kompiliertes Programm wird vor der Ausführung als Ganzes übersetzt. Das Ergebnis ist eine neue Datei mit einer Reihe von Anweisungen in Maschinensprache. Dieser Satz wird für wiederholten Gebrauch gesichert – Die Übersetzung wird nur einmal durchgeführt, die kompilierte Version des Programms steht für eine erneute Ausführung zur Verfügung. Diese Phase ist vollkommen unabhängig von jeglicher Verwendung des Programms. Das Programm, das die Übersetzung durchführt, heißt *Compiler*.

Kompilierte Objekte

Der Compiler in 4D kompiliert die Datenbankmethoden, Projektmethoden, Trigger, Formularmethoden und Objektmethoden in Ihrer Datenbank. Haben Sie keines dieser Elemente in einer Anwendung, hat der Compiler nichts zum Kompilieren.

Bei erfolgreich abgeschlossener Kompilierung ist die Verwendung der kompilierten Datenbank identisch zur Originaldatenbank.

Warum kompilieren?

Der größte Vorteil von Kompilieren ist natürlich die beschleunigte Ausführung. Weitere Vorteile sind:

- Systematische Überprüfung des Code
- Anwendung und Komponenten sind vor Veränderungen geschützt

Beschleunigte Ausführung

Die gesteigerte Geschwindigkeit beruht auf zwei Merkmalen des kompilierten Code: Direkte Übersetzung des Code, auf einmal und komplett, sowie direkter Zugriff auf die Adressen von Variablen und Methoden.

- **Direkte und endgültige Übersetzung des Code**

Der Code der in 4D geschriebenen Methoden wird über den Compiler auf einmal und für alle übersetzt. Die Zeit, die im interpretierten Modus zum Übersetzen aller Anweisungen erforderlich ist, wird eingespart, wenn Sie mit einer kompilierten Datenbank arbeiten. Nehmen wir z.B. eine Schleife mit einer Folge von Anweisungen, die 50 mal wiederholt wird:

```
For($i;1;50)
  `Folge von Anweisungen
End for
```

In einer interpretierten Datenbank wird jede Anweisung in der Folge 50 mal übersetzt. Der Compiler eliminiert die Übersetzungsphase für jede Anweisung. Sie sparen also für jede Anweisung in der Folge 50 Übersetzungen.

- **Direkter Zugriff auf die Variablen- und Methodenadressen**

In interpretierten Datenbanken wird auf Variablen über ihren Namen zugegriffen. Deshalb muss 4D auf den Namen zugreifen, um den Wert der Variablen zu erhalten. Im kompilierten Modus weist der Compiler jeder Variablen eine Adresse zu, schreibt diese Adresse direkt in den Code und geht – wenn erforderlich – direkt zu dieser Adresse.

Hinweise:

- Das betrifft nicht Operationen, die auf die Festplatte zugreifen müssen, da ihre Ausführungsgeschwindigkeit durch die Übertragungsrate zwischen Rechner und Peripherie (Treiber oder Festplatte) bestimmt wird.
- Kommentare werden nicht übersetzt, sie erscheinen also nicht im kompilierten Code. Folglich beeinträchtigen Kommentare nicht die Ausführungszeit im kompilierten Modus.

Code prüfen

Der Compiler dient auch als Syntax-Prüfer für Ihre Datenbanken. Er überprüft automatisch Ihren Code und bemerkt eventuelle Zweideutigkeiten. 4D erkennt das dagegen nur beim Ausführen der Methode.

Nehmen wir an, eine Ihrer Methoden enthält eine Reihe von Tests und eine Folge von auszuführenden Anweisungen. Bei einer hohen Anzahl von Tests ist eher unwahrscheinlich, dass Sie alle Fälle vollständig durchtesten. Es kann also passieren, dass ein Syntaxfehler in einem nicht getesteten Fall erst auftritt, wenn der Endbenutzer darauf stößt.

Probleme dieser Art werden vermieden, wenn Sie eine kompilierte Datenbank einsetzen. Beim Kompilieren einer Datenbank durchleuchtet der Compiler die gesamte Datenbank und analysiert jede Anweisung. Er findet jede Abweichung von der Normalität und erzeugt eine Fehlermeldung bzw. Warnung.

Anwendung und Komponenten schützen

st Ihre Datenbank kompiliert, können Sie den interpretierten Code über den „Applikation Builder“ löschen. In diesem Fall ist der Zugriff auf die Designumgebung mit Ausnahme der Datensätze blockiert. Weitere Informationen zum Erstellen der Anwendung finden Sie im Kapitel **Eigenständige Anwendung erstellen und weitergeben**.

In einer kompilierten Datenbank sind die Befehle zum Entwickeln deaktiviert.

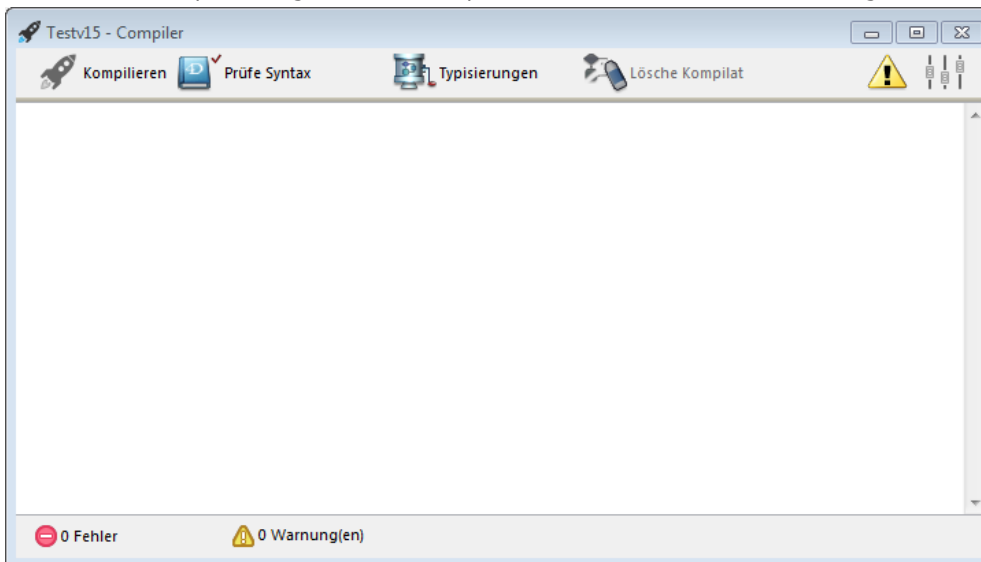
Ist in der Host Datenbank eine kompilierte Komponente installiert, sind die gemeinsam genutzten Projektmethoden im Explorer verfügbar und lassen sich in Methoden der Host Datenbank aufrufen. Ihr Inhalt erscheint jedoch weder in der Vorschau noch im Debugger. Andere Projektmethoden der Komponente erscheinen nie. Weitere Informationen zu Komponenten finden Sie im Kapitel **4D Komponenten entwickeln und installieren**.

Die Vorteile sind:

- Die Struktur der Datenbank lässt sich nicht verändern, weder absichtlich noch aus Versehen
- Ihre Methoden sind geschützt

Kompilieren in 4D

In 4D ist ein Compiler integriert. Das Kompilieren der Datenbank wird über folgendes Dialogfenster ausgeführt:



Sie können das Kompilieren auch direkt starten. Wählen Sie dazu im Menü **Design** oder in der Werkzeugleiste im DropDown-Menü zum Icon **Compiler** den Befehl **Kompilierung starten** bzw. **Kompilieren**.

Die allgemeinen Optionen zum Kompilieren werden in den Datenbank-Eigenschaften auf der **Seite Compiler** definiert.

Auch bei kompilierter Datenbank können Sie jederzeit zwischen interpretiertem und kompiliertem Modus wechseln. Sie müssen das Programm nicht verlassen. Dazu rufen Sie in 4D im Menü **Start>Neustart kompiliert bzw. interpretiert** auf – außer der interpretierte Code wurde entfernt (siehe voriger Abschnitt).

Sie können auch beim Starten der Datenbank im Öffnen-Dialog zwischen kompiliertem und interpretiertem Modus wählen. Weitere Informationen dazu finden Sie im Abschnitt **Optionen des Öffnen-Dialogs**.

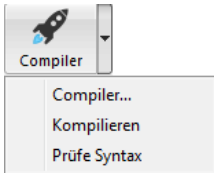
Ändern Sie die Struktur Ihrer Datenbank im interpretierten Modus, müssen Sie erneut kompilieren, damit die Änderungen berücksichtigt werden.

Wechseln Sie den Modus, schließt 4D den aktuellen Modus und öffnet den neuen, d.h. die Anwendung wird geschlossen und wieder geöffnet. Folglich werden - sofern angegeben - nacheinander die beiden Datenbankmethoden aufgerufen:

Semaphore -> Datenbankmethode On Startup

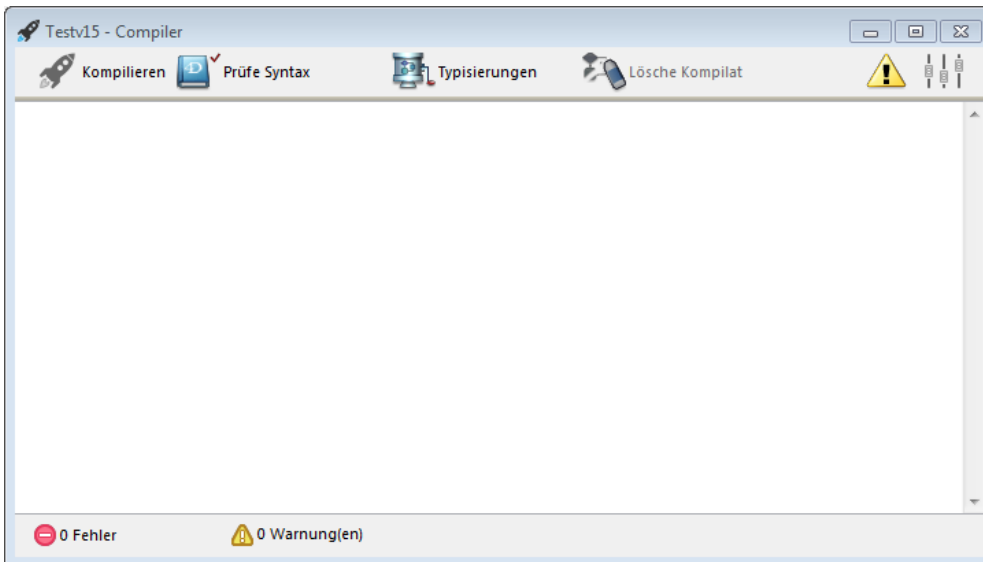
Compiler Fenster

Um das Dialogfenster Compiler aufzurufen, wählen Sie im Menü **Design** oder in der Werkzeugleiste im DropDown-Menü zum Icon **Compiler** den Befehl **Kompilierung starten bzw. Kompilieren**:



Hinweis: Diese Befehle werden erst aktiv, wenn die Datenbank mindestens eine Methode enthält.

In diesem Fenster starten Sie das Kompilieren der Datenbank oder prüfen die Syntax der Methoden. Über weitere Schaltflächen können Sie typisierte Methoden in der Datenbank erzeugen/wieder erzeugen, den kompilierten Code entfernen, Warnungen ein- oder ausblenden und direkt auf die Datenbank-Eigenschaften zugreifen.



Hinweis: Zum Kompilieren der Datenbank benötigen Sie eine Lizenz der 4D Developer Edition. Ohne diese Lizenz können Sie nicht kompilieren, die Schaltfläche **Kompilieren** ist inaktiv. Sie können aber trotzdem die Syntax überprüfen und typisierte Methoden erzeugen.

Kompilieren

Über die Schaltfläche **Compiler** startet der Kompilierungsprozess der Datenbank. Er startet auch über das Menü **Design>Kompilierung starten**. Wurde die Datenbank bereits kompiliert, ersetzt der neue kompilierte Code den bisherigen.

Zu Beginn gibt es, je nach den gewählten Einstellungen auf der **Seite Compiler** der Datenbank-Eigenschaften verschiedene Durchläufe zum Prüfen, Typisieren und Initialisieren.

Wurden keine Fehler gefunden, beginnt die eigentliche Kompilierung. Treten Fehler auf, stoppt der Prozess. Im Fenster erscheint eine hierarchische Liste mit den betroffenen Methodennamen und Zeilennummern:

Doppelklicken Sie auf jeden gefundenen Fehler, um die entsprechende Methode direkt im 4D **Methodeneditor** zu öffnen; die fehlerhafte Zeile wird hervorgehoben und der Fehlertyp erscheint im Syntaxbereich des Fensters. Mit den Befehlen **Vorheriger Fehler/Nächster Fehler** im Menü **Methode** des Editors können Sie zwischen den Zeilen navigieren.

Methode	Fenster	Hilfe
Gehe zu Definition...		Ctrl+K
Suche Referenzen...		Ctrl+Alt+K
Suche Aufrufe...		
Gehe zu Zeile...		Ctrl+L
Kommentar An/Aus		Ctrl+/
Ausdruck tauschen		Ctrl+Alt+T
Block auswählen		Ctrl+B
Beginn des Blocks		Ctrl+Up Arrow
Ende des Blocks		Ctrl+Down Arrow
Zeilen nach oben verschieben		Alt+Up Arrow
Zeilen nach unten verschieben		Alt+Down Arrow
Nächster Fehler		Ctrl+=
Vorheriger Fehler		Ctrl+Alt+=
Anzeigen		

Hinweis: Die Anzahl der Fehler kann beim ersten Kompilieren spektakulär sein. Das sollte Sie nicht verunsichern. Bei genauerer Betrachtung sehen Sie, dass viele Fehler auftreten, weil die Konventionen der Datenbank nicht berücksichtigt wurden. Der Compiler liefert immer eine genaue Diagnose, anhand der Sie die Fehler korrigieren können.

Prüfe Syntax

Die Schaltfläche **Prüfe Syntax** startet die Syntaxprüfung. Die gefundenen Fehler erscheinen im darüberliegenden Informationsbereich. Sie können auf eine Zeile doppelklicken, um die Methode im Methodeneditor zu öffnen.

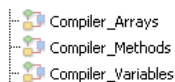
Haben Sie keine Lizenz zum Kompilieren von Anwendungen, ist nur diese Option verfügbar.

Sie können die Syntaxprüfung auch direkt über das Icon **Compiler** in der Werkzeuggestreife starten.

Typisierungen

Die Schaltfläche **Typisierungen** erstellt oder aktualisiert die typisierenden "Compilermethoden". Das sind Projektmethoden, die alle Variablen (Prozess-, Interprozess-Variablen und Interprozess-Arrays) zum Typisieren zusammenfassen. Diese Methoden verwendet der Compiler direkt während dem Kompilieren von Code. Sind die Methoden bereits vorhanden, wird der Inhalt aktualisiert.

Diese Methoden werden von 4D angelegt, ihr Name muss mit „Compiler_“ beginnen. Sie können in den Datenbank-Eigenschaften standardmäßig bis zu 5 Compilermethoden generieren. Weitere Informationen dazu finden Sie im Abschnitt **Compilermethoden für ...**. Die von 4D generierten und verwalteten Compilermethoden haben automatisch das Attribut „unsichtbar“. Sie sind im Explorer mit einem spezifischen Icon gekennzeichnet:



Es werden nur die notwendigen Compilermethoden erstellt, d.h. solche, für die bereits Einträge in der Datenbank vorhanden sind.

Der Informationsbereich zeigt alle Fehler an, die während dem Erstellen oder Aktualisieren einer Methode auftreten. Doppelklick auf eine Zeile öffnet die dazugehörige Methode im Methodeneditor.

Weitere Informationen zum Typisieren von Methoden finden Sie im Handbuch *4D Programmiersprache* im Kapitel **Compiler**.

Lösche Kompilat

Die Schaltfläche **Lösche Kompilat** löscht den kompilierten Code der Strukturdatei. Klicken Sie auf das Icon, wird der gesamte Code gelöscht, der während dem Kompilieren erzeugt wurde. Die Größe der Strukturdatei wird entsprechend verkleinert, wenn Sie eine Komprimierung durchführen (siehe **GET STYLE SHEET INFO**).

In diesem Fall ist der Befehl **Neustart kompiliert** im Menü **Start** inaktiv. Die Option **Kompilierte Datenbank** im Öffnen-Dialog der Datenbank erscheint in Grauschrift.

Beachten Sie, dass evtl. erzeugte Compiler-Methoden durch diese Option nicht gelöscht werden.

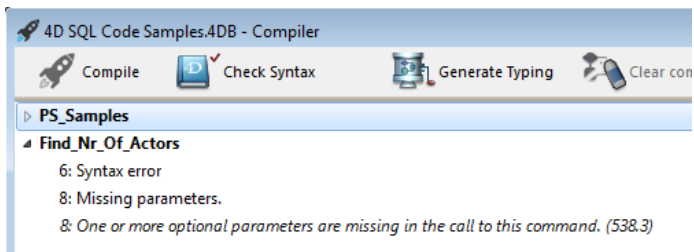
Warnungen anzeigen

Warnungen sind spezifische Meldungen, die der Compiler beim Prüfen der Syntax erstellt. Solche Meldungen sollen Ihre Aufmerksamkeit auf Anweisungen lenken, die bei der Ausführung zu Fehlern führen können. Sie unterbinden nicht die Kompilierung. Weitere Informationen dazu finden Sie im Abschnitt **Warnungen**.

Diese Warnungen sind je nach Gegebenheiten und Programmierstil mehr oder weniger wichtig. Das Optionsfeld **Warnungen anzeigen** ist standardmäßig aktiviert. Sie können die Warnungen im Informationsbereich ein- oder ausblenden.



Die Warnungen erscheinen im Dialogfenster nach den anderen Fehlertypen in Kursivschrift, gefolgt von der Nummer der Warnung:



Doppelklicken Sie auf eine Warnung, erscheint die dazugehörige Methode.

Warnungen während dem Kompilieren deaktivieren

Sie können selektiv bestimmte Warnungen während dem Kompilieren deaktivieren. Dazu fügen Sie folgenden Kommentar in den Code einer 4D Methode ein:

```
//%W-<warning number>
```


Warnnummern erscheinen am Ende jeder Meldung in der Liste der Kompilierungsfehler. Wollen Sie z.B. folgende Warnung deaktivieren:

1: Zeiger in einer Array Deklaration (518.5)

... setzen Sie folgenden Kommentar in die 4D Methode, vorzugsweise in eine Methode COMPILER_xxx (zuerst kompilierte Methode):

```
//%W-518.5
```

Direkter Zugang zu den Einstellungen des Compiler

Über die Schaltfläche  können Sie direkt die **Seite Compiler** der Datenbank-Eigenschaften aufrufen. Die hier definierten Optionen gelten nur für die geöffnete Datenbank.

Optionen zum Kompilieren

Dieser Bereich enthält allgemeine Optionen, die beim Kompilieren verwendet werden.

- **Bereichsüberprüfung:** Mit dieser Option erfolgt eine zusätzliche Überprüfung, die den Code zur Laufzeit und abhängig vom Status der Datenbankobjekte in einem bestimmten Moment überprüft. Weitere Informationen dazu finden Sie im Abschnitt **Bereichsprüfung**.
- **Erzeuge Symboldokument:** Diese Option generiert eine Textdatei vom Typ ASCII (text only), welche die Variablen mit Typ und Methode, aus der sie stammen, auflistet. Die Liste enthält auch die Methoden und Funktionen mit den dazugehörigen Parametertypen und Ergebnissen (sofern vorhanden). Die Datei liegt im gleichen Ordner wie die Strukturdatei der Datenbank. Sie lautet: *DatabaseName_symbols.txt*. Weitere Informationen dazu finden Sie im Abschnitt **Symboldatei**.
- **Erzeuge Fehlerdokument:** Diese Option erzeugt während der Syntaxprüfung ein Fehlerdokument. Es zeigt allgemeine Fehler, in spezifischen Zeilen auftretende Fehler sowie Warnungen. Generell sind alle Fehler, die der Compiler anzeigt, automatisch im Menü **Methode** verfügbar. Ein Fehlerdokument ist jedoch hilfreich, denn es lässt sich leicht auf andere Rechner übertragen, insbesondere wenn mehrere Entwickler in einer Client/Server-Umgebung zusammenarbeiten. Das Fehlerdokument wird im XML Format erstellt, um das automatische Analysieren (Parsen) seines Inhalts zu erleichtern. Damit können Sie auch eigene Formate zum Anzeigen der Fehler erstellen. Das Fehlerdokument erhält den Namen *DatabaseName_errors.xml* und liegt neben der Strukturdatei der Datenbank. Weitere Informationen dazu finden Sie im Abschnitt **Fehlerdokument**.
- **Mehrfach Kompilierung (32-Bit und 64-Bit):** Ist diese Option nicht markiert (Standardeinstellung), wird nur der kompilierte Code der aktuellen Architektur (32-Bit oder 64-Bit) erzeugt. Ist diese Option markiert, enthalten die Dateien *.4DC* und *.4DB* den kompilierten Code für 32-Bit und für 64-Bit. Sie können dann entweder mit einer 4D Applikation in 32-Bit oder in 64-Bit laufen. Beachten Sie, dass:
 - die 4D Server Version in 64-Bit für Windows und für Mac OS verfügbar ist
 - ab 4D v15 R2 die 4D Version in 64-Bit als Vorabversion verfügbar ist.
 - Nur 4D Datenbanken, die in Unicode Modus konvertiert wurden, sich für 64-Bit kompilieren lassen.
- **Lokale Variablen initialisieren:** Diese Option definiert den Initialisierungsmodus der lokalen Variable zu Beginn der Methoden:
 - **Auf 'Null':** Variablen werden standardmäßig auf Null gesetzt (leerer String für Zeichenketten, 0 für Ziffern...).
 - **Auf zufälligen Wert:** Der Compiler weist Variablen je nach Typ einen bestimmten Wert zu (1919382119 für Lange Ganzzahl, "rgrg" für Zeichenketten, True für Boolean, ...). Mit dieser Option können Sie lokale Variablen abstecken, die Sie vergessen haben, zu initialisieren.
 - **Nicht:** Der Compiler initialisiert die Variablen nicht. So gewinnen Sie Zeit beim Ausführen der Datenbank. Das setzt allerdings eine korrekte Initialisierung voraus.
- **Kompilationspfad:** Diese Option definiert die Anzahl der Durchgänge, die der Compiler ausführt und somit die Dauer der Kompilierung.
 - **Variablen typisieren:** Durchläuft alle Stadien der Typisierung, die für die Kompilierung erforderlich sind.
 - **Prozess- und Interprozessvariablen sind typisiert:** Der Durchlauf zum Typisieren von Prozess- und Interprozessvariablen wird nicht ausgeführt. Verwenden Sie diese Option, wenn Sie alle Prozess- und Interprozessvariablen bereits selbst oder über die Funktion „Compilermethoden automatisch generieren“ typisiert haben.
 - **Alle Variablen sind typisiert:** Der Durchlauf zum Typisieren von lokalen Variablen, Prozess- und Interprozessvariablen wird nicht ausgeführt. Verwenden Sie diese Option, wenn Sie sicher sind, dass alle Prozess-, Interprozess- und lokalen Variablen eindeutig typisiert wurden.

Compilermethoden für ...

In diesem Bereich können Sie Compilermethoden umbenennen, die der Compiler automatisch generiert. Diese Methoden fassen alle Deklarationen zum Typisieren von Variablen, Prozess- und Interprozess-Arrays sowie von lokalen Variablen zusammen. Sie werden im Kompilierungsfenster generiert. Weitere Informationen dazu finden Sie im Abschnitt **Compiler Fenster**.

Es lassen sich bis zu 5 Compilermethoden generieren; die Methode wird nur angelegt, wenn die Datenbank die entsprechenden Einträge enthält:

- **Variablen:** Gruppiert Compiler-Deklarationen von Prozessvariablen
- **Interprozess-Variablen:** Gruppiert Compiler-Deklarationen von Interprozessvariablen
- **Arrays:** Gruppiert Compiler-Deklarationen von Prozess Arrays
- **Interprozess-Arrays:** Gruppiert Compiler-Deklarationen von Interprozess Arrays
- **Methoden:** Gruppiert Compiler-Deklarationen von lokalen Variablen, die Parameter angeben, z.B. `C_LONGINT(MeineMethode;$1)`.

Sie können die Methoden im jeweiligen Eingabebereich umbenennen.

Sie beginnen jedoch immer mit dem fest vorgegebenen Eintrag "Compiler_". Der Methodename inkl. Vorsilbe muss einmalig sein und kann max. 31 Zeichen enthalten. Zeichen aus dem erweiterten ASCII Code (Akzente, Symbole) sowie Leerzeichen sind nicht zulässig.

Standard Typisierung

In diesem Bereich setzen Sie den Standardtyp für zweideutige Datenbankobjekte.

- **Numerisch:** Diese Option sorgt für eine eindeutige Zahlentypisierung, entweder als **Zahl** oder als **Lange Ganzzahl**. Sie hat keinen Vorrang gegenüber Direktiven, die u.U. in Ihrer Datenbank festgelegt wurden. Sie können das Laufen Ihrer Datenbank optimieren, wenn Sie den Typ Lange Ganzzahl wählen.
- **Schaltflächen:** Diese Option sorgt für eine eindeutige Typisierung von Schaltflächen, entweder als **Zahl** oder als **Lange Ganzzahl**. Sie hat keinen Vorrang gegenüber Direktiven, die u.U. in Ihrer Datenbank festgelegt wurden. Sie gilt für Standard Schaltflächen und für folgende Objekte: Kontrollkästchen, 3D Kontrollkästchen, hervorgehobene Schaltflächen, Unsichtbare Schaltflächen, 3D Schaltflächen, Bildschaltflächen, Schaltflächengitter, Optionsfelder, 3D Optionsfelder, Bild Optionsfelder, Bild PopUp-Menüs, hierarchische PopUp-Menüs und PopUp/DropDown-Listen.

Diagnosehilfen beim Kompilieren

Es gibt drei Arten, die beim Analysieren und Korrigieren von Datenbanken unterstützen:

- Die aktuelle Analysehilfe ist über die Symboldatei verfügbar. Über diese Tabelle finden Sie rasch durch Ihre Variablen durch. Sie ist ein wertvolles Werkzeug zum Interpretieren der Fehlermeldungen, die der Compiler erstellt.
- Die Korrekturhilfe ist über die Fehlerdatei verfügbar, die Sie als Textdatei verwenden können.
- Die Ausführungshilfe bzw. Bereichsprüfung bietet weitere Hilfsmittel zum Überwachen der Konsistenz und Verlässlichkeit Ihrer Anwendungen.

Hinweis: Eine wichtige Hilfe ist auch die Typisierung von Variablen über die automatischen Compilermethoden. Weitere Informationen dazu finden Sie im Abschnitt **Typisierungen**.

Symboldatei

Die Symboldatei ist eine Textdatei vom Typ ASCII (text only), deren Länge von der Größe Ihrer Datenbank abhängt. Diese Datei wird während dem Kompilieren nicht standardmäßig erzeugt. Dazu müssen Sie in den Datenbank-Eigenschaften auf der Seite Compiler die Option Symboldatei erzeugen aktivieren. Weitere Informationen dazu finden Sie im Abschnitt **Optionen zum Kompilieren**. Die erzeugte Datei liegt im gleichen Ordner wie die Strukturdatei der Datenbank. Sie lautet *DatenbankName_symbols.txt*.

Die Symboldatei könnte folgendermaßen aussehen, wenn sie mit einem Texteditor geöffnet wird:

```
4D_Airports2003      10/9/2003      5:34 PM
<>4DCOMMENTS      Text          (M)  Compiler_Variables_Inter
<>4DQUERY         Text          (M)  Compiler_Variables_Inter
<>4DQUERYCOMMENTS Text          (M)  Compiler_Variables_Inter
<>4DTABLETITLE    Text          (M)  Compiler_Variables_Inter
<>AAIRCRAFTDISTINCTRANGEKM Long integer 1 dimension (M)
Compiler_Arrays_Inter
<>AAIRCRAFTFAMILY Text          1 dimension (M)  PS_InitArrays
<>AAIRCRAFTID     Long integer 1 dimension (M)  PS_InitArrays
<>AAIRCRAFTMANUFACTURER Text      1 dimension (M)  PS_InitArrays
<>AAIRCRAFTNAME   Text          1 dimension (M)  PS_InitArrays
<>AAIRCRAFTRANGEKM Long integer 1 dimension (M)  PS_InitArrays
<>AAIRCRAFTSPEED  Long integer 1 dimension (M)  PS_InitArrays
<>AAIRPORTGROUPID Long integer 1 dimension (M)  PS_InitArrays
<>AAIRPORTIATACODE Fixed string:3 1 dimension (M)  PS_InitArrays
<>AAIRPORTID     Long integer 1 dimension (M)  PS_InitArrays
<>AAIRPORTLEVEL  Long integer 1 dimension (M)  GetCitiesOfContry
<>ACAPITALAIRPORTS1 Text      1 dimension (M)  Compiler_Arrays_Inter
<>ACAPITALAIRPORTS2 Text      1 dimension (M)  Compiler_Arrays_Inter
<>ACAPITALAIRPORTSID1 Long integer 1 dimension (M)
Compiler_Arrays_Inter
<>ACAPITALAIRPORTSID2 Long integer 1 dimension (M)
Compiler_Arrays_Inter
<>ACITY          Text          1 dimension (M)  PS_InitArrays
<>ACOMPANYID     Long integer 1 dimension (M)  PS_InitArrays
<>ACOMPANYNAME   Text          1 dimension (M)  PS_InitArrays
<>ACOUNTRY       Text          1 dimension (M)  PS_InitArrays
<>ACOUNTRYCITY  Text          1 dimension (M)  GetCitiesOfContry
<>ACOUNTRYCITYID Long integer 1 dimension (M)  GetCitiesOfContry
```

Der Kopfteil zeigt den Namen der Datenbank sowie Datum und Uhrzeit der Erstellung. Das Dokument ist in vier Bereiche gegliedert:

- Liste der Interprozessvariablen
- Liste der Prozessvariablen
- Liste der lokalen Variablen in ihren Methoden
- Komplette Liste der Projekt- und Datenbankmethoden mit den dazugehörigen Parametern – sofern vorhanden

Liste der Prozess- und Interprozessvariablen

Diese beiden Listen sind in vier Spalten unterteilt:

- Die erste Spalte enthält die Namen der Variablen und Arrays, die in Ihrer Datenbank verwendet werden. Sie erscheinen in alphabetischer Reihenfolge.
- Die zweite Spalte enthält die Art der Variablen. Die Arten werden über die Compiler-Befehle bestimmt oder über den Compiler, der sich nach der Verwendung der Variablen richtet. Lässt sich der Variablentyp nicht bestimmen, ist die Spalte leer.
- Die dritte Spalte zeigt die Anzahl der Dimensionen, wenn die Variable vom Typ Array ist
- Die vierte Spalte enthält eine Referenz auf den Kontext, in welchem der Compiler den Variablentyp bestimmt hat. Wird die Variable in verschiedenen Kontexten verwendet, wird der Kontext genannt, welchen der Compiler zur Bestimmung des Typs verwendet.
 - Wurde die Variable in einer Datenbankmethode gefunden, wird ihr Name wiedergegeben wie in 4D definiert, beginnend mit **(M)***.
 - Wurde die Variable in einer Projektmethode gefunden, wird sie identifiziert wie in 4D definiert, beginnend mit **(M)**.
 - Wurde die Variable in einem Trigger (Tabellenmethode) gefunden, wird der Tabellenname wiedergegeben, beginnend mit **(TM)**.
 - Wurde die Variable in einer Formularmethode gefunden, wird der Formularname wiedergegeben, beginnend mit **(FM)** und mit vorangestelltem Tabellennamen.

- Wurde die Variable in einer Objektmethode gefunden, wird der Name der Objektmethode wiedergegeben, beginnend mit **(OM)** und mit vorangestelltem Formular- und Tabellennamen.
- Ist die Variable ein Objekt in einem Formular und erscheint sie weder in einer Projekt-, Formular-, Objektmethode oder Trigger, wird der Name des Formulars wiedergegeben, in dem sie erscheint, beginnend mit **(F)**.

Am Ende jeder Liste finden Sie die Größen der Prozess- und Interprozessvariablen in Bytes.

Hinweis: Der Compiler kann beim Kompilieren nicht feststellen, in welchem Prozess eine bestimmte Prozessvariable verwendet wird. Eine Prozessvariable kann in jedem Prozess einen anderen Wert haben. Folglich werden alle Prozessvariablen systematisch dupliziert, wenn ein neuer Prozess gestartet wird. Von daher empfiehlt es sich, auf den Speicherplatz zu achten, den sie beanspruchen. Bedenken Sie auch, dass der Platz für Prozessvariablen nicht mit der Stapelgröße für den Prozess zusammenhängt.

Liste der lokalen Variablen

Die Liste der lokalen Variablen wird in derselben Reihenfolge wie in 4D sortiert, d.h. in der Reihenfolge Datenbankmethode, Projektmethode, Trigger (Tabellenmethode), Formularmethode und Objektmethode.

Die Liste ist in drei Spalten gegliedert:

- Die erste Spalte enthält die Liste der lokalen Variablen, die in Methoden verwendet werden
- Die zweite Spalte enthält die Art der Variablen
- Die dritte Spalte zeigt die Nummer der Dimensionen, wenn die Variable ein Array ist

Komplette Liste der Methoden

Am Ende der Datei erscheint die komplette Liste ihrer Datenbank- und Projektmethoden mit

- dem Typ (Vorgang oder Funktion, die einen Wert zurückgibt)
- den Datentypen ihrer Parameter und dem zurückgegebenen Ergebnis
- Anzahl der Aufrufe
- Der Eigenschaft thread-safe oder thread-unsafe (siehe **Preemptive 4D Prozesse**)

Diese Information erscheint in folgender Form:

Vorgang oder Funktion <Methodenname>(Parameter Datentypen):Ergebnis Datentyp, Anzahl der Aufrufe, Thread Safe oder Thread Unsafe

Fehlerdokument

Sie können wählen, ob Sie während dem Kompilieren ein Fehlerdokument erzeugen wollen. Dazu wählen Sie in den Datenbank-Eigenschaften auf der Seite Compiler die Option „Erzeuge Fehlerdokument“ (siehe **Optionen zum Kompilieren**). Das erzeugte Fehlerdokument erhält den Namen *DatenbankName_errors.xml* und wird neben der Strukturdatei der Datenbank angelegt.

Die Fehler sind zwar direkt im Compiler-Fenster verfügbar. Ein Fehlerdokument ist jedoch hilfreich, denn es lässt sich leicht auf andere Rechner übertragen, insbesondere wenn mehrere Entwickler in einer Client/Server-Umgebung zusammenarbeiten. Das Fehlerdokument wird im XML Format erstellt, um das automatische Analysieren seines Inhalt zu erleichtern. Damit können Sie auch eigene Formate zum Anzeigen der Fehler erstellen.

Seine Länge richtet sich nach der Anzahl der Fehler und Warnungen, die der Compiler anzeigt. Öffnen Sie das Fehlerdokument über einen Texteditor, könnte es folgendermaßen aussehen:

```
<?xml version="1.0" encoding="iso-8859-1"??>
<!-- Created by 4th Dimension on: Thu, 09 Oct 2003 14:38:42 GMT-->
<error_list>
  <method name="4DSEL_ManyToMany">
    <error line="12" warning="true">Pointer in SELECTION TO ARRAY</error>
  </method>
  <method name="4DSortDefine">
    <error line="87" warning="true">Pointer in an array declaration</error>
    <error line="88" warning="true">Pointer in an array declaration</error>
  </method>
  <method name="MAP_FindAirlinesGroupToGroupOLD">
    <error line="13" warning="true">Pointer in an array declaration</error>
    <error line="108" warning="true">Pointer in an array declaration</error>
    <error line="116" warning="true">Pointer in an array declaration</error>
    <error line="297" warning="true">Pointer in an array declaration</error>
    <error line="309" warning="true">Pointer in an array declaration</error>
  </method>
  <method name="MAP_FindAirlinesGroupToGroup">
    <error line="15" warning="true">Pointer in an array declaration</error>
    <error line="113" warning="true">Pointer in an array declaration</error>
    <error line="121" warning="true">Pointer in an array declaration</error>
    <error line="306" warning="true">Pointer in an array declaration</error>
    <error line="318" warning="true">Pointer in an array declaration</error>
  </method>
  <method name="[AIRPORTS].Input2000">
    <error line="241" warning="true">Pointer in an array declaration</error>
    <error line="242" warning="true">Pointer in an array declaration</error>
    <error line="124" warning="true">Missing parameter in the plug-in procedure call.</error>
    <error line="142" warning="true">Missing parameter in the plug-in procedure call.</error>
    <error line="146" warning="true">Missing parameter in the plug-in procedure call.</error>
    <error line="149" warning="true">Missing parameter in the plug-in procedure call.</error>
  </method>
</error_list>
```

Das Dokument ist folgendermaßen gegliedert:

- Der Kopfteil zeigt den Namen der Datenbank sowie Datum und Uhrzeit der Erstellung
- Im Bereich *****General errors***** werden alle Typisierungsprobleme und nicht eindeutige Zuordnungen aufgelistet. Diese Fehler und Warnungen enthalten folgende Elemente:
 - Erstens, Zeilennummer in der Methode. Allgemeine Fehler haben die Nummer 0 (Null);
 - Status der Warnung: er gibt an, ob es sich um eine Warnung (warning="true") bzw. einen Fehler (warning="false") handelt;

- Drittens, Beschreibung des Fehlers, z.B. >Fehlender Parameter in ...<
Enthält Ihre Datenbank keine allgemeinen Fehler, fällt dieser Teil weg.

Eine Fehlerdatei kann drei Arten von Meldungen enthalten:

- Fehler in einer spezifischen Zeile
- Allgemeine Fehler
- Warnungen

Fehler in einer spezifischen Zeile

Diese Fehler erscheinen im Kontext, d.h. in der Zeile, wo sie gefunden werden – mit einer Erklärung. Der Compiler meldet diese Art Fehler, wenn er einen Ausdruck findet, in dem eine Inkonsistenz in Bezug auf den Datentyp oder die Syntax vorliegt.

Doppelklicken Sie im Compiler-Fenster auf jeden gefundenen Fehler, um die unmittelbar betroffene Methode im 4D Methodeneditor mit dem in der Zeile hervorgehobenen Fehler zu öffnen.

Die Liste der Syntax- bzw. Typisierungsfehler finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Fehlermeldungen**.

Allgemeine Fehler

Treten diese Fehler auf, kann die Datenbank nicht kompiliert werden. Der Compiler meldet in folgenden Fällen einen allgemeinen Fehler:

- Der Datentyp einer Prozessvariablen ließ sich nicht bestimmen
- Zwei unterschiedliche Objektarten haben denselben Namen

Diese Fehler sind allgemein, da sie sich keiner bestimmten Methode zuordnen lassen. Im ersten Fall kann der Compiler keine spezifische Typisierung in der Datenbank durchführen. Im zweiten Fall kann er einen bestimmten Namen nicht eindeutig einem Objekt zuordnen.

Die Liste der allgemeinen Fehler finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Fehlermeldungen**.

Warnungen

Warnungen sind keine Fehler. Eine Datenbank lässt sich trotz Warnungen kompilieren, sie geben lediglich potentielle Fehlercodes an.

Warnungen erscheinen im Compiler-Fenster in Kursivschrift. Doppelklicken Sie auf jede Warnung, um die unmittelbar betroffene Methode im 4D Methodeneditor mit der in der Zeile hervorgehobenen Warnung zu öffnen.

Die Liste der Warnungen finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Warnungen**.

Bestimmte Warnungen können Sie auch deaktivieren (siehe Abschnitt **Warnungen während dem Kompilieren deaktivieren**).

Bereichsprüfung

Bereichsprüfung ist standardmäßig in den Datenbank-Eigenschaften markiert. Weitere Informationen dazu finden Sie im Abschnitt **Optionen zum Kompilieren**).

Bereichsprüfung beginnt nicht, wie bei den anderen Optionen, während dem Kompilieren, sondern erst, wenn die kompilierte Datenbank läuft. Das ist eine zusätzliche Analyse des Codes auf logische Übereinstimmung und schlüssige Syntax. Die Bereichsprüfung bewertet den Status von Objekten in der Datenbank zu einem bestimmten Zeitpunkt.

Hier ein Beispiel: Nehmen wir an, Sie haben das Array `MeinArray` als Text deklariert. Die Anzahl der Elemente in `MeinArray` variiert je nach der aktuellen Methode. Um den Wert „Hallo“ dem Element 5 von `MeinArray` zuzuordnen, schreiben Sie:

```
MeinArray{5}:="Hallo"
```

Hat das Array zu dieser Zeit fünf oder mehr Elemente, ist alles in Ordnung. Die Zuweisung erfolgt ganz normal. Hat `MeinArray` dagegen zu diesem Zeitpunkt weniger als fünf Elemente, macht Ihre Zuweisung keinen Sinn.

Solch eine Situation lässt sich beim Kompilieren nicht feststellen, da davon ausgegangen wird, dass die Methoden ausgeführt werden. Der Compiler kennt nicht die Umstände, unter denen diese Methode aufgerufen wird. Nur über die Bereichsprüfung können Sie überwachen, was gerade passiert, wenn die Datenbank läuft. Im obigen Beispiel würde der Compiler einen Fehler bei der Ausführung innerhalb von 4D melden. In diesem Fall ist die Bereichsprüfung eine wertvolle Hilfe beim Arbeiten mit Arrays, Zeigern und Zeichenketten.

Fordern Sie die Bereichsprüfung an, erscheinen die vom Compiler gesendeten Meldungen in einem spezifischen Fehler-Dialogfenster. Weitere Informationen finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Meldung bei Bereichsprüfung**.

Bereichsprüfung lokal deaktivieren

Ist die Bereichsprüfung aktiviert, wollen Sie diese in manchen Fällen auf bestimmte Teile des Code, die zuverlässig arbeiten, nicht anwenden. Das gilt insbesondere für Schleifen, die viele Male wiederholt werden. Hier kann die Bereichsprüfung die Bearbeitung signifikant verlangsamen, wenn die kompilierte Datenbank auf älteren Rechnern läuft. Sind Sie sicher, dass der betreffende Code korrekt ist und keine Systemfehler verursacht, können Sie die Bereichsprüfung für diese Teile des Codes deaktivieren.

Dazu versehen Sie den entsprechenden Code mit den Kommentaren `//%R-` und `//%R+`. `//%R-` deaktiviert die Bereichsprüfung, `//%R+` aktiviert sie wieder:

```
... //Bereichsprüfung ist aktiviert  
// %R-
```

```
... //Setzen Sie hier Code, der von der Bereichsprüfung ausgenommen ist
// %R+
... //Bereichsprüfung ist wieder aktiviert für die restliche Methode
```

Hinweis: Diese Operation funktioniert nur, wenn die Bereichsprüfung aktiviert ist.












Unregelmäßigkeiten behandeln

Nehmen wir an, Sie stoßen beim Laufen der Datenbank auf Unregelmäßigkeiten. Ehe Sie über die Ursachen dafür spekulieren, sollten Sie die vom Compiler angebotene Unterstützung nutzen.

Es gibt verschiedene Möglichkeiten:

- 4D zeigt seine eigenen Fehlermeldungen an.
Korrigieren Sie Ihre Datenbanken, soweit möglich, gemäß den Anweisungen von 4D. Sind sie zu allgemein, kompilieren Sie Ihre Datenbank erneut, vergewissern Sie sich, dass die Option **Bereichsprüfung** aktiviert ist. Testen Sie Ihre Datenbank erneut. Dort, wo anfangs die Fehlermeldung erschienen ist, setzt der Compiler eine Meldung mit weiteren Informationen.
- Ihre kompilierte Datenbank arbeitet anders als die interpretierte Datenbank. Sehen Sie sich die Warnungen genauer an.
- Variablen vom Typ Zahl oder String geben nicht die erwarteten Werte zurück. Überprüfen Sie in den Einstellungen der Datenbank die standardmäßig definierten Optionen zur Typisierung und prüfen Sie in der Symboldatei, ob all Ihre Variablen korrekt typisiert sind.
- Ihre Datenbank funktioniert im interpretierten Modus, im kompilierten Modus stürzt sie ab. Stellen Sie sicher, dass die Datenbank mit der Option **Bereichsprüfung** kompiliert wurde und überprüfen Sie, ob die kompilierte Datenbank dieselben Plug-Ins verwendet, wie während dem Kompilieren.

Eigenständige Anwendung erstellen und weitergeben

-  Möglichkeiten zur Weitergabe
-  Anwendung erstellen
-  Seite Kompilierte Struktur
-  Seite Anwendung
-  Seite Client/Server
-  Seite Plug-Ins und Komponenten
-  Seite Lizenzen und Zertifikate
-  Standard Datendatei in doppelklickbaren Anwendungen verwalten
-  Verbindungsprozess für eingebundene Clients
-  Automatisches Updaten von Server oder Einzelplatzanwendungen
-  Icon der eigenständigen Anwendung anpassen

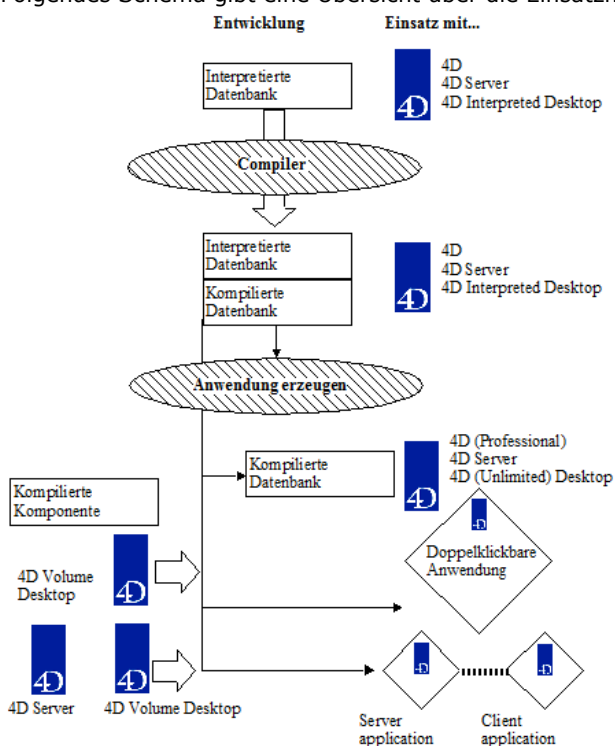
Möglichkeiten zur Weitergabe

4D bietet die Möglichkeit, Ihre Datenbank als eigenständige Anwendung weiterzugeben. Der "Application Builder" vereinfacht die Fertigstellung und Weitergabe kompilierter 4D Anwendungen. Er unterstützt automatisch die spezifischen Funktionalitäten der verschiedenen Betriebssysteme, insbesondere die Erstellung von Software Paketen unter Mac OS und den Einsatz im Client/Server-Betrieb.

Mit dieser Funktion erzeugen Sie:

- Eine kompilierte Datenbank ohne interpretierten Code
- Eine eigenständige, doppelklickbare Anwendung mit integrierter 4D Volume Desktop, d.h. das 4D Programm selbst ist nicht mehr erforderlich
- Über ein XML Projekt verschiedene Anwendungen aus derselben kompilierten Datenbank
- Aufeinander abgestimmte Client/Server Anwendungen
- Client/Server-Anwendungen mit automatischer Aktualisierung von Client und Server Bereich

Hinweis zu 4D Server: Das Erstellen eigenständiger Anwendungen ist nur in der Einzelplatzversion von 4D verfügbar. Folgendes Schema gibt eine Übersicht über die Einsatzmöglichkeiten für 4D Datenbanken:

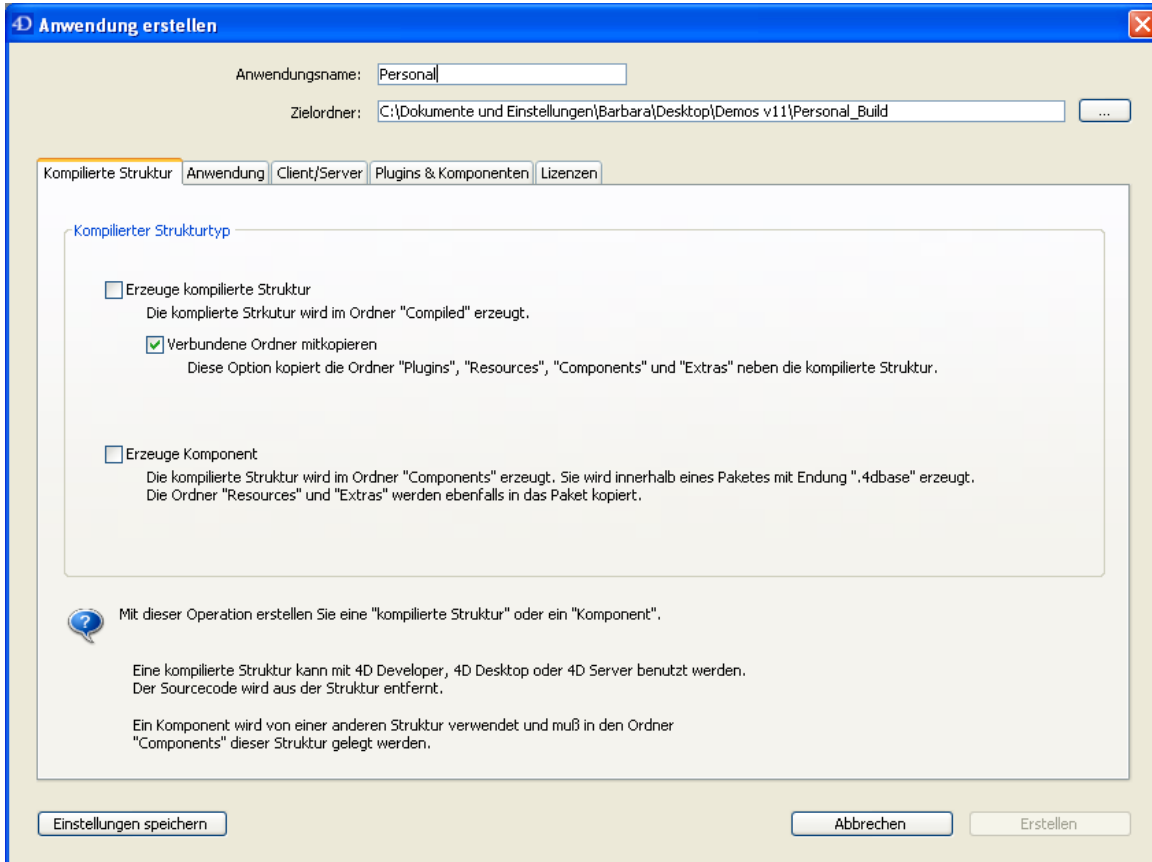


Anwendung erstellen

Die Anwendung wird im Fenster "Anwendung erstellen" fertiggestellt. Gehen Sie dazu in die Designumgebung und wählen im Menü **Design>Anwendung erstellen**.

Die Fertigstellung ist nur für kompilierte Datenbanken möglich. Wählen Sie den Befehl ohne vorherige Kompilierung oder entspricht der kompilierte Code nicht dem interpretierten Code, erscheint eine Meldung, dass die Datenbank kompiliert werden muss.

Das Dialogfenster enthält mehrere Registerkarten, über die Sie auf die verschiedenen Seiten gehen können:



- Die **Seite Kompilierte Struktur** erstellt eine kompilierte Version nur von der Struktur oder einer Komponente
- Die **Seite Anwendung** erstellt Sie die Einzelplatzversion der Anwendung
- Die **Seite Client/Server** richtet die Mehrplatzversion der Anwendung ein.
- Die **Seite Plug-Ins und Komponenten** legt fest, welche Plug-Ins und Komponenten in die Anwendung integriert werden
- Die **Seite Lizenzen und Zertifikate** definiert die zu integrierenden Lizenznummern

Sie können für die Anwendung gleichzeitig verschiedene Zielparameter setzen, also Einzelplatz und Client/Server; jeder muss explizit über das entsprechende Kontrollkästchen gewählt werden.

Klicken Sie auf die Schaltfläche **Erstellen**, wird die Anwendung gemäß den gesetzten Zielparametern erstellt. 4D zeigt in einem Ablaufbalken die Ausführung der verschiedenen Phasen an. Die neuen Parameter werden im Anwendungsprojekt gespeichert (siehe).

Klicken Sie auf die Schaltfläche **Sichern**, werden die definierten Parameter im Projekt gespeichert, die Anwendung wird jedoch nicht erzeugt.

Klicken Sie auf die Schaltfläche **Abbrechen**, wenn während dem Vorgang ein Fehler auftritt. Die Dateien im Aufbau werden dann entfernt und Sie erhalten eine Meldung über die Fehlerursache.

Name und Ziel der Anwendung bestimmen

Im oberen Bereich des Fensters legen Sie Name und Ziel der zu generierenden Dateien fest.

Der Bereich **Anwendungsname** enthält standardmäßig den Namen der Strukturdatei der Datenbank. Er wird für die generierten Dateien verwendet, d.h. die kompilierte Datenbank, Komponente, doppelklickbare Anwendung und die Client/Server Anwendung. 4D fügt automatisch die jeweils passende Endung, wie .4dc, .exe, server... hinzu. Behalten Sie den Standardnamen bei, gibt er jede Änderung am Namen der Strukturdatei wieder. Ändern Sie den Namen der Anwendung, wird standardmäßig dieser Name für jedes neue Erzeugen der aktuellen Datenbank verwendet. Er darf weder eine Endung noch vom Betriebssystem nicht zugelassene Zeichen enthalten. Das sind z.B. unter Windows die Zeichen ".?!", auf Mac OS ":".

Im Bereich **Zielordner** geben Sie den Speicherort für die generierten Dateien an. 4D verwendet standardmäßig einen Ordner mit Namen *Strukturname_Build*, der neben dem Ordner der gewählten Datenbank liegt (auf Mac OS neben dem Software-Paket).

Um den Zielordner zu ändern, klicken Sie rechts neben der Anzeige auf die Schaltfläche [...]. Auf dem Bildschirm erscheint ein Auswahldialog, um den neuen Zielordner anzugeben. Sobald Sie diesen Dialog bestätigen, wird der vollständige Zugriffspfad des Ordners angezeigt. Der neue Pfad wird standardmäßig für jedes nachfolgende Erstellen der aktuellen Datenbank verwendet.

Beim Erstellen legt 4D am angegebenen Ort einen oder mehrere Zwischenordner an, die je nach Art der Einstellung "Kompilierte Datenbank", "Final Application", "Komponente", „Anwendungsname Client“ oder „Anwendungsname Server“ lauten. Das verhindert, dass Dateien mit demselben Namen versehentlich gelöscht werden. Außerdem können Sie mehrere Arten gleichzeitig ausführen.

XML Schlüssel der Parameter

Jeder Parameter dieses Fensters wird als XML Schlüssel im Projektordner mit Namen "BuildApp.XML" gespeichert und in den Ordner **Preferences** der Datenbank im Unterordner **BuildApp** gelegt.

Beim ersten Aufruf des Dialogfensters werden die Standardparameter verwendet. Klicken Sie auf die Schaltfläche **Erstellen** oder **Sichern**, wird der Inhalt der Projektdatei bei Bedarf aktualisiert.

Sie können für dieselbe Datenbank mehrere XML-Projekte definieren und über den 4D Befehl **BUILD APPLICATION** einsetzen.

Beachten Sie, dass XML Schlüssel zusätzlich zu den Optionen zum Erstellen einer fertigen Anwendung weitere Funktionalitäten bieten. Sie können:

- Eine IP Adresse oder die TCP Portnummer des Server bestimmen
- Den Bereich kompatibler Versionsnummern zwischen der Client- und der Server Anwendung definieren, z.B. dass sich Client Anwendungen 1.1 bis 1.3 an die Server Anwendung 1.3 anmelden können.
- Einen Pfadnamen für die Datendatei festlegen, der zum Erstellen von Anwendungen dient. Sie sind dann sofort einsetzbar, ohne dass auf dem Rechner des Benutzers eine Datendatei zugewiesen werden muss
- Zugriffsrechte unter Windows aufheben, um die automatische Installation doppelklickbarer Anwendungen in geschützten Systembereichen zu erlauben

Weitere Informationen dazu finden Sie im Handbuch **4D XML Keys BuildApplication**.

Logbuch

Wird eine Anwendung zum ersten Mal erstellt, erstellt 4D ein Logbuch in zwei Formaten: xml und xml/html. Diese Dateien lauten *Projektname.log.xml* und *Projektname.log_html.xml*. *Projektname* ist der Name des Anwendungsprojekts, z.B. standardmäßig BuildApp. Sie werden im Ordner **Logs** neben der Strukturdatei der Anwendung angelegt. Für jedes Anwendungsprojekt werden diese beiden Logbücher angelegt.

Der Inhalt beider Dateien ist identisch, nur das Format ist unterschiedlich. Beide Dateien werden beim Aufbau einer Anwendung mit derselben Information aktualisiert. Im xml/html Format werden Fehler und Warnungen grafisch dargestellt.

Das Logbuch speichert folgende Informationen:

- Start und Ende der erstellten Zieldateien
- Name und kompletten Zugriffspfad der erstellten Dateien
- Datum und Zeit der Erstellung
- Alle aufgetretenen Fehler

Auf der Seite **Kompilierte Struktur** können Sie eine standardmäßig kompilierte Strukturdatei bzw. Komponente anlegen. Diese Dateien sind in beiden Fällen gleich. Nur die Architektur der angelegten Ordner ist unterschiedlich. Beim Erzeugen erstellt 4D automatisch mehrere Ordner zum Ablegen der generierten Dateien. Egal welche Option gewählt ist, die aktuelle Datenbank wird nicht geändert, sondern es werden Kopien auf der Festplatte angelegt. Enthalten die kompilierten Strukturdateien sowohl 32-bit als auch 64-bit Code (siehe **Optionen zum Kompilieren**), werden die entsprechenden Versionen der kompilierten Strukturdateien erzeugt. Haben Sie die Optionen definiert, klicken Sie auf die Schaltfläche **Erstellen**, um die gewünschten Dateien am angegebenen Ort anzulegen.

Kompilierte Struktur erzeugen

Mit dieser Option erstellen Sie eine Datenbank, die nur kompilierten Code enthält. Haben Sie z.B. als Anwendungsname "MeineDatenbank" definiert, erhalten Sie:

- eine Datei mit Namen *MeineDatenbank.4dc* (kompilierte Datei der Datenbank)
- eine Datei mit Namen *MeineDatenbank.4DIndy* (Indexdatei der Struktur)

Die Datei .4dc lässt sich mit 4D, 4D Server oder 4D Desktop öffnen. 4D löscht beim Erstellen den interpretierten Code der Strukturdatei. Sie können dann nicht mehr auf den Designmodus dieser Datenbank zugreifen.

Die kompilierte Datenbank wird folgendermaßen abgelegt:

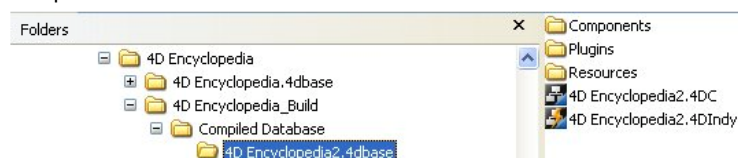
- In einen Ordner vom Typ *MeineDatenbank.4dbase*, der auf Mac OS spezifische Eigenschaften hat (siehe Handbuch *4D Programmiersprache*, **Ordner .4dbase**) — *MeineDatenbank* ist der Name der Anwendung
- Dieser Ordner liegt im Ordner mit Namen *Compiled Database*
- Dieser wiederum wird im angegebenen "Zielordner" erstellt.

Warnung: 4D löscht vor Erzeugen einer kompilierten Datenbank den Inhalt des Ordners "Compiled Database". Wollen Sie Zwischenversionen behalten, müssen Sie die Anwendung umbenennen oder kompilierte Versionen bzw. weitere Elemente daraus entfernen und an anderer Stelle sichern.

Verbundene Ordner mitkopieren

Markieren Sie diese Option, werden mit der Datenbank verbundene Ordner in den Ordner **Compiled Database** kopiert. Das sind "Plugins," "Resources," "Components". Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Architektur der Anwendung**.

Auf diese Weise können Sie einsatzbereite kompilierte Strukturdateien erstellen. Hier ein Beispiel für die Ordner einer kompilierten Struktur:



Komponente erzeugen

Mit dieser Option erstellen Sie eine kompilierte Komponente aus der Struktur.

Eine Komponente ist eine standardmäßige 4D Strukturdatei, in der spezifische Funktionalitäten entwickelt wurden. Ist sie konfiguriert und in einer anderen Datenbank (der Host Datenbank) installiert, sind ihre Funktionalitäten aus der Host Datenbank verfügbar. Weitere Informationen dazu finden Sie im Abschnitt **4D Komponenten entwickeln und installieren**.

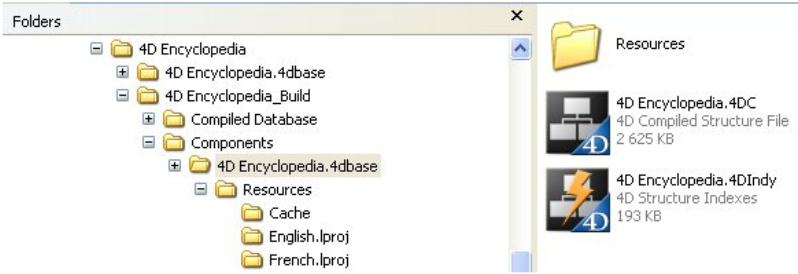
Haben Sie im Bereich Anwendungsname "MyComponent" definiert, erhalten Sie:

- Eine Datei mit Namen *MyComponent.4dc* (kompilierte Strukturdatei)
- Eine Datei mit Namen *MyComponent.4DIndy* (Indexdatei der Struktur)

Die erzeugten Elemente sind ähnlich wie die einer kompilierten Struktur. Es gibt jedoch folgende Unterschiede:

- Die Komponente wird als Ordner bzw. Paket in einem Ordner mit Namen „Components“ angelegt.
- Der zugewiesene Ordner „Resources“ wird automatisch in den Ordner der Komponente kopiert. Dagegen werden die anderen Ordner „Components“ bzw. „Plugins“ nicht kopiert. Eine Komponente kann weder Plug-Ins noch andere Komponenten verwenden.

Hier ein Beispiel für die Ordner einer Komponente:



Mit 4D können Sie eine doppelklickbare Anwendung direkt aus der Datenbank erzeugen. Sie benötigen lediglich 4D Volume Desktop, die 4D Engine der Datenbank und eine entsprechende Lizenz. Sie bereiten diese Operation auf der Seite **Anwendung** des Application Builders vor.

Mit der Option "Erzeuge doppelklickbare Anwendung" können Sie direkt aus 4D eigenständige Versionen (.exe) Ihrer kompilierten 4D Datenbanken erstellen.

Diese Funktion verwaltet auf Mac OS auch das Erzeugen von Software Paketen.

Das Prinzip ist, eine kompilierte Strukturdatei mit 4D Volume Desktop zusammenzuführen. Die jeweils gelieferte Funktionalität richtet sich nach dem erworbenen Produkttyp. Weitere Informationen dazu finden Sie der 4D Produktbroschüre und im Internet unter (<http://www.4d.com/de/products.html>).

Sie können eine Standard Datendatei definieren oder Benutzern ermöglichen, ihre eigene Datendatei anzulegen und zu verwenden. Weitere Informationen dazu finden Sie im Abschnitt **Standard Datendatei in doppelklickbaren Anwendungen verwalten**.

Das Update doppelklickbare Einzelplatz-Anwendungen lässt sich über Befehle der Programmiersprache auch automatisieren. Weitere Informationen dazu finden Sie im Abschnitt **Automatisches Updaten von Server oder Einzelplatzanwendungen**.

Ordner 4D Volume Desktop auswählen

Vor Erzeugen einer doppelklickbaren Anwendung müssen Sie zuerst den Pfad des Ordners *4D Volume Desktop* festlegen. Die Optionen zum Erstellen einer doppelklickbaren Anwendung sind inaktiv, solange im darunterliegenden Bereich kein Ordner angegeben wurde oder der definierte Ordner keine gültige Datei *4D Volume Desktop* enthält.

Sie müssen den Ordner mit der Datei 4D Volume Desktop auswählen:

- Unter Windows enthält der Ordner die Dateien 4D Volume Desktop.4DE, 4D Volume Desktop.RSR und verschiedene Dateien und Ordner zum Ausführen der Operation. Sie müssen auf derselben Ebene wie die gewählten Ordner liegen.
- Auf Mac OS steht *4D Volume Desktop* in Form eines Software Pakets mit verschiedenen generischen Dateien und Ordnern zur Verfügung.

Um den Ordner *4D Volume Desktop* auszuwählen, klicken Sie unter dem Anzeigebereich auf die Schaltfläche [...]. Auf dem Bildschirm erscheint ein Dialogfenster, um unter Windows den Ordner 4D Volume Desktop, auf Mac OS das Software-Paket zu bestimmen. Haben Sie den Ordner gewählt, erscheint der vollständige Pfad. Ist *4D Volume Desktop* enthalten, sind die Optionen zum Aufbau einer doppelklickbaren Anwendung aktiviert:

Anwendungstyp

Erzeuge doppelklickbare Anwendung

Die Anwendung wird im Ordner "Final application" im Zielordner erzeugt.
Die Ordner "Plugins", "Resources", "Extras" und "Components" neben der Struktur werden ebenfalls kopiert.

4D Volume Desktop Ordner: C:\Dokumente und Einstellungen\Barbara\Desktop\4D v11 SQL release 1\Deployment\4D Volume Desktop\

...

Hinweis: Ab 4D v15 muss die Versionsnummer von 4D Volume Desktop zur Versionsnummer von 4D Developer Edition passen. Verwenden Sie z.B. 4D Developer v15.4, müssen Sie auch 4D Volume Desktop in v15.4 wählen.

Datendatei-Verknüpfungsmodus

Damit wählen Sie den Verknüpfungsmodus zwischen doppelklickbarer Anwendung und der lokalen Datendatei. Es gibt zwei Arten mit jeweiligen Vor- und Nachteilen. Wählen Sie die Art, die am besten zu Ihren Anforderungen passt:

Anwendung erstellen

Anwendungsname: Angestellte

Zielordner: "Angestellte_Build" in Volume "C:"

Kompilierte Struktur | Anwendung | Client/Server | Plugins & Komponenten | Lizenzen und Zertifikate

Erzeuge doppelklickbare Anwendung

Die Anwendung wird im Ordner "Final application" im Zielordner erzeugt. Die Ordner "Plugins", "Resources" und "Components" neben der Struktur werden ebenfalls kopiert.

4D Unlimited Desktop Ordner: Wählen Sie die 4D Volume Desktop

Datendatei-Verknüpfungsmodus basiert auf: Anwendungsname

Hinweis: Diese Option ist nur verfügbar, wenn die Option **Benutze neue Architektur für Anwendungen-Weitergabe** auf der Seite Kompatibilität markiert ist (siehe Abschnitt [Seite Kompatibilität](#)).

- Über **Anwendungsname** (Standard)
Damit öffnet die 4D Anwendung automatisch die zuletzt geöffnete Datendatei, die zur Strukturdatei gehört. Mit diesem flexiblen und intuitiven Modus können Sie das Anwendungspaket frei auf der Festplatte bewegen. Das ist die reguläre Nutzung für doppelklickbare Anwendungen, außer Sie müssen Ihre Anwendung duplizieren.
- Über **Anwendungspfad**
Damit analysiert die doppelklickbare 4D Anwendung die Datei *lastDataPath.xml* der Anwendung und versucht, die Datendatei mit einem Attribut "executablePath" zu öffnen, die zum vollständigen Pfad der Anwendung passt. Wird solch eine Eingabe gefunden, wird die dazugehörige Datendatei geöffnet (definiert durch ihr Attribut "dataFilePath"). Andernfalls wird die zuletzt geöffnete Datendatei geöffnet (Standardmodus).
Mit dieser Option können Sie Ihre doppelklickbaren Anwendungen duplizieren, ohne die Verbindung zur Datendatei zu unterbrechen (siehe oben).
Wird dagegen das Anwendungspaket auf der Festplatte verschoben, wird der Benutzer nach der Datendatei gefragt, da der Pfad der Anwendung nicht mehr mit dem Attribut "executablePath" übereinstimmt (hat ein Benutzer eine Datendatei gewählt, wird die Datei *lastDataPath.xml* entsprechend aktualisiert).

Weitere Informationen dazu finden Sie im Abschnitt [Datendatei-Verknüpfungsmodus wählen](#).

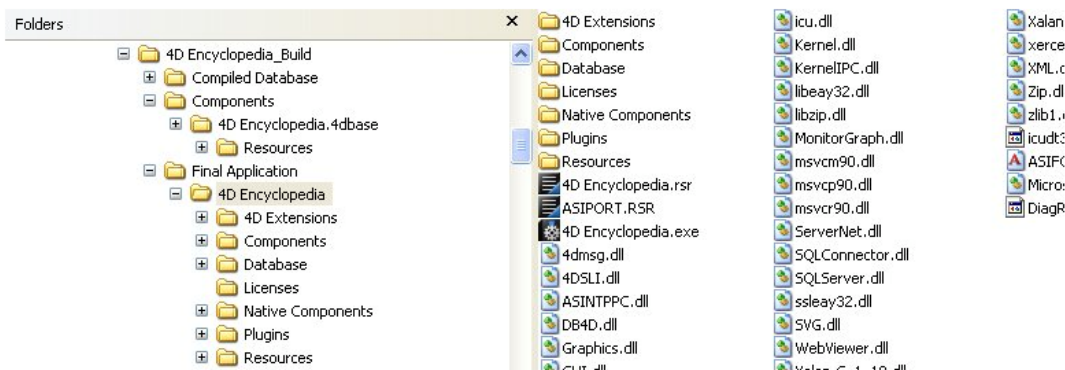
Generierte Dateien

Klicken Sie auf die Schaltfläche **Erstellen**, um die doppelklickbare Anwendung einzurichten. 4D erstellt im Zielordner automatisch den Unterordner **Final Application** und legt dort einen Unterordner mit dem Namen der Anwendung ab. Haben Sie z.B. im Bereich Anwendungsname „MeineAnw“ eingetragen, liegen dort folgende Ordner:

- Unter Windows
 - MeineAnw.EXE, Ihre ausführbare Anwendung und MeineAnw.RSR mit den Ressourcen der Anwendung.
 - Die Ordner 4D Extensions und Resources, sowie verschiedene Libraries (DLL) und zum Arbeiten der Anwendung erforderliche Dateien.
 - DatenbankName.4DC und DatenbankName.RSR mit der kompilierten Struktur und der Ordner Resources der Datenbank.
Hinweis: Dieser Ordner enthält auch den Ordner *Default Data*, wenn er definiert wurde. Weitere Informationen dazu finden Sie im Abschnitt [Standard Datendatei in doppelklickbaren Anwendungen verwalten](#).
 - (Optional) Ordner Components und PlugIns mit den evtl. in die Datenbank integrierten Komponenten bzw. Plug-Ins. Weitere Informationen dazu finden Sie auf der [Seite Plug-Ins und Komponenten](#).
 - Ordner Licenses mit der Liste der in die Anwendung integrierten Lizenznummern in Form einer XML Datei. Weitere Informationen dazu finden Sie auf der [Seite Lizenzen und Zertifikate](#).
 - Evtl. Zusatzelemente, die im Ordner 4D Volume Desktop hinzugefügt wurden. Weitere Informationen dazu finden Sie im Abschnitt [Ordner 4D Volume Desktop anpassen](#).
- Auf Mac OS
 - Ein Software Paket mit Namen „MeineAnw.app“. Es enthält Ihre Anwendung und alle dafür notwendigen Elemente, inkl. Plug-Ins, Komponenten und Lizenzen. Weitere Informationen finden Sie auf der [Seite Plug-Ins und Komponenten](#) und [Seite Lizenzen und Zertifikate](#).

Hinweis: Under Mac OS gibt die 4D Funktion **Application file** den Zugriffspfad zur Datei AnwendungName zurück (im Ordner Contents:Mac OS Ordner des Software Pakets) und nicht den Pfad zur Datei .comp (im Ordner Contents:Resources des Software Pakets).

Hier ein Beispiel für die Architektur einer doppelklickbaren Anwendung unter Windows:



Ordner 4D Volume Desktop anpassen

Beim Erzeugen einer doppelklickbaren Anwendung kopiert 4D den Inhalt des Ordners *4D Volume Desktop* in den Unterordner **Final Application** des Zielordners. Sie können dann den Inhalt des Originalordners *4D Volume Desktop* an Ihre Bedürfnisse anpassen. Sie können beispielsweise:

- Eine Version von 4D Volume Desktop für eine bestimmte Sprache einrichten
- Einen eigenen Ordner PlugIns hinzufügen
- Den Inhalt des Ordners Resources anpassen

Hinweis: Auf Mac OS wird 4D Volume Desktop als Software Paket geliefert. Um es zu verändern, müssen Sie zuerst seinen Inhalt anzeigen. Klicken Sie dazu bei gedrückter **ctrl-Taste** auf das Icon.

Position der Web Dateien

Werden Ihre doppelklickbaren Anwendungen als Web Server genutzt, müssen die dafür erforderlichen Dateien und Ordner an bestimmter Stelle installiert werden. Es gibt folgende Elemente:

- cert.pem und key.pem Dateien (optional): Diese Dateien werden für SSL-Verbindungen und von Befehlen zur Verschlüsselung der Daten verwendet
- Standard Web root Ordner,
- logweb.txt Datei (Logbuch für Web Anfragen)

Unter Windows müssen diese Dateien im Unterordner Final Application*MeineAnw*\Datenbank liegen.
Auf Mac OS müssen diese Dateien neben dem Software-Paket *MeineAnw.app* liegen.

In 4D können Sie eigene Client/Server Anwendungen erstellen, die homogen und crossplattform sind und eine automatische Update-Funktion bieten. Dazu gehen Sie im Dialogfenster „Anwendung erstellen“ auf die Seite **Client/Server**.

Was ist eine Client/Server Anwendung?

Eine Client/Server Anwendung ist die Kombination von drei Teilen:

- Eine kompilierte 4D Datenbank
- Eine 4D Server Anwendung
- Eine 4D Volume Desktop Anwendung (Mac OS und/oder Windows)

Ist sie eingerichtet, besteht sie aus dem Server-Teil (einmalig) und dem Client-Teil (zur Installation auf jedem Client-Rechner). Zum Vergleich rufen Sie sich ins Gedächtnis, dass zum Verwenden von 4D Server die 4D Server Applikation, die Strukturdatei und die Datendatei der Datenbank, sowie die 4D Applikation im remote Modus erforderlich sind.

Eine eigene Client/Server-Anwendung ist individuell angepasst und ihre Arbeitsweise ist vereinfacht:

- Zum Starten des Server-Teils muss der Benutzer nur auf die Server Anwendung doppelklicken. Die Strukturdatei muss nicht ausgewählt werden.
- Zum Starten des Client-Teils muss der Benutzer nur auf die Client Anwendung doppelklicken. Es muss keine Datenbank im Verbindungsdialog ausgewählt werden. Der Client findet den Server entweder über seinen Namen, wenn Client und Server auf dem gleichen Unter-Netzwerk liegen, oder über seine IP Adresse, die über den XML Schlüssel **IPAddress** in der Datei *buildapp.xml* definiert wurde (siehe nächster Abschnitt). Schlägt die Verbindung fehl, lassen sich spezifische alternative Mechanismen einbauen, die im Abschnitt **Verbindungsprozess für eingebundene Clients** beschrieben werden.
Die Anzeige dieses Dialogs lässt sich auch erzwingen. Starten Sie dazu die Client Anwendung unter Windows mit gedrückter **Alt-Taste**, auf Mac OS mit gedrückter **Wahltaste**.
Nur der zugehörige Client-Teil kann sich an den entsprechenden Server-Teil anmelden. Versucht ein Benutzer, sich an den Server-Teil mit einer Standard 4D Anwendung anzumelden, erscheint eine Fehlermeldung und die Verbindung ist nicht möglich.

Es lässt sich auch eine Client/Server Anwendung einrichten, bei der sich der Client-Teil automatisch über das Netzwerk aktualisieren lässt. Weitere Informationen dazu finden Sie im Abschnitt **Kopie der Client Anwendung innerhalb der Server Anwendung**.

Das Update des Server lässt sich über eine Reihe Befehle der 4D Programmiersprache auch automatisieren. Weitere Informationen dazu finden Sie im Abschnitt **Automatisches Updaten von Server oder Einzelplatzanwendungen**.

Client/Server-Anwendung

- **Server Anwendung erstellen:** Mit dieser Option richten Sie den Server-Teil Ihrer Anwendung ein. Sie müssen den Speicherort der 4D Server Anwendung auf Ihrer Festplatte angeben. Sie muss zur aktuellen Plattform passen, die dann auch die Plattform der Server Anwendung wird.
Klicken Sie auf die Schaltfläche [...], und wählen im Dialogfenster den Ordner *4D Server* aus, in welchen die Anwendung gelegt werden soll. Auf Mac OS müssen Sie direkt das 4D Server Paket auswählen.
- **Aktuelle Version:** Hier geben Sie die aktuelle Versionsnummer für die angelegte Anwendung an. Sie können dann Anmeldungen von Client Anwendungen je nach ihrer Versionsnummer akzeptieren oder zurückweisen. Das Intervall für Kompatibilität zwischen Client- und Server-Anwendung wird über spezifische XML Keys eingerichtet. Weitere Informationen dazu finden Sie im Abschnitt **XML Schlüssel der Parameter**.
- **Client Anwendung erstellen:** Mit dieser Option richten Sie den Client-Teil Ihrer Anwendung ein. Sie müssen den Speicherort Ihres *4D Volume Desktop* auf der Festplatte angeben. Es muss zur aktuellen Plattform passen, welche auch die Plattform der Client Anwendung sein wird. Wollen Sie eine Client Anwendung für eine "konkurrierende" Plattform erstellen, müssen Sie dieselbe Operation mit der 4D Version für diese Plattform ausführen. Das ist nur für die erste Version der Client Anwendung erforderlich, da sich nachfolgende Updates auf derselben Plattform über den automatischen Update-Mechanismus verwalten lassen. Weitere Informationen dazu finden Sie im nächsten Absatz. Um den Ordner 4D Volume Desktop auszuwählen, klicken Sie auf die Schaltfläche [...], und markieren im sich öffnenden Dialogfenster das passende Programm. Auf Mac OS müssen Sie direkt das 4D Volume Desktop Paket auswählen.

Hinweis: Ab 4D v15 müssen die Versionsnummer von 4D Server und 4D Volume Desktop zur Versionsnummer von 4D Developer Edition passen. Verwenden Sie z.B. 4D Developer v15.4, müssen Sie auch 4D Server v15.4 und 4D Volume Desktop v15.4 wählen.

Um die Client-Anwendung über eine spezifische Adresse an den Server anzumelden (d.h. nicht über den im Unter-Netzwerk veröffentlichten Server Namen), müssen Sie den XML Schlüssel **IPAddress** in der Datei *buildapp.xml* wählen. Weitere Informationen dazu finden Sie unter dem Befehl **BUILD APPLICATION**. Sie können auch spezifische Mechanismen einbauen, wenn eine Verbindung fehlschlägt. Die verschiedenen Szenarien werden im Abschnitt **Verbindungsprozess für eingebundene Clients** beschrieben.

4D Server bzw. 4D Client anpassen

Beim Einrichten der ausführbaren Client/Server-Anwendung dupliziert 4D den Inhalt des Ordners *4D Server* in den Server Unterordner des Zielordners und den Inhalt des Ordners *4D Volume Desktop* in den Client Unterordner des Zielordners. Sie können dann den Inhalt der Originalordner *4D Server* und *4D Volume Desktop* komplett anpassen.

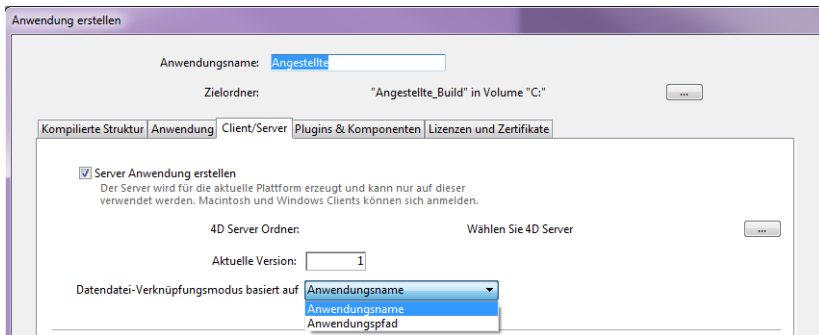
Sie können z.B. folgendes ausführen:

- Eine Version von 4D Server für eine bestimmte Sprache installieren
- Im Ordner Plugins Dateien oder Ordner hinzufügen
- Den Inhalt des Ordners 4D Extensions anpassen

Hinweis: Auf Mac OS wird 4D Server als Paket geliefert. Sie müssen zuerst seinen Inhalt anzeigen (**Ctrl + Klick** auf seinen Icon), um ihn ändern zu können.

Datendatei-Verknüpfungsmodus

Damit wählen Sie den Verknüpfungsmodus zwischen der doppelklickbaren Anwendung und der lokalen Datendatei. Es gibt zwei Arten mit jeweiligen Vor- und Nachteilen. Wählen Sie die Art, die am besten zu Ihren Anforderungen passt:



Hinweis: Diese Option ist nur verfügbar, wenn die Option **Benutze neue Architektur für Anwendungen-Weitergabe** markiert ist (siehe [Seite Kompatibilität](#)).

- Über **Anwendungsname** (Standard)
Damit öffnet die 4D Anwendung automatisch die zuletzt geöffnete Datendatei, die zur Strukturdatei gehört. Mit diesem flexiblen und intuitiven Modus können Sie das Anwendungspaket frei auf der Festplatte bewegen. Das ist die reguläre Nutzung für doppelklickbare Anwendungen, außer Sie müssen Ihre Anwendung duplizieren.
- Über **Anwendungspfad**
Damit analysiert die doppelklickbare 4D Anwendung die Datei *lastDataPath.xml* der Anwendung und versucht, die Datendatei mit einem Attribut "executablePath" zu öffnen, die zum vollständigen Pfad der Anwendung passt. Wird solch eine Eingabe gefunden, wird die dazugehörige Datendatei geöffnet (definiert durch ihr Attribut "dataFilePath"). Andernfalls wird die zuletzt geöffnete Datendatei geöffnet (Standardmodus).
Mit dieser Option können Sie Ihre doppelklickbaren Anwendungen duplizieren, ohne die Verbindung zur Datendatei zu unterbrechen (siehe oben).
Wird dagegen das Anwendungspaket auf der Festplatte verschoben, wird der Benutzer nach der Datendatei gefragt, da der Pfad der Anwendung nicht mehr mit dem Attribut "executablePath" übereinstimmt (hat ein Benutzer eine Datendatei gewählt, wird die Datei *lastDataPath.xml* entsprechend aktualisiert).

Weitere Informationen dazu finden Sie im Abschnitt [Datendatei-Verknüpfungsmodus wählen](#).

Kopie der Client Anwendung innerhalb der Server Anwendung

Über die Optionen in diesem Bereich lassen sich die 4D Volume Desktop-Teile Ihrer Client/Server Anwendungen bei einer neuen Version automatisch über das Netzwerk aktualisieren.

- **Automatisches Update für Windows / Mac OS Clients erlauben:** Damit aktivieren Sie für Ihre eigenen Client/Server Anwendungen das automatische Update der Client-Rechner über das Netzwerk.
Wollen Sie automatische Updates für eine multiplattformfähige Client Anwendung erstellen, müssen Sie den Speicherort von *4D Volume Desktop* mit der Client Anwendung (Paket auf Mac OS) der anderen Plattform angeben. Wollen Sie Ihre Anwendung z.B. unter Windows erstellen, müssen Sie über die Schaltfläche [...] den Ordner von *4D Volume Desktop* für Mac OS angeben, der auf Ihren Windows-Rechner kopiert wurde.

Ein Update der Client Anwendung ist notwendig, wenn Sie eine neue Version von 4D Server oder eines Teils der erstellten Anwendung verwenden wollen. Die Änderungen an der angepassten Fassung, sei es an der Struktur oder an den Daten, werden immer automatisch an jeden Client-Rechner übertragen.

Diese Funktionsweise erspart dem Entwickler, auf jedem Client-Rechner manuell neue Versionen von *4D Volume Desktop* zu installieren.

Das automatische Ablaufen des Update für 4D Client Anwendungen wird über HTTP ausgeführt und verwendet das Hilfsprogramm "Updater", das auf Client Rechnern installiert ist und zum Verwalten von Updates geladen wird.

Update Meldung anzeigen

Die Meldung über das Update der Client Anwendung wird automatisch nach dem Update der Server Anwendung ausgeführt.

Es funktioniert wie folgt: Wird eine neue Version der Client/Server Anwendung mit dem Application Builder erstellt, wird der neue Client-Teil als komprimierte Datei in den Unterordner *Upgrade4DClient* des Ordners *Anwendungsname_Server* kopiert. Auf Mac OS sind diese Ordner im Server Paket enthalten. Haben Sie die Anweisungen zum Erstellen einer crossplattform Client Anwendung befolgt, ist für jede Plattform eine Update-Datei vorhanden:

Name	Änderungsdatum	Typ	Größe
archive.mac	07.07.2014 14:42	MAC-Datei	60.683 KB
archive.win	07.07.2014 14:41	WIN-Datei	53.432 KB
info	07.07.2014 14:42	WAKANDA.json	1 KB
info	07.07.2014 14:42	XML-Datei	1 KB
update.mac.4darchive	07.07.2014 14:41	4DARCHIVE-Datei	60.679 KB
update.win.4darchive	07.07.2014 14:40	4DARCHIVE-Datei	53.430 KB

Hinweis: Die zusätzlichen Dateien *archive.mac* und *archive.win* werden zum Update von v13 Anwendungen auf v14 verwendet. Sie können das automatische Erstellen dieser Dateien über den XML Key **BuildV13ClientUpgrades** mit dem Befehl **BUILD APPLICATION** deaktivieren.

Um die Meldung für ein Update der Client Anwendung auszulösen, ersetzen Sie einfach die alte Version der Server Anwendung mit der neuen und führen sie dann aus. Der weitere Ablauf ist automatisch.

Auf der Client-Seite erscheint eine Meldung, wenn die „alte“ Client Anwendung versucht, sich an eine aktualisierte Server Anwendung anzumelden. Der Benutzer kann dann die Meldung bestätigen, um seinen Client upzudaten oder die Aktion abbrechen.

- Klickt der Benutzer auf die Schaltfläche **OK**, wird die neue Version über das Netzwerk auf den Client-Rechner geladen. Ist das Download abgeschlossen, wird die alte Client Anwendung geschlossen, die neue Version gestartet und mit dem Server verbunden. Die alte Version wird dann in den Papierkorb des Rechners gelegt.
- Klickt der Benutzer auf die Schaltfläche **Abbrechen**, wird das Update abgebrochen; ist die alte Version nicht innerhalb der Versionen, die der Server akzeptiert, wird die Anwendung geschlossen und es ist keine Verbindung möglich. Sonst wird die Verbindung hergestellt.

Automatisches Update erzwingen

In bestimmten Fällen wollen Sie verhindern, dass Client Anwendungen das Download des Update annullieren können. Verwenden Sie z.B. eine neue Version der ursprünglichen Applikation von 4D Server, muss auf jedem Client-Rechner zwingend die neue Version der Client Anwendung installiert werden.

Um das Update zu erzwingen, schließen Sie einfach im Bereich der Versionsnummern, die mit der Server Anwendung kompatibel sind, die aktuelle Versionsnummer der Client Anwendung aus (x-1 und früher). Dann verhindert der Update-Mechanismus, dass sich nicht-upgedatete Client Anwendungen anmelden können. Ist z.B. die neue Version der Client/Server-Anwendung 6, können Sie festsetzen, dass jede Client Anwendung mit einer Versionsnummer niedriger als 6 sich nicht anmelden kann.

Die aktuelle Versionsnummer wird im Dialogfenster Anwendung erstellen auf der Seite Client/Server angegeben (siehe oben). Die Intervalle der zugelassenen Nummern werden im Anwendungsprojekt über spezifische XML Schlüssel gesetzt. Weitere Informationen dazu finden Sie im Abschnitt **XML Schlüssel der Parameter**.

Fehler beim Update

Kann 4D das Update der Client Anwendung nicht ausführen, zeigt der Client-Rechner die Meldung: „Update der Client Anwendung ist fehlgeschlagen. Die Anwendung wird nun beendet.“

Dieser Fehler kann verschiedene Ursachen haben. Wenn Sie diese Meldung erhalten, sollten Sie zuerst folgende Parameter überprüfen:

- **Pfadnamen:** Prüfen Sie die Gültigkeit der Pfadnamen, die im Anwendungsprojekt über das Dialogfenster *Anwendung erstellen* oder über XML Keys (z.B. ClientMacFolderToWin) gesetzt wurden. Prüfen Sie insbesondere die Pfadnamen der Versionen von 4D Volume Desktop.
- **Lese-/Schreibrechte:** Prüfen Sie auf dem Client-Rechner, ob der aktuelle Benutzer Schreibrechte für das Update der Client Anwendung hat.

Erzeugte Dateien

Ist die Client/Server Anwendung eingerichtet, enthält der Zielordner einen neuen Ordner mit Namen **Client Server executable**. Er hat die beiden Unterordner *AnwendungsnameClient* und *AnwendungsnameServer*:

Name	Änderungsdatum	Typ
Rechnungen Client	07.07.2014 14:40	Dateiordner
Rechnungen Server	07.07.2014 14:40	Dateiordner

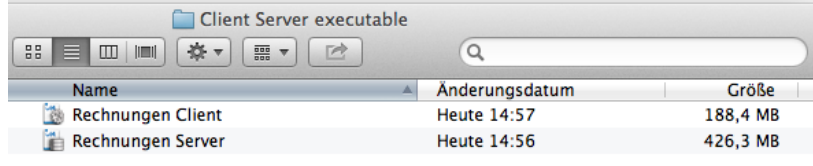
Hinweis: Diese Ordner werden nicht angelegt, wenn ein Fehler auftritt. Öffnen Sie in diesem Fall das Logbuch, um die Ursache dafür zu finden. Weitere Informationen dazu finden Sie im Abschnitt **Logbuch**.

Der Ordner **Anwendungsname Client** enthält den Client-Teil der Anwendung für die Ausführungsplattform des Application Builder. Dieser Ordner muss auf jedem Client-Rechner installiert werden. Der Ordner *Anwendungsname Server* enthält den Server-Teil der Anwendung.

Der Inhalt der Ordner variiert je nach der aktuellen Plattform:

- **Unter Windows:** Jeder Ordner enthält die Datei *Anwendungsname Client.exe* für den Client-Teil und *Anwendungsname Server.exe* für den Server-Teil, sowie die entsprechenden .rsr Dateien. Außerdem gibt es verschiedene Dateien und Ordner, die zum Betrieb der Anwendung erforderlich sind und angepasste Elemente, die in die Originalordner *4D Server* und *4D Volume Desktop* integriert werden können.
- **Auf Mac OS:** Die Ordner enthalten jeweils das Paket *Anwendungsname Client* für den Client-Teil und *Anwendungsname Server* für den Server-Teil. Jedes Paket enthält alle erforderlichen Elemente, damit die

Anwendung funktioniert. Auf Mac OS starten Sie ein Paket durch Doppelklick:



Hinweis: Die erzeugten Mac OS Pakete enthalten die gleichen Elemente wie die Unterordner unter Windows. Mit **Ctrl + Klick** auf das Icon können Sie den Inhalt anzeigen, um ihn bei Bedarf zu verändern.

Markieren Sie die Option "Automatisches Update für Windows / Mac OS Clients erlauben", wird im Ordner/Paket *Anwendungsname Server* ein Ordner **Upgrade4DClient** hinzugefügt. Dieser Unterordner enthält die Client Anwendung im Mac OS bzw. Windows Format als komprimierte Datei. Diese Datei wird beim automatischen Update der Client Anwendung verwendet.

Speicherort der Web Dateien

Wird der Server- bzw. Client-Teil Ihrer doppelklickbaren Anwendung als Web Server verwendet, müssen die vom Server erforderlichen Dateien und Ordner an spezifischen Stellen installiert werden. Das sind folgende Elemente:

- cert.pem und key.pem (optional): Diese Dateien werden für SSL Verbindungen und von Befehlen zur Datenverschlüsselung verwendet
- WebFolder: Das ist der Standard Web Root Ordner

Unter Windows

- Server Anwendung: Diese Elemente müssen im Unterordner Client Server executable\Anwendungsname Server\Server Datenbank liegen.
- Client Anwendung: Diese Elemente müssen im Unterordner Client Server executable\Anwendungsname Client liegen

Auf Mac OS

- Server Anwendung: Diese Elemente müssen neben dem Software Paket Anwendungsname Server installiert werden.
- Client Anwendung: Diese Elemente müssen neben dem Software Paket Anwendungsname Client installiert werden.

Seite Plug-Ins und Komponenten

Auf der Seite **Plugins und Komponenten** des Application Builder legen Sie fest, welche Plug-Ins und Komponenten Sie in Ihrer Einzelplatz- oder Client/Server Anwendung verwenden wollen. Weitere Informationen dazu finden Sie im Abschnitt **Plug-Ins oder Komponenten installieren**.

Die Seite zeigt die Elemente an, die vom aktuellen 4D Programm geladen wurden:

Aktiv	Plugins und Komponenten	ID	Typ
<input checked="" type="checkbox"/>	4D SVG	0	Component
<input checked="" type="checkbox"/>	4D Widgets	0	Component

Die Spalte **Typ** zeigt die Art des Eintrags: Plug-In oder Komponente.

Die Spalte **Aktiv** gibt an, dass die Einträge in die erstellte Anwendung integriert werden. Standardmäßig sind alle Einträge markiert. Um ein Plug-In bzw. eine Komponente auszuschließen, deaktivieren Sie das dazugehörige Kästchen in der Spalte **Aktiv**.

Wollen Sie andere Plug-Ins in die ausführbare Anwendung integrieren, müssen Sie diese lediglich in einen Ordner *PlugIns* oder *Components* neben dem Programm *4D Volume Desktop* bzw. *4D Server* legen. Sie können zum Integrieren von Dateien in eine ausführbare Anwendung genauso vorgehen wie beim Kopieren des Inhalts des Quellordners des Programms. Weitere Informationen dazu finden Sie im Abschnitt **Ordner 4D Volume Desktop anpassen**.

- Bei einem Konflikt zwischen zwei verschiedenen Versionen des gleichen Plug-In (eine Version wird von 4D geladen, die andere liegt im Quellordner der Anwendung), hat das Plug-In Priorität, das im Ordner *4D Volume Desktop/4D Server* installiert wurde.
- Gibt es dagegen zwei Instanzen derselben Komponente, wird das Programm nicht geöffnet.

Hinweis: Zum Verwenden von Plug-Ins bzw. Komponenten in einer Deployment-Version sind die entsprechenden Lizenznummern erforderlich.

Auf der Seite **Lizenzen und Zertifikate** führen Sie folgendes aus:

- Sie geben die Lizenznummer(n) an, die Sie in Ihre eigenständige Einzelplatzversion integrieren wollen
- Sie signieren die Anwendung über ein Zertifikat auf OS X.

Lizenzen

Diese Seite listet die verfügbaren Deployment Lizenzen auf, die Sie in Ihre Anwendung integrieren können. Die Liste ist standardmäßig leer. Sie müssen Ihre Lizenz für *4D Developer Professional* sowie jede Lizenz für *4D Desktop Volume*, die in der erstellten Anwendung verwendet wird, explizit hinzufügen. Sie können auch eine andere Lizenznummer als die aktuell verwendete hinzufügen.

Über die Schaltflächen [+] und [-] unter der Liste können Sie eine Lizenz entfernen bzw. hinzufügen.

Klicken Sie auf die Schaltfläche [+], erscheint ein Öffnen-Dialog, der standardmäßig den Inhalt des Ordners **Licenses** auf Ihrem Rechner anzeigt. Weitere Informationen zum Speicherort dieses Ordners finden Sie im Handbuch *4D Programmiersprache* in den Abschnitten **Ordner Licenses** und **Get 4D folder**.

Sie müssen die Dateien angeben, welche Ihre Developer Lizenz und welche Ihre Deployment-Lizenzen enthalten. Diese Dateien wurden beim Erwerb der Lizenzen für *4D Developer Professional* und *4D Desktop Volume* angelegt oder aktualisiert.

Haben Sie eine Datei ausgewählt, zeigt die Liste die Merkmale der enthaltenen Lizenz:

Kompilierte Struktur Anwendung Client/Server Plugins & Komponenten Lizenzen und Zertifikate

Lizenz #	Lizenz	Ablaufdatum	Pfad
PNDP150UUC...	4D Developer Professio...	01.02.2016	C:\ProgramData\4D\Licenses\R-PNDP150UUC00\license4C ^
PNUD150UUC...	4D Unlimited Desktop ...		C:\ProgramData\4D\Licenses\R-PNUD150UUC00\license4I

- **Lizenz #:** Lizenznummer des Produkts
- **Lizenz:** Name des Produkts
- **Ablaufdatum:** Ablaufdatum der Lizenz (falls vorhanden)
- **Pfad:** Speicherort auf der Festplatte

Bei einer ungültigen Lizenz erhalten Sie eine Meldung.

Sie können beliebig viele gültige Dateien festlegen. 4D wählt beim Erstellen einer ausführbaren Anwendung unter den Lizenzen diejenige aus, die am besten geeignet ist.

Hinweis: Zum Erstellen von Anwendungen, die auf einer R-Release Version basieren, benötigen Sie spezifische "R" Lizenzen. Lizenznummern für "R" Versionen beginnen mit **R-4DDP**.

Ist die Anwendung eingerichtet, wird im Ordner **Licenses** automatisch eine neue Deployment Lizenzdatei integriert, unter Windows neben der ausführbaren Anwendung, auf Mac OS im Software-Paket.

Anwendungen auf OS X zertifizieren

Der Application Builder kann doppelklickbare 4D Anwendungen auf OS X zertifizieren (Einzelplatz Anwendungen, 4D Server und Client Teile auf OS X). Eine signierte Anwendung autorisiert die Ausführung über die Funktionalität Gatekeeper von OS X, wenn die Option "Mac App Store und verifizierte Entwickler" markiert ist (siehe unten "Über Gatekeeper").

- **Signiere Anwendung:** Markieren Sie diese Option, um bei der Application Builder Prozedur für OS X die Zertifizierung miteinzubinden. 4D prüft beim Erstellen, ob die erforderlichen Elemente für die Zertifizierung verfügbar sind:

OS X Signierungs-Zertifikat

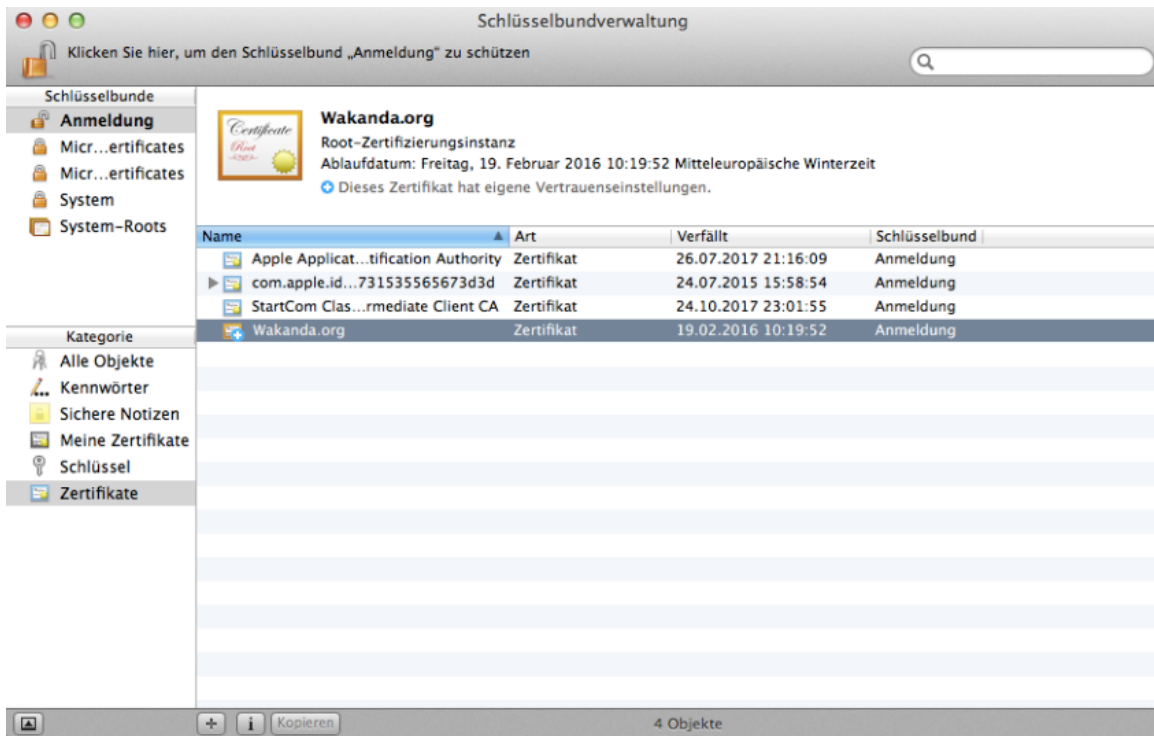
Signiere Anwendung

Apple erwartet das Anwendungen, die aus dem Internet geladen wurden, vom Entwickler signiert werden.

Name des Zertifikats:

Diese Option erscheint unter Windows und auf OS X, wird aber nur für die OS X Versionen berücksichtigt.

- **Name des Zertifikats:** Geben Sie im Eingabebereich den Namen Ihres Entwicklerzertifikats ein, das von Apple bestätigt wurde. Das ist in der Regel der Name des Zertifikats in der Schlüsselbundverwaltung:



Um ein Entwicklerzertifikat von Apple, Inc. zu erhalten, können Sie die Befehle des Menüs **Schlüsselbund Zugriff** in OS X verwenden oder folgenden Link öffnen: <http://developer.apple.com/library/mac/#documentation/Security/Conceptual/CodeSigningGuide/Procedures/Procedures.html>

Hinweis: Für Zertifikate muss das Apple Hilfsprogramm codesign verfügbar sein. Es wird standardmäßig mitgeliefert und liegt im Ordner "/usr/bin/". Tritt ein Fehler auf, überprüfen Sie, ob dieses Hilfsprogramm auf Ihrer Festplatte vorhanden ist.

Über Gatekeeper

Gatekeeper ist ein Sicherheits-Feature von OS X, um die Ausführung von Anwendungen, die über das Internet geladen werden, zu steuern. Die Option "Mac App Store und verifizierte Entwickler" ist ab OS X.8 Mountain Lion standardmäßig ausgewählt. Apple empfiehlt nicht die unterste Ebene "Keine Einschränkungen". Kommt eine geladene Anwendung nicht aus dem Apple Store oder ist sie nicht signiert, wird sie abgewiesen und lässt sich nicht starten.



Die Option **Signiere Anwendung** des 4D Application Builder ermöglicht, Anwendungen zu generieren, die standardmäßig mit dieser Option kompatibel sind.

Die Spezifikationen zu *Gatekeeper* entwickeln sich mit jeder Version von OS X weiter. Insbesondere die codesign Signaturen sind entweder vom Typ "v1" und/oder "v2", abhängig vom dem Betriebssystem, auf dem sie generiert wurden. *Gatekeeper* akzeptiert "v1" oder "v2" Signaturen, abhängig vom Betriebssystem, auf dem die Endanwendung ausgeführt wird. Nachfolgende Tabelle gibt einen Überblick:

OS X Version	von codesign generierte Signatur	von Gatekeeper akzeptiert
10.8.x und früher	v1	nur Anwendungen signiert v1
10.9.0 bis 10.9.4	v1 und v2	Anwendungen signiert v1 oder v2
10.9.5 und höher	v1 und v2	nur Anwendungen signiert v2

Standard Datendatei in doppelklickbaren Anwendungen verwalten

Datendatei öffnen

Startet ein Benutzer eine doppelklickbare Anwendung oder ein Update (Einzelplatz oder Client/Server), versucht 4D, eine gültige Datendatei auszuwählen, d.h. es werden mehrere Speicherorte nacheinander geprüft.

Die Reihenfolge zum Starten einer doppelklickbaren Anwendung ist:

1. 4D versucht, die zuletzt verwendete Datendatei zu öffnen wie unten beschrieben (gilt nicht beim ersten Starten).
2. Wird sie nicht gefunden, versucht 4D die Datendatei im Ordner **default data** neben der .4DC Datei im Nur-Lesen Modus zu öffnen (neu in 4D v15, wird unten beschrieben)
3. Wird sie nicht gefunden, versucht 4D die standardmäßige **Default** Datendatei zu öffnen (gleicher Name und gleiche Stelle wie die Datei .4DC).
4. Wird sie nicht gefunden, zeigt 4D das standardmäßige Dialogfenster "Öffne Datendatei".

Zuletzt geöffnete Datendatei

Position für den Pfad der letzten Datendatei

Ist die Option **Benutze neue Architektur für Anwendungen-Weitergabe** auf der Seite Kompatibilität markiert (siehe [Seite Kompatibilität](#)), wird bei doppelklickbaren 4D Anwendungen im Einzelplatz oder im Client/Server-Betrieb die zuletzt geöffnete Datendatei im Ordner *Benutzereinstellungen* der Anwendung abgelegt.

Hinweis zur Kompatibilität: In bisherigen 4D Versionen wurde diese Information in der Strukturdatei gespeichert.

Die Position des Ordners *Benutzereinstellungen* der Anwendung entspricht dem Pfad, den folgende Anweisung zurückgibt:

```
userPrefs:=Get 4D folder(Active 4D Folder)
```

Der Pfad zur Datendatei wird in einer eigenen Datei mit Namen *lastDataPath.xml* gespeichert.

Mit dieser neuen Architektur wird bei Liefern einer neuen Version Ihrer Anwendung die Datendatei des lokalen Nutzers (zuletzt verwendete Datendatei) automatisch beim nächsten Start geöffnet.

Dieser Mechanismus passt in der Regel bei standardmäßiger Weitergabe von Anwendungen. Für spezifische Anforderungen, wie z.B. beim Duplizieren Ihrer doppelklickbaren Anwendungen, müssen Sie evtl. die Art, wie die Datendatei mit der Anwendung verbunden ist, verändern. Weitere Informationen dazu finden Sie im nächsten Abschnitt.

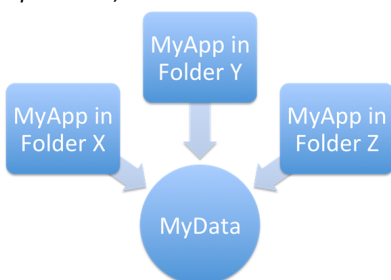
Datendatei-Verknüpfungsmodus wählen

4D verwendet in Ihren kompilierten Anwendungen automatisch die zuletzt geöffnete Datendatei. Standardmäßig, d.h. wenn die neue Architektur aktiviert ist (ab 4D v15 R4, siehe Abschnitt oben), wird der Pfad der Datendatei im Ordner *Benutzereinstellungen* gesichert und mit dem **Anwendungsnamen** verknüpft.

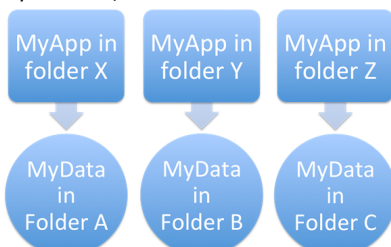
Dieses Vorgehen ist ungeeignet, wenn Sie eine doppelklickbare Anwendung duplizieren wollen, die mit unterschiedlichen Datendateien arbeiten soll. Duplizierte Pakete nutzen den Ordner *Benutzereinstellungen* gemeinsam und verwenden so immer die gleiche Datendatei -- selbst bei Umbenennung, da die zuletzt verwendete Datendatei der Anwendung geöffnet wird.

Deshalb ermöglicht 4D, den Pfad der Datendatei mit dem **Anwendungspfad** zu verbinden. In diesem Fall wird die Datendatei über einen spezifischen Pfad verbunden und es ist nicht lediglich die zuletzt geöffnete Datendatei.

Duplikation, wenn Daten über Anwendungsname verbunden sind:



Duplikation, wenn Daten über Anwendungspfad verbunden sind:



Sie können beim Erstellen der Anwendung einen Datendatei-Verknüpfungsmodus wählen, und zwar:

- im Dialogfenster "Anwendung erstellen" auf der [Seite Anwendung](#) oder [Seite Client/Server](#).
- über den XML Schlüssel **LastDataPathLookup** (Einzelplatzanwendung) oder **LastDataPathLookup** (Serveranwendung).

Standard Datenordner definieren

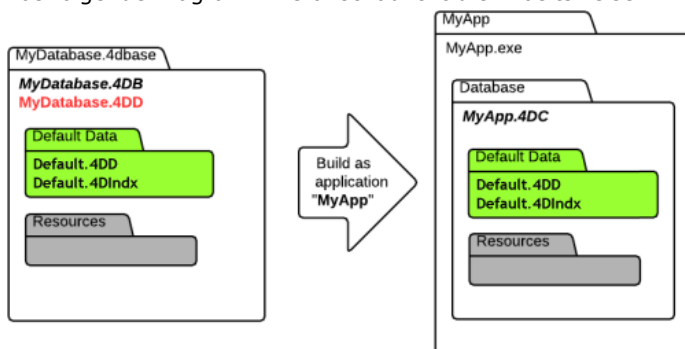
4D erlaubt, beim Aufbau einer Anwendung eine Standard Datendatei zu definieren. Wird beim ersten Starten der doppelklickbare Anwendung keine lokale Datendatei gefunden, öffnet 4D automatisch die Standard Datendatei im Nur-Lesen Modus. Auf diese Weise haben Entwickler eine bessere Kontrolle beim Erstellen oder Öffnen der Datendatei, wenn eine doppelklickbare Anwendung zum ersten Mal geöffnet wird. Das ist besonders in folgenden Fällen von Vorteil:

- Beim Starten einer neuen oder aktualisierten doppelklickbaren Anwendung wird nicht das 4D Dialogfenster "Öffne Datendatei" angezeigt. Sie können beispielsweise in der **Datenbankmethode On Startup** feststellen, dass die Standard Datendatei geöffnet wurde und so Ihren eigenen Code ausführen bzw. bestimmte Dialogfenster öffnen, um eine lokale Datendatei zu erstellen oder auszuwählen.
- Doppelklickbare Anwendungen mit Daten im Nur-Lesen Modus ausliefern, z.B. für Demo-Anwendungen.

Um eine Standarddatei zu definieren:

- Legen Sie eine Standard Datendatei mit Namen "Default.4DD an" und speichern diese in einem Ordner mit Namen "Default Data" innerhalb des Datenbank Package (4dbase). Diese Standard Datendatei muss alle erforderlichen Dateien enthalten: Index, Journal, externe BLOBS, etc. Der Entwickler muss selbst dafür sorgen, dass die Standard Datendateien funktionsfähig sind. Da die Standard Datendatei jedoch im Nur-Lesen Modus geöffnet wird, ist es wichtig, vor dem Erstellen der Datendatei die Option "Benutze Logbuch" in der Original Strukturdatei zu deaktivieren.
- Beim Erstellen der Anwendung wird dieser Standard Datenordner in die doppelklickbare Anwendung integriert. Alle Dateien mit dem Standard Dateiordner werden ebenfalls eingebunden.

Das folgende Diagramm veranschaulicht die Arbeitsweise:



Wird beim ersten Starten ein Standard Datenordner gefunden, wird dieser automatisch im Nur-Lesen Modus geöffnet. Das ermöglicht Ihnen, beliebige eigene Operationen auszuführen, die die Datendatei selbst nicht ändern.

Verbindungsprozess für eingebundene Clients

In 4D v15 R4 wurde der client-seitige Prozess geändert, der die Verbindungen zum Server verwaltet, um dem Entwickler mehr Kontrolle und größere Flexibilität bei Verbindungsfehlern zu bieten.

Kompatibilität

Die hier beschriebenen Mechanismen sind ab 4D v15 R4 verfügbar. Für konvertierte Anwendungen aus älteren Anwendungen müssen Sie explizit die Option **Benutze neue Architektur für Anwendungen-Weitergabe** auf der Seite "Kompatibilität" in den Datenbank-Eigenschaften markieren (siehe [Seite Kompatibilität](#)).

Ablauf beim Anmelden

Bei Anwendungen mit eingebundenen Clients gibt es verschiedene Möglichkeiten, wenn der dedizierte Server nicht verfügbar ist. Für die 4D Client-Anwendung läuft jetzt folgendes Szenario ab:

- Die Client-Anwendung versucht, sich über den Suchdienst (discovery service) an den Server anzumelden (basiert auf dem Server Namen, der im gleichen Unternetz verbreitet wird).
ODER
Ist in der Datei *EnginedServer.4DLink* innerhalb der Client-Anwendung eine gültige Verbindungsinformation gespeichert, versucht die Client-Anwendung, sich an die angegebene Server Adresse anzumelden.
Hinweis zur Kompatibilität: Ist die Option **Benutze neue Architektur für Anwendungen-Weitergabe** auf der Seite Kompatibilität nicht markiert und schlägt die Verbindung fehl, erscheint direkt das standardmäßige Dialogfenster zum Anmelden an den Server.
- Schlägt das fehl, versucht die Client-Anwendung, sich über die Angaben an den Server anzumelden, die im Ordner *Benutzereinstellungen* gespeichert sind (Datei *lastServer.xml*, siehe letzter Schritt).
- Schlägt das auch fehl, zeigt die Client-Anwendung einen Verbindungsfehler.
 - Klickt der Benutzer auf die Schaltfläche **Auswählen**, zeigt die Client-Anwendung, falls vom 4D Entwickler beim Einrichten erlaubt (siehe unten), den Standarddialog für die Server Verbindung.
 - Klickt der Benutzer auf die Schaltfläche **Beenden**, beendet die Client-Anwendung.
- Bei erfolgreicher Anmeldung speichert die Client-Anwendung die aktuelle Verbindungsinformation in ihrem Ordner *Benutzereinstellungen* zur späteren Wiederverwendung.

Den letzten Serverpfad speichern

Der zuletzt benutzte und bestätigte Serverpfad wird automatisch in einer Datei mit Namen *lastServer.xml* im Ordner *Benutzereinstellungen* gesichert. Dieser Ordner wird an folgender Stelle gespeichert:

```
userPrefs:=Get 4D folder(Active 4D Folder)
```

Hinweis zur Kompatibilität: Ist die Option **Benutze neue Architektur für Anwendungen-Weitergabe** auf der Seite **Kompatibilität** nicht markiert, wird der Pfad nicht gesichert.

Dieser Mechanismus deckt den Fall ab, dass der primäre Zielsever temporär nicht verfügbar ist, z.B. im Wartungsmodus. Beim ersten Auftreten wird der Server Auswahldialog angezeigt (falls erlaubt, siehe unten) und der Benutzer kann manuell einen alternativen Server auswählen, dessen Pfad bei erfolgreicher Anmeldung dann gesichert wird. Jede nachfolgende Nicht-Verfügbarkeit wird dann automatisch über die Pfadinformation in *lastServer.xml* verwaltet.

Hinweise:

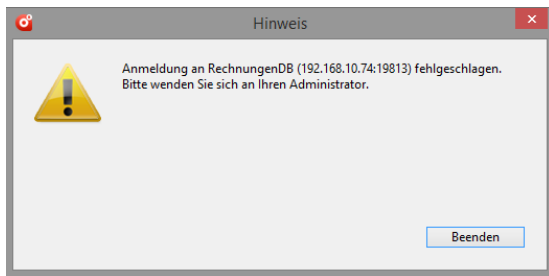
- Können Client-Anwendungen den Suchdienst nicht permanent nutzen, z.B. wegen der Netzwerk-Konfiguration, sollte der Entwickler weiterhin beim Aufbau der Anwendung über den Schlüssel **IPAddress** in der Datei *BuildApp.xml* einen Host-Namen liefern. Dieser neue Mechanismus hilft bei temporärer Nicht-Verfügbarkeit.
- Die Tastenkombination **Alt-/Wahltaste** beim Starten unterstützt weiterhin in allen Fällen die Anzeige des Dialogfensters zum Auswählen des Server.

Verfügbarkeit des Server Auswahldialog, wenn ein Fehler auftritt

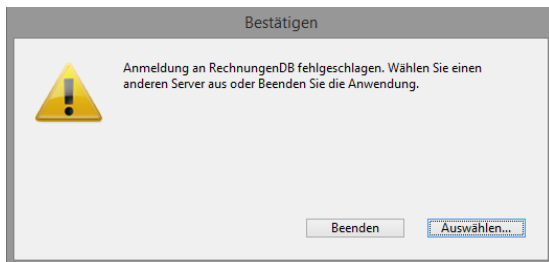
Ab 4D v15 R4 lässt es sich einrichten, dass der standardmäßige Auswahldialog für den Server nicht mehr automatisch auf den eingebundenen Client Rechnern erscheint, wenn der Server nicht erreichbar ist. Das hängt jetzt von der Option **Benutze neue Architektur für Anwendungen-Weitergabe** (siehe Abschnitt **Kompatibilität**) und dem Wert des XML Schlüssels **ServerSelectionAllowed** ab.

Es gibt drei Möglichkeiten:

- **Fehlermeldung anzeigen ohne Möglichkeit, auf den Auswahldialog für den Server zuzugreifen**
Standardeinstellung für Anwendungen, die ab 4D v15 R4 erstellt wurden. Die Applikation kann nur beenden. Diese Funktionsweise erhalten Sie mit folgender Konfiguration:
 - Option **Benutze neue Architektur für Anwendungen-Weitergabe** ist markiert
 - XML Schlüssel *ServerSelectionAllowed* wird weggelassen oder hat den Wert **Falsch**



- **Fehlermeldung anzeigen mit Möglichkeit, auf den Auswahldialog für den Server zuzugreifen**
 Der Benutzer kann über die Schaltfläche **Auswählen** auf den Auswahldialog für den Server zugreifen. Diese Funktionsweise erhalten Sie mit folgender Konfiguration:
 - Option **Benutze neue Architektur für Anwendungen-Weitergabe** ist markiert
 - XML Schlüssel *ServerSelectionAllowed* hat den Wert **Wahr**



=>



- **Den Auswahldialog für den Server direkt anzeigen**
 Standardeinstellung für konvertierte Anwendungen. Das ist die Funktionsweise bisheriger 4D Versionen. Sie erhalten diese mit folgender Konfiguration:
 - Option **Benutze neue Architektur für Anwendungen-Weitergabe** ist nicht markiert
 - XML Schlüssel *ServerSelectionAllowed* wird ignoriert



Hinweis: Weitere Informationen zum XML Schlüssel **ServerSelectionAllowed** finden Sie im Handbuch **4D XML Keys BuildApplication**.

Automatisches Updaten von Server oder Einzelplatzanwendungen

Bisher war zum Aktualisieren von Server Anwendungen oder doppelklickbaren Einzelplatz-Anwendungen eine Aktion des Benutzers oder Programmieren eigener System-Routinen erforderlich: War eine neue Version der doppelklickbaren Anwendung verfügbar, mussten Sie die laufende Anwendung verlassen und die alten Dateien manuell durch die neuen ersetzen; dann die Anwendung neu starten und die aktuelle Datendatei auswählen.

Sie können dieses Vorgehen über die Befehle **SET UPDATE FOLDER, RESTART 4D** und die Funktion **Get last update log path** zum Überwachen der Operationen weitgehend automatisieren. Dahinter steckt die Idee, in Ihre Anwendung eine Funktion einzubauen, die das automatische Ablaufen des Update auslöst. Das können ein Menübefehl oder ein Prozess im Hintergrund sein, die in regelmäßigen Abständen prüfen, ob eine Datei auf dem FTP Server vorhanden ist.

Hier sehen Sie das Szenario zum Updaten einer Server- oder einer doppelklickbaren Einzelplatz-Anwendung:

1. Sie übertragen, z.B. über einen FTP Server, die neue Version der Server Anwendung oder der doppelklickbaren Einzelplatz Anwendung auf den Rechner in Produktion.
2. Sie rufen in der laufenden Anwendung den Befehl **SET UPDATE FOLDER** auf. Er gibt den Speicherort des Ordners an, in dem das anstehende Update für die aktuelle Anwendung liegt. Optional können Sie in diesen Ordner die eigenen Elemente der laufenden Version (Benutzerdateien) kopieren.
3. Sie rufen in der laufenden Anwendung den Befehl **RESTART 4D** auf: Er löst die Ausführung eines Hilfsprogramms mit Namen "Updater" aus. Es schließt die aktuelle Anwendung, ersetzt sie, sofern vorhanden, mit dem anstehenden Update und startet die Anwendung mit der aktuellen Datendatei neu. Die frühere Version wird umbenannt.

Hinweise:

- Dies ist kompatibel mit Server Anwendungen unter Windows, die als Service laufen (siehe **Datenbank als Dienst registrieren**).
- Es gibt auch neue XML Schlüssel, um Installationsrechte zu erweitern, so dass Sie unter Windows geschützte Dateien verwenden können. Weitere Informationen dazu finden Sie im Handbuch **4D XML Keys BuildApplication**.

Protokoll zum Update

Der Installationsprozess erstellt jetzt ein Protokoll, das die Update Operationen für doppelklickbare Anwendungen (Client, Server oder Einzelplatz) auf den Zielrechnern auflistet. Es ist hilfreich, um Fehler zu analysieren, die während der Installation auftreten.

Das Protokoll hat den Namen **YYYY-MM-DD_HH-MM-SS_log_<sequence>.txt**, zum Beispiel **2013-08-25_14-23-00_log_1.txt** für ein Protokoll, erstellt am 25. August 2013 um 14.23.

Diese Datei wird im Application Ordner von "Updater" erstellt. z.B.:

- auf OS X:
`{userName}/Library/Application Support/{ProductName}/4D/Updater/`
- unter Windows:
`\{userName}\AppData\Roaming\{ProductName}\4D\Updater\`

Mit der Funktion **Get last update log path** können Sie jederzeit den Speicherort dieser Datei herausfinden.

Icon der eigenständigen Anwendung anpassen

4D weist doppelklickbaren Anwendungen ein Standard-Icon zu. Sie können es aber auch für jede Anwendung individuell gestalten.

Auf Mac OS

Beim Anlegen einer doppelklickbaren Anwendung übernimmt 4D die Anpassung der Icons.

Dazu müssen Sie vor Erzeugen der Anwendung eine Datei **Icon** mit der Endung **.icns** erstellen und diese auf dieselbe Ebene wie die interpretierte Strukturdatei legen.

Hinweis: Apple, Inc. liefert ein spezielles Tool zum Erstellen von **.icns** Dateien. Weitere Informationen dazu finden Sie in der [Apple documentation](#).

Ihre Icon-Datei muss denselben Namen wie die interpretierte Strukturdatei und die Endung **.icns** haben. 4D berücksichtigt diese Datei automatisch beim Erstellen der doppelklickbaren Anwendung. Die Datei wird umbenannt in *AnwendungName.icns* und in den Ordner *Resources* kopiert; Die Eingabe *CFBundleFileIcon* in der Datei "info.plist" wird aktualisiert

Unter Windows





Beim Erzeugen einer doppelklickbaren Anwendung übernimmt 4D die Anpassung der Icons.

Dazu müssen Sie vor Erzeugen der Anwendung eine Datei **Icon** mit der Endung **.ico** erstellen und diese auf dieselbe Ebene wie die interpretierte Strukturdatei legen.

Diese Datei muss denselben Namen wie die interpretierte Strukturdatei und die Endung **.ico** haben.

4D berücksichtigt diese Datei automatisch beim Erstellen der doppelklickbaren Anwendung.

4D Komponenten entwickeln und installieren

-  Überblick
-  Komponente installieren
-  Komponente entwickeln
-  Interaktion zwischen Komponenten und Host Datenbanken

Eine 4D Komponente besteht aus einer Reihe von 4D Objekten mit einer oder mehreren Funktionalitäten, die sich in verschiedenen Datenbanken installieren lassen. Sie können z.B. eine 4D E-Mail Komponente entwickeln, die alle Aspekte von Versenden, Empfangen und Speichern von E-Mails in 4D Datenbanken verwaltet.

4D Komponenten werden direkt aus 4D erstellt und installiert. Komponenten werden ähnlich wie Plug-Ins verwaltet. Es gelten folgende Regeln:

- Eine Komponente besteht aus einer gängigen Strukturdatei (kompiliert oder nicht) mit der Standardarchitektur oder in Form eines Pakets. Weitere Informationen dazu finden Sie im Handbuch *4D Programmiersprache* im Abschnitt **Ordner .4dbase**).
- Um eine Komponente in einer Datenbank zu installieren, genügt es, diese in den Ordner **Components** der Datenbank zu kopieren, der neben der Strukturdatei oder dem ausführbaren 4D Programm liegt. Sie können sogar eine Verknüpfung (Windows) oder ein Alias (Mac OS) benutzen. Um sie zu deinstallieren, entfernen Sie die Komponente einfach aus dem Ordner.
- In 4D Komponenten können Sie weder Standardtabellen noch Datendateien verwenden. Eine Komponente kann jedoch Tabellen, Felder und Datendateien über Operationen externer Datenbanken erstellen bzw. verwenden. Dies sind separate 4D Datenbanken, in denen Sie mit SQL Befehlen arbeiten. Weitere Informationen dazu finden Sie im Handbuch *4D SQL Reference* im Abschnitt **About external databases**

Zum Erstellen von Komponenten ist eine entsprechende Lizenz notwendig. Zum Installieren und Einsetzen in einer 4D Anwendung wird dagegen keine Lizenz benötigt.

Definitionen

Zur Handhabung von Komponenten in 4D sind folgende Begriffe und Konzepte von Bedeutung:

- **Matrix Datenbank:** 4D Datenbank zum Entwickeln der Komponente. Die Matrix Datenbank ist eine Standarddatenbank ohne spezifische Attribute. Sie dient zum Erstellen einer einzelnen Komponente. Die Matrix Datenbank wird kompiliert oder unkompiliert in den Ordner **Components** der Datenbank kopiert, welche die Komponente verwendet. Das ist die Host Datenbank.
- **Host Datenbank:** Datenbank, in der eine Komponente installiert und verwendet wird.
- **Komponente:** Matrix Datenbank, kompiliert oder nicht, die in den Ordner **Components** der Host Datenbank kopiert wird und deren Inhalt in der Host Datenbank verwendet wird.
- **Projektformular:** Formular, das keiner Tabelle zugeordnet ist. Projektformulare sind besonders zum Erstellen von Komponenten geeignet. Weitere Informationen dazu finden Sie im Abschnitt **Tabellenformulare und Projektformulare**.
- **Tabellenformular** (auch Standardformular genannt): Einer Tabelle zugewiesenes Formular. Dieser Formulartyp lässt sich in einer Komponente nicht verwenden.

Beachten Sie, dass eine Datenbank sowohl vom Typ "matrix" als auch "host" sein kann, d.h. eine Matrix Datenbank kann selbst eine oder mehrere Komponenten verwenden. Eine Komponente kann dagegen selbst keine untergeordneten Komponenten verwenden.

Komponenten durch Kompilieren schützen

Standardmäßig sind alle Projektmethoden einer Matrix Datenbank, die als Komponente installiert ist, von der Host Datenbank aus potentiell sichtbar. Das bedeutet im einzelnen:

- Gemeinsam verwendete Projektmethoden erscheinen im Explorer auf der **Seite Methoden**. Sie lassen sich in den Methoden der Host Datenbank aufrufen. Ihr Inhalt lässt sich auswählen und aus der Vorschau des Explorers kopieren. Sie sind auch im Debugger sichtbar. Sie lassen sich dagegen weder im Methodeneditor öffnen, noch verändern.
- Andere Projektmethoden der Matrix Datenbank erscheinen nicht im Explorer. Sie sind jedoch ebenfalls im Debugger der Host Datenbank sichtbar.

Um Projektmethoden einer Komponente effektiv zu schützen, müssen Sie lediglich die Matrix Datenbank kompilieren und in Form einer .4dc Datei zur Verfügung stellen. Eine kompilierte Datenbank enthält keinen interpretierten Code. Wird die kompilierte Matrix Datenbank dann als Komponente installiert, gilt folgendes:

- Die gemeinsam genutzten Projektmethoden erscheinen im Explorer auf der **Seite Methoden** und lassen sich in den Methoden der Host Datenbank aufrufen. Ihr Inhalt erscheint jedoch weder in der Vorschau noch im Debugger.
- Die anderen Projektmethoden der Matrix Datenbank erscheinen nie.

Komponente installieren

Ordner Components

Um eine Komponente in einer 4D Datenbank zu installieren, kopieren Sie einfach die Strukturdatei der Matrix Datenbank in den Ordner **Components** der Host Datenbank. Sie können den Ordner **Components** an zwei Stellen setzen:

- Auf Ebene des ausführbaren 4D Programms: In diesem Fall sind die Komponenten für alle Datenbanken verfügbar, die mit diesem Programm geöffnet werden.
- Auf derselben Ebene wie die Strukturdatei der Datenbank: In diesem Fall sind die Komponenten nur in dieser Datenbank verfügbar.

Weitere Informationen dazu finden Sie im Abschnitt **Position der Ordner PlugIns und Components**.

4D sucht im Ordner **Components** nach Matrix Datenbanken vom Typ .4db (interpretiert), .4dc (kompiliert) oder .4dbase (Package). Andere Elemente, wie Datendateien oder Benutzer Strukturdateien (.4DA), werden ignoriert. Sie können Aliase oder Verknüpfungen auf diese Matrix Datenbanken verwenden. Das ist besonders in der Entwicklungsphase einer Komponente hilfreich, da alle Änderungen in der Matrix Datenbank sofort an alle Host Datenbanken übertragen werden.

Im Ordner **Components** können Sie unter Windows eine Verknüpfung, auf Mac OS ein Alias auf eine andere Datenbank einrichten, die selbst als Host Datenbank dient. Über ein Alias kann eine Host Datenbank so zu einer Komponente werden und umgekehrt. Beachten Sie jedoch, dass in diesem Fall nur eine Ebene Komponenten geladen wird. Nicht geladen werden die Komponenten einer Host Datenbank, die selbst als Komponente verwendet wird.

Interpretiert/ Kompiliert / Unicode

Eine Host Datenbank kann im interpretierten Modus sowohl interpretierte als kompilierte Komponenten, im Unicode Modus oder nicht, verwenden. Sie können in derselben Host Datenbank interpretierte und kompilierte Komponenten installieren. Sind dagegen mehrere kompilierte Komponenten vorhanden, müssen sie im gleichen Unicode Modus ausgeführt werden.

Eine Host Datenbank im kompilierten Modus kann keine interpretierten Komponenten verwenden. Hier sind nur kompilierte Komponenten möglich. Der Unicode Modus muss für Host Datenbanken und Komponenten derselbe sein.

Nachfolgende Tabelle zeigt die verschiedenen Möglichkeiten:

		Interpretierte Komponenten		Kompilierte Komponenten	
		Unicode	Nicht-Unicode	Unicode	Nicht-Unicode
Interpretierte Host Datenbank	Unicode	X	X	X (*)	X (*)
	Nicht-Unicode	X	X	X (*)	X (*)
Kompilierte Host Datenbank	Unicode	-	-	X	-
	Nicht-Unicode	-	-	-	X

(*) Sind mehrere kompilierte Komponenten installiert, müssen Sie im selben Unicode Modus operieren.

Hinweise:

- Eine interpretierte Datenbank mit interpretierten Komponenten ist kompilierbar, wenn sie keine Methoden von interpretierten Komponenten aufruft. Sonst erscheint beim Wählen des Menübefehls Compiler eine Warnung und Kompilieren ist nicht möglich.
- Eine interpretierte Methode kann in der Regel eine kompilierte Methode aufrufen, jedoch nicht umgekehrt. Davon ausgenommen ist der Aufruf durch die 4D Befehle **EXECUTE METHOD** und **EXECUTE FORMULA**.

Weitere Informationen dazu finden Sie im Abschnitt **Interaktion zwischen Komponenten und Host Datenbanken**.

Mac OS / Windows

Eine auf Mac OS entwickelte interpretierte Komponente lässt sich in einer Windows Umgebung installieren und umgekehrt. Im Gegensatz dazu müssen kompilierte Komponenten auf der Plattform ausgeführt werden, auf der sie kompiliert wurden, außer sie wurden für beide Plattformen kompiliert.

Client-Server

In der Server Datenbank installierte Komponenten werden, ähnlich wie Plug-Ins, automatisch auf die Client-Rechner übertragen. Wir raten jedoch davon ab, eine Komponente im Client/Server-Modus zu ändern, da dies nur lokal gespeichert und die Komponente nicht auf dem Server-Rechner aktualisiert wird.

Komponenten bei Startup laden

Komponenten werden beim Öffnen der Host Datenbank geladen.

- Enthält eine Komponente kompilierten Code und interpretierte Code, die nicht zusammenpassen, erscheint eine Fehlermeldung und die Komponente wird nicht in die Host Datenbank geladen.
- Fehlt eine Komponente beim Startup, öffnet sich die Host Datenbank nicht und der Fehler -10509 (Datenbank kann nicht geöffnet werden) wird erzeugt. Mit der 4D Funktion **COMPONENT LIST** können Sie beim Öffnen einer Datenbank prüfen, ob Komponenten vorhanden sind. Soll die Host Datenbank sowohl mit als auch ohne bestimmte Komponenten funktionieren, können Sie folgende Anweisung verwenden:

```

ARRAY TEXT($arrComponents_Txt;0)
COMPONENT LIST($arrComponents_Txt)
If(Find in array($arrComponents_Txt;"ComponentA")>0) // Komponente A soll nicht da sein
    EXECUTE METHOD("ComponentAMethod")
End if

```

Initialisierungscode ausführen

Eine Komponente kann 4D Code automatisch ausführen, wenn die Host Datenbank geöffnet und geschlossen wird, z.B. um Voreinstellungen oder Benutzerzustände in Bezug auf die Operation der Host Datenbank zu laden und sichern.

Aktionen, wie Initialisierung ausführen oder Code schließen, werden über die **Datenbankmethode On Host Database Event** verwaltet (siehe Handbuch *4D Programmiersprache*).

Aus Sicherheitsgründen müssen Sie, damit Sie diese Datenbank-methode aufrufen können, ihre Ausführung explizit in der Host Datenbank zulassen. Dazu müssen Sie in den Datenbank-Eigenschaften auf der Seite **Seite Sicherheit** die Option **Führe "On Host Database Event" Methode der Komponenten aus** markieren:

Optionen

Benutzer-Einstellungen in externer Datei

Führe "On Host Database Event" Methode der Komponenten aus

Namenskonflikte (verdeckte Methoden)

Im Gegensatz zu gemeinsam genutzten Objekte (siehe **Gemeinsam und nicht-gemeinsam genutzte Objekte**) sind gemeinsam genutzte Projektmethoden tatsächlich in der Datenbank vorhanden und werden nicht nur durch Ausführen von Code aufgerufen.

Folglich können Namenskonflikte auftreten, wenn eine gemeinsam genutzte Projektmethode in einer Komponente denselben Namen wie eine Projektmethode der Host Datenbank hat. In diesem Fall wird beim Ausführen des Code in der Host Datenbank die Methode der Host Datenbank aufgerufen. Es ist also möglich, die Methode einer Komponente mit einer eigenen Methode zu verdecken, z.B. um eine andere Funktionalität zu erhalten.

Wird der Code in der Komponente ausgeführt, wird natürlich die Methode der Komponente aufgerufen.

Beim Kompilieren der Datenbank wird die Verdeckung als Warnung angezeigt.

Hinweis: Nutzen zwei Komponenten gemeinsam Methoden mit demselben Namen, wird beim Kompilieren der Host Datenbank ein Fehler erzeugt.

Komponente entwickeln

Da eine Komponente die Form einer 4D Datenbank hat, lässt sie sich ähnlich wie eine Funktionalität in einer Datenbank entwickeln. Es gibt jedoch bestimmte Einschränkungen und Regeln, die in der Natur von Komponenten liegen.

Verwendbare und nicht-verwendbare Objekte

Eine Komponente kann die meisten Objekte von 4D aufrufen: Projektmethoden, Projektformulare, Menüleisten, Auswahllisten, Bilder aus der Bildbibliothek, usw.

Folgende Objekte lassen sich in einer Komponente nicht verwenden:

- Standard 4D Tabellen und Felder (Sie können aber Tabellen und Felder über externe Datenbanken erstellen)
- Tabellenformulare und dazugehörige Formularmethoden (eine Komponente kann dagegen ein Tabellenformular der Host Datenbank aufrufen)
- Benutzerformulare
- Datenbankmethoden und Trigger

Gibt es diese Objekte in Matrix Datenbanken, müssen sie nicht entfernt werden. Ein nicht-verwendbares Objekt wird beim Installieren der Komponente einfach ignoriert.

Hinweis: Benutzer und Gruppen sowie alle in der Matrix Datenbank vergebenen Zugriffsrechte werden in der Host Datenbank nicht berücksichtigt.

In der Host Datenbank sind im Designmodus nur die Projektmethoden sichtbar und auswählbar, die von der Komponente zur Mitbenutzung freigegeben sind. Die Komponente kann umgekehrt Projektmethoden aufrufen, die von der Host Datenbank zur Mitbenutzung freigegeben sind. Weitere Informationen dazu finden Sie im Abschnitt **Projektmethoden gemeinsam nutzen**.

Nur die von der Komponente veröffentlichten Formulare sind sichtbar und lassen sich als Unterformulare in Formulare der Host Datenbank integrieren. Weitere Informationen dazu finden Sie im Abschnitt **Formulare gemeinsam nutzen**.

Andere Objekte der Komponente, wie nicht-veröffentlichte Projektformulare, Auswahllisten, Menüs, etc. sind von der Komponente zwar verwendbar, jedoch nicht als Strukturobjekte in der Host Datenbank. Beachten Sie, dass manche Objekte nur einem Teil zugewiesen sind, andere von der Host Datenbank und Komponenten gemeinsam genutzt werden. Weitere Informationen dazu finden Sie im Abschnitt **Gemeinsam und nicht-gemeinsam genutzte Objekte**.

Eine Komponente kann Plug-Ins verwenden, die im 4D Programm oder in der Host Datenbank installiert sind. Sie können jedoch Plug-Ins nicht im Ordner **Components** installieren.

Datenbankmethoden und generische Parameter von Matrix Datenbanken (Web Ordner, Datenbank-Eigenschaften, usw.) werden nie berücksichtigt.

Nicht verwendbare Befehle

Die folgenden 4D Befehle sind für Komponenten nicht geeignet, da sie die Strukturdatei verändern - die nur im Lesemodus geöffnet ist. Bei Ausführung in einer Komponente wird der Fehler -10511 erzeugt mit der Meldung: „Der Befehl *Befehlsname* kann nicht in einer Komponente aufgerufen werden“:

ON EVENT CALL
Method called on event
SET PICTURE TO LIBRARY
REMOVE PICTURE FROM LIBRARY
SAVE LIST
ARRAY TO LIST
EDIT FORM
CREATE USER FORM
DELETE USER FORM
CHANGE PASSWORD
EDIT ACCESS
Set group properties
Set user properties
DELETE USER
CHANGE LICENSES
BLOB TO USERS
SET PLUGIN ACCESS

Hinweise:

- Die 4D Funktion **Current form table** gibt *Nil* zurück, wenn sie innerhalb einer Projektmethode aufgerufen wird. Folglich lässt sie sich in einer Komponente nicht verwenden.
- Auch Befehle zu Definition von SQL Daten, wie z.B. die SQL Befehle **CREATE TABLE**, **DROP TABLE**, sind in Komponenten nicht verwendbar.

Fehlerverwaltung

Eine Fehlerverwaltungsmethode, die über den Befehl **ON ERR CALL** eingerichtet wurde, gilt nur für die laufende Datenbank. Erzeugt eine Komponente einen Fehler, wird **nicht** die Fehlerverwaltungsmethode ON ERR CALL der Host-Datenbank aufgerufen, und umgekehrt.

Formulare verwenden

- Eine Komponente kann nur "Projektformulare" verwenden, also Formulare, die keiner Tabelle zugeordnet sind. Sie kann alle Projektformulare aus der Matrix Datenbank verwenden.
- Eine Komponente kann Tabellenformulare der Host Datenbank aufrufen. Beachten Sie, dass Sie in diesem Fall zur Angabe von Formularen im Code nicht Tabellennamen in eckigen Klammern [], sondern Zeiger verwenden müssen. **Hinweis:** Verwendet eine Komponente den 4D Befehl **ADD RECORD**, wird das aktuelle Eingabeformular im Kontext der Host Datenbank angezeigt. Folglich kann die Komponente, falls das Formular Variablen enthält, nicht darauf zugreifen. Weitere Informationen dazu finden Sie im Abschnitt **Interaktion zwischen Komponenten und Host Datenbanken**.
- Sie können Formulare von Komponenten als Unterformulare in der Host Datenbank verwenden, d.h. Sie können insbesondere Komponenten mit grafischen Objekten entwickeln. Zum Beispiel basieren **Widgets** von 4D auf Unterformularen in Komponenten. Weitere Informationen dazu finden Sie im Abschnitt **Formulare gemeinsam nutzen**.

Tabellen und Felder verwenden

Eine Komponente kann keine Tabelle und Felder verwenden, die in der 4D Struktur der Matrix Datenbank definiert wurden. Sie können jedoch **externe Datenbanken** erstellen und verwenden, und dann deren Tabellen und Felder gemäß Ihren Bedürfnissen verwenden. Sie können externe Datenbanken über SQL anlegen und verwalten. Eine externe Datenbank ist eine 4D Datenbank, die unabhängig von der 4D Hauptdatenbank ist, sie kann jedoch von der 4D Hauptdatenbank aus arbeiten. Eine externe Datenbank verwenden bedeutet, diese Datenbank temporär als die aktuelle Datenbank festlegen, d.h. als Zieldatenbank für SQL Anfragen, die von 4D ausgeführt werden. Sie können externe Datenbanken über den SQL Befehl **CREATE DATABASE** anlegen. Weitere Informationen dazu finden Sie im Handbuch *SQL Reference* unter **About external databases**.

Beispiel

Der folgende Code ist in einer Komponente enthalten und führt mit einer externen Datenbank drei Grundaktionen aus:

- Erstellt eine externe Datenbank, sofern noch keine existiert
- Fügt in der externen Datenbank Daten hinzu
- Liest Daten aus einer externen Datenbank ein

Die externe Datenbank erstellen:

```
<>MyDatabase:=Get 4D folder+"\MyDB" // (Windows) speichert die Daten in einem autorisierten Verzeichnis
```

Begin SQL

```
CREATE DATABASE IF NOT EXISTS DATAFILE :[<>MyDatabase];
USE DATABASE DATAFILE :[<>MyDatabase];
CREATE TABLE IF NOT EXISTS KEEPIT
(
  ID INT32 PRIMARY KEY,
  kind VARCHAR,
  name VARCHAR,
  code TEXT,
  sort_order INT32
);

CREATE UNIQUE INDEX id_index ON KEEPIT (ID);

USE DATABASE SQL_INTERNAL;
```

End SQL

In die externe Datenbank schreiben:

```
$Ptr_1:=$2 // Findet über Zeiger Daten aus der Host Datenbank
```

```
$Ptr_2:=$3
```

```
$Ptr_3:=$4
```

```
$Ptr_4:=$5
```

```
$Ptr_5:=$6
```

Begin SQL

```
USE DATABASE DATAFILE :[<>MyDatabase];
```

```
INSERT INTO KEEPIT
(ID, kind, name, code, sort_order)
VALUES
(:[$Ptr_1], :[$Ptr_2], :[$Ptr_3], :[$Ptr_4], :[$Ptr_5]);
```

```
USE DATABASE SQL_INTERNAL;
```

End SQL

Aus einer externen Datenbank einlesen:

```
$Ptr_1:=$2 // Greift über Zeiger auf Daten der Host Datenbank zu  
$Ptr_2:=$3  
$Ptr_3:=$4  
$Ptr_4:=$5  
$Ptr_5:=$6
```

Begin SQL

```
USE DATABASE DATAFILE :[<>MyDatabase];  
  
SELECT ALL ID, kind, name, code, sort_order  
FROM KEEPIT  
INTO :$Ptr_1, :$Ptr_2, :$Ptr_3, :$Ptr_4, :$Ptr_5;  
  
USE DATABASE SQL_INTERNAL;
```

End SQL

Ressourcen verwenden

Komponenten können Ressourcen verwenden, entweder die konventionellen Mac OS Ressourcen oder Dateien vom Typ XLIFFComponents.

Gemäß den Richtlinien zur Ressourcen-Verwaltung (siehe **Architektur der Anwendung**), müssen Ressource Dateien von Komponenten im Ordner **Resources** abgelegt werden, der neben der Datei .4db oder .4dc der Komponente liegt. Bei Komponenten mit der Architektur .4dbase (empfohlene Architektur) muss der Ordner **Resources** innerhalb dieses Ordners liegen.

Folgende Operationen laufen automatisch ab: Die Komponente lädt automatisch die XLIFF Dateien, die in ihrem Ordner Resources gefunden werden. Eine Komponente verwendet auch automatisch die konventionellen Mac OS Ressourcen aus der .rsr Datei, die neben den Dateien .4db oder .4dc liegt. "Konventionelle" Ressource Dateien im Ordner **Resources** müssen über 4D Befehle im Kapitel "**Ressourcen**" explizit in die Komponente geladen werden.

In einer Host Datenbank mit einer oder mehreren Komponenten haben jede Komponente sowie die Host Datenbanken ihre eigenen Ressource-Dateien. Ressourcen sind zwischen den verschiedenen Datenbanken aufgeteilt: Es ist nicht möglich, von Komponente B oder von der Host Datenbank auf die Ressourcen von Komponente A zuzugreifen. Weitere Informationen dazu finden Sie im Abschnitt **Gemeinsam und nicht-gemeinsam genutzte Objekte**.

Online Hilfe für Komponenten

4D bietet Entwicklern die Möglichkeit, ihre Komponenten mit Online Hilfe zu versehen. Die Vorgehensweise ist dieselbe wie für 4D Datenbanken (siehe **Anhang A: Eigene Online-Hilfe zuordnen**):

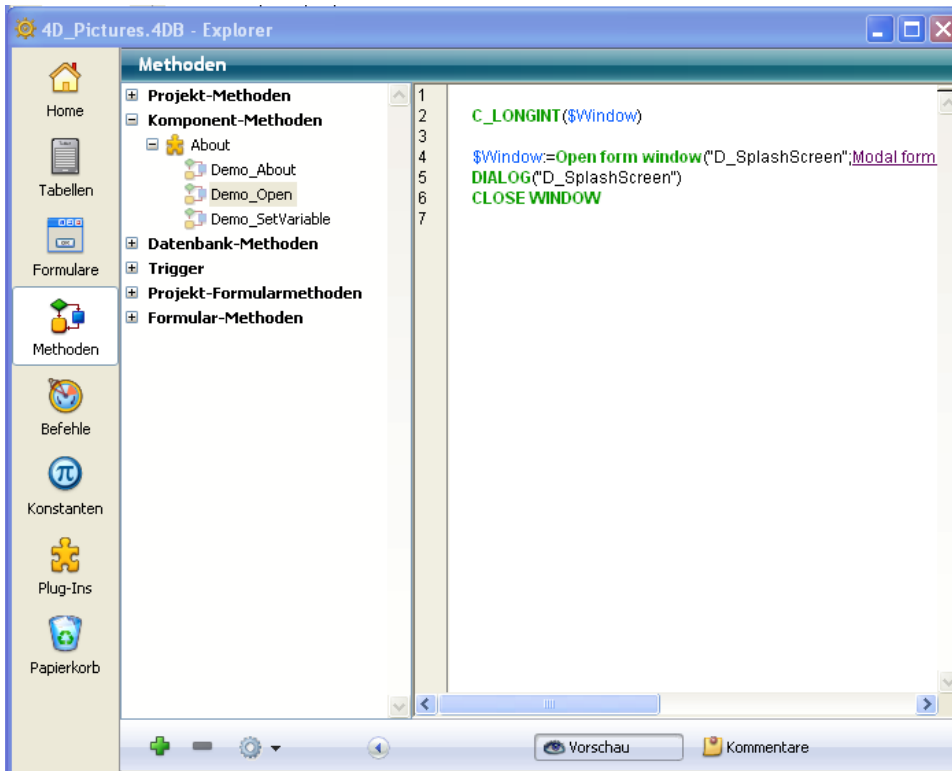
- Die Hilfedatei für Komponenten muss die Endung .htm, .html oder nur für Windows .chm haben.
- Die Hilfedatei muss neben der Strukturdatei der Komponente liegen und denselben Namen wie die Strukturdatei haben.
- Diese Datei wird dann automatisch in das Menü **Hilfe** der Anwendung geladen. Sie hat den Namen **Hilfe für...**, gefolgt vom Namen der Hilfedatei.

Interaktion zwischen Komponenten und Host Datenbanken

Komponenten anzeigen

Ist eine Komponente in einer Host Datenbank installiert, erscheint ihr Name im Explorer der Host Datenbank auf der Seite Methoden im Bereich Komponent-Methoden. Zur Mitbenutzung freigegebene Projektmethoden sind als hierarchische Listen aufgeführt.

Bei interpretierter Komponente erscheint der Inhalt der Methode in der Vorschau.



Weitere Informationen dazu finden Sie im Abschnitt [Projektmethoden gemeinsam nutzen](#).

Gemeinsam und nicht-gemeinsam genutzte Objekte

Komponenten haben ihren eindeutigen Namensbereich, d.h. in einer Komponente definierte Objekte sind namentlich in anderen Komponenten oder in der Host Datenbank nicht bekannt. So werden Namenskonflikte vermieden. Sie werden partitioniert genannt. So sind z.B. Variablen partitioniert, d.h. die von einer Komponente erstellte und verwendete Variable `<>Meinevar` vom Typ Lange Ganzzahl kann gleichzeitig existieren mit einer Variablen `<>Meinevar` vom Typ Text, die von einer Host Datenbank oder einer anderen Komponente erstellt und eingesetzt wird.

Andere Objekte teilen sich denselben Namensraum zwischen Host Datenbank und Komponenten. Solche Objekte müssen mit größerer Sorgfalt verwendet werden, sie ermöglichen jedoch der Host Datenbank und den Komponenten, miteinander zu kommunizieren. Diese Objekte werden mit „gemeinsam genutzt“ oder „nicht-partitioniert“ genannt.

Beispielsweise werden Mengen gemeinsam genutzt, d.h. erstellt eine Komponente die Menge *MeineMenge*, wird sie gelöscht, wenn die Host Datenbank folgende Anweisung ausführt:

```
CLEAR SET(mySet)
```

Nicht-gemeinsam genutzte Objekte

Folgende Objekte werden von Komponenten und der Host-Datenbank nicht gemeinsam genutzt (partitioniert):

- Stilvorlagen
- Hilfe Tipps
- Auswahllisten
- Über den Menüeditor erstellte Menüs und Menüleisten
- Projektmethoden ohne die Eigenschaft "Von Komponenten und Host Datenbank gemeinsam genutzt"
- Semaphoren
- Prozesse
- Variablen (lokal, Prozess und Interprozess)
- Systemvariablen (OK, Document, etc.)
- Tabellenformulare
- Projektformulare ohne die Eigenschaft "Als Unterformular veröffentlichen"
- Ressourcen und Referenzen zum Öffnen von Ressource Dateien

Gemeinsam genutzte Objekte

Folgende Objekte werden von Komponenten und der Host-Datenbank gemeinsam genutzt (nicht-partitioniert):

- Mengen
- Temporäre Auswahlen
- Hierarchische Listen, die eine Referenz verwenden (erstellt mit den 4D Funktionen **New list**, **Load list**, **Copy list** oder **BLOB to list**)
- Menüs und Menüleisten mit Kennnummer, die von der Funktion **Create menu** zurückgegeben wird
- Projektmethoden mit der Eigenschaft "Von Komponenten und Host Datenbank gemeinsam genutzt"
- Fensterreferenzen (**WinRef**)
- Projektformulare mit der Eigenschaft "Als Unterformular für die Host-Anwendung veröffentlichen"
- Referenzen auf die XML Struktur
- Referenzen auf offene Dateien (ausgenommen Resource Dateien)
- Zeiger

Hinweis: Die Host Datenbank ignoriert nicht-verwendbare Objekte aus der Matrix Datenbank. Weitere Informationen dazu finden Sie im Abschnitt **Verwendbare und nicht-verwendbare Objekte**.

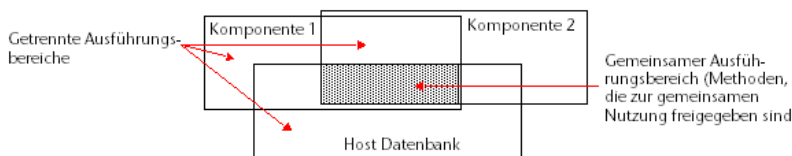
Projektmethoden gemeinsam nutzen

Per Definition werden alle Projektmethoden in die Komponente integriert (die Datenbank ist die Komponente), d.h. die Komponente kann sie aufrufen und ausführen.

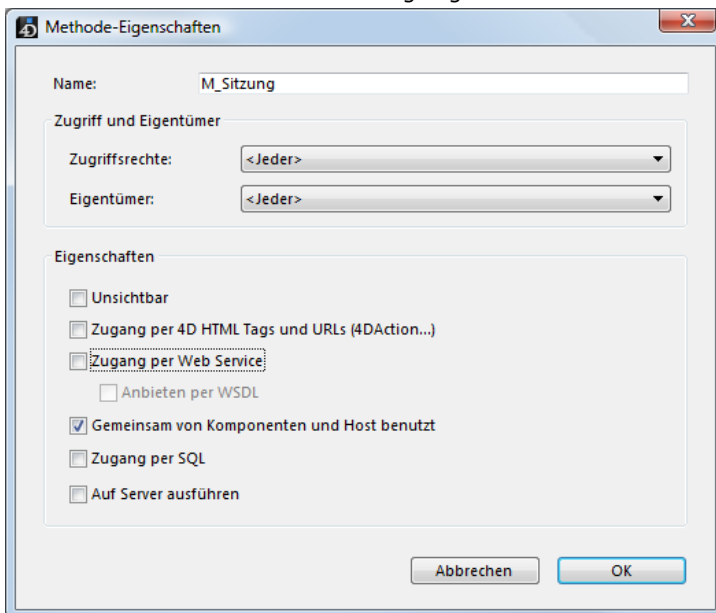
Andererseits sind diese Projektmethoden standardmäßig nicht sichtbar und lassen sich in der Host Datenbank auch nicht aufrufen. In der Matrix Datenbank müssen Sie die Methoden, die Sie mit der Host Datenbank teilen wollen, explizit freigeben. Diese Projektmethoden sind dann im Explorer auf der **Seite Methoden** sichtbar und lassen sich im Code der Host Datenbank aufrufen. Sie lassen sich jedoch im Methodeneditor der Host Datenbank nicht verändern. Diese Methoden sind die Schnittstelle zum Aufrufen der Komponente.

Aus Sicherheitsgründen kann die Komponente dagegen standardmäßig keine Projektmethoden ausführen, die zur Host Datenbank gehören. In manchen Fällen müssen Sie jedoch zulassen, dass eine Komponente auf die Projektmethoden Ihrer Host Datenbank zugreifen kann.

Dazu müssen Sie die Projektmethode explizit der Host Datenbank zuweisen, die Sie für die Komponente zugänglich machen wollen.



Diese Konfiguration wird im Dialogfenster **Methodeneigenschaften** über die Eigenschaft **Gemeinsam von Komponenten und Host Datenbank benutzt** festgelegt:



Sie können diese Eigenschaft auch auf mehrere Methoden gleichzeitig anwenden. Rufen Sie dazu im Kontextmenü des Explorers den Eintrag „Methodenauswahl-Eigenschaften“ auf (siehe **Methodeneigenschaften als Gruppe bearbeiten**).

Die Auswirkung dieser Option richtet sich nach der Verwendung der Datenbank: Wird sie als Komponente verwendet, ist die Methode in der Host Datenbank zugänglich und im Explorer sichtbar. Bei einer Host Datenbank ist die Methode für die Komponenten aufrufbar.

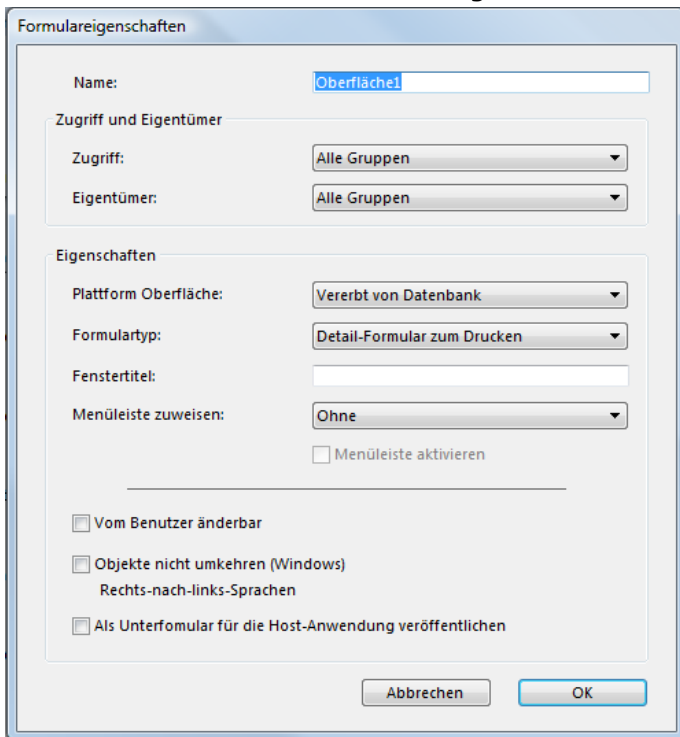
Formulare gemeinsam nutzen

Sie können die Formulare der Matrix Datenbank gemeinsam nutzen und als Unterformulare in der Host Datenbank verwenden.

Unterformular veröffentlichen (Komponente)

Auf der Seite der Komponenten (Matrix-Anwendung) lassen sich nur Projekt Unterformulare veröffentlichen.

Damit ein Komponentenformular als Unterformular in einer Host-Anwendung ausgewählt wird, muss es explizit als "veröffentlichtes Formular" deklariert sein. Wählen Sie dazu im Fenster **Formulareigenschaften** die Option **Als Unterformular für die Host-Anwendung veröffentlichen**:



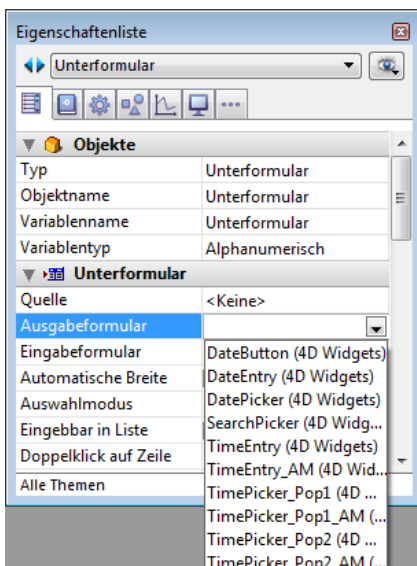
Hinweis: Sie erhalten dieses Dialogfenster über den Befehl **Formulareigenschaften** des Kontextmenüs des Formulars oder im Explorer über das Menü **Optionen** am unteren Rand (siehe **Formulareigenschaften (Explorer)**).

Sie müssen die Interaktionen zwischen Unterformular und Elternformular der Host Datenbank mit den Operationen und Werkzeugen verwalten, wie im Abschnitt **Unterformulare als Seite** beschrieben.

Unterformular aus Komponente verwenden (Host Datenbank)

Auf der Seite der Host Datenbank müssen Unterformulare aus Komponenten zwingend im Seitenmodus verwendet werden. Deaktivieren Sie dazu im Formulareditor, wenn das Unterformular im Elternformular ausgewählt ist, in der Eigenschaftenliste unter der Gruppe "Unterformular" die Option **Ausgabeformular**.

Wählen Sie im DropDown-Menü "Quelle" die Option **<Keine>**. Jetzt erscheinen die von Komponenten veröffentlichten Formulare im DropDown-Menü **Eingabeformular**. Der Name der Komponente wird in Klammern an den Formularnamen angehängt. In dieser Liste können Sie einfach das Formular auswählen, das in der Liste Eingabeformular verwendet werden soll.



Im Formular wird dann sofort eine neue Instanz des gewählten Objekts erstellt.

Variablen übergeben

Lokale, Prozess- und Interprozess Variablen werden nicht von Komponenten und Host Datenbanken gemeinsam genutzt, da sie nur im jeweiligen Namensraum existieren. Die einzige Möglichkeit, aus der Host Datenbank auf Variablen von Komponenten zuzugreifen und umgekehrt, sind Zeiger.

Beispiel mit Array:

- In der Host Datenbank:

```
ARRAY INTEGER(MyArray;10)
AMethod(->MyArray)
```

- In der Komponente enthält die Projektmethode **AMethod** folgendes:

```
APPEND TO ARRAY($1->;2)
```

Beispiele mit Variablen:

```
C_TEXT(myvariable)
component_method1(->myvariable)
C_POINTER($p)
$p:=component_method2(...)
```

Über Zeiger können Komponenten und Host Datenbank miteinander kommunizieren. Dabei müssen Sie folgendes berücksichtigen:

- Die 4D Funktion **Get pointer** gibt keinen Zeiger auf eine Variable der Host Datenbank zurück, wenn sie von einer Komponente aus aufgerufen wird und umgekehrt.
- Die Architektur für Komponenten lässt zu, dass dieselbe interpretierte Datenbank sowohl interpretierte als auch kompilierte Komponenten enthalten kann. In einer kompilierten Datenbank können dagegen nur kompilierte Komponenten verwendet werden. Für diesen Fall gilt folgende Regelung für Zeiger: Der Interpreter kann einen im kompilierten Modus erstellten Zeiger auflösen. Er kann jedoch im kompilierten Modus keinen Zeiger auflösen, der im interpretierten Modus erstellt wurde. Zeiger auf lokale Variablen können nicht verwendet werden. Dieses Prinzip wollen wir an folgendem Beispiel erläutern: Wir gehen von den beiden Komponenten C (kompiliert) und I (interpretiert) aus, die in derselben Host Datenbank installiert sind.
 - Definiert Komponente C die Variable *myCvar*, kann Komponente I über den Zeiger *->myCvar* auf diese Variable zugreifen.
 - Definiert Komponente I die Variable *myIvar* variable, kann Komponente C über den Zeiger *->myIvar* nicht auf diese Variable zugreifen. Diese Syntax verursacht einen Ausführungsfehler.
- Wir raten davon ab, Zeiger bei Komponenten über den 4D Befehl **RESOLVE POINTER** miteinander zu vergleichen, da durch die Partitionierung von Variablen gleichnamige Variablen nebeneinander existieren können, die jedoch in einer Komponente ganz anderen Inhalt haben können als in der Host Datenbank bzw. einer anderen Komponente. Auch der Variablentyp kann in beiden Kontexten unterschiedlich sein. Zeigen z.B. *myptr1* und *myptr2* jeweils auf eine Variable, kann der folgende Vergleich ein inkorrektes Ergebnis zurückgeben:

```
RESOLVE POINTER(myptr1;vVarName1;vtablenum1;vfieldnum1)
RESOLVE POINTER(myptr2;vVarName2;vtablenum2;vfieldnum2)
If(vVarName1=vVarName2)
//Dieser Test gibt Wahr zurück, auch wenn die Variablen unterschiedlich sind.
```

In diesem Fall müssen Sie die Zeiger miteinander vergleichen:

```
If(myptr1=myptr2) // Dieser Test gibt Falsch zurück
```

Auf Tabellen der Host Datenbank zugreifen

Auch wenn Komponenten keine Tabellen verwenden können, lassen sich trotzdem folgende 4D Routinen aufrufen:

```
DEFAULT TABLE
NO DEFAULT TABLE
Current default table
```

Diese Befehle sind in der Regel hilfreich, wenn eine Komponente Tabellen in der Host Datenbank verwenden muss. In diesem Fall sorgen Zeiger dafür, dass die Host Datenbank und Komponenten miteinander kommunizieren können:

```
C_LONGINT($1) //Nummer der Tabelle in Host Datenbank
$tablepointer:=Table($1)
DEFAULT TABLE($tablepointer->)
CREATE RECORD //Verwende die Standardtabelle der Host Datenbank
$fieldpointer:=Field($1;1)
$fieldpointer->:="Wert"
SAVE RECORD
```

Reichweite der Befehle der Programmiersprache

Eine Komponente kann jede Routine der 4D Programmiersprache verwenden, außer sie gehören zur Liste der nicht erlaubten Befehle. Weitere Informationen dazu finden Sie im Abschnitt **Nicht verwendbare Befehle**.

Von einer Komponente aufgerufene Befehle laufen im Kontext dieser Komponente, außer für den 4D Befehl **EXECUTE METHOD**. Dieser verwendet den Kontext der durch den Befehl angegebenen Methode. Eine Komponente kann auch die Lesebefehle des Kapitels "**Benutzer und Gruppen**" verwenden. Die Befehle lesen jedoch die Benutzer und Gruppen der Host Datenbank, da eine Komponente keine eigenen Benutzer und Gruppen hat.

Die 4D Routinen **SET DATABASE PARAMETER** und **Get database parameter** bilden hier eine Ausnahme. Sie gelten global für die Datenbank. Ruft eine Komponente diese Befehle auf, werden sie auf die Host Datenbank angewendet.

Darüberhinaus wurden spezifische Maßnahmen für die 4D Funktionen **Structure file** und **Get 4D folder** definiert, wenn sie im Rahmen von Komponenten verwendet werden (siehe Handbuch *4D Programmiersprache*).

Der 4D Befehl **COMPONENT LIST** ruft die Liste der Komponenten auf, die von der Host Datenbank geladen wurden.

Debugging

Arbeiten Sie mit nicht-kompilierten Komponenten, erscheint ihr Code im Debugger der Host Datenbank.

Der Debugger berücksichtigt den Ausführungsbereich partitionierter Objekte. Zeigen Sie z.B. im Fenster *Individuelle Überprüfung* den Wert der Variablen *var1* an und führen dann den zur Komponente gehörenden Code aus, die ebenfalls eine Variable *var1* enthält, wird der angezeigte Wert nicht aktualisiert. Sie müssen im Fenster *Individuelle Überprüfung* eine andere Instanz der Variablen anzeigen, um ihren Wert im aktuellen Kontext zu erhalten.

Anhänge

Dynamic Form Event

☰ Anhang A: Eigene Online-Hilfe zuordnen

☰ Anhang B: 4D über Befehlszeilen verwenden

☰ Anhang C: XLIFF Architektur

☰ Anhang D: Beschreibung der Protokolldateien

Dynamic Form Event

☰ Anhang A: Eigene Online-Hilfe zuordnen

Mit 4D können Sie jeder 4D Datenbank eine eigene Online-Hilfedatei zuordnen. So erhält der Benutzer zusätzliche Erläuterungen zur Funktionsweise Ihrer Anwendung.

Die selbst erstellte Online-Hilfe ist mit allen 4D Umgebungen kompatibel: Einzelplatz, Client/Server-Betrieb, interpretiert oder kompiliert, 4D Desktop Interpreted oder mit integrierter 4D Volume Desktop.

4D unterstützt für Hilfedateien die Formate CHM und HTML. Diese Formate entsprechen den Standards der Software-Industrie.

Außerdem können Sie jedem Formular der Datenbank einen bestimmten Teil dieser Hilfedatei zuordnen, und so dem Benutzer kontextuelle Online-Hilfe zur Verfügung stellen.

Online-Hilfedatei erstellen

Dateiformate

Sie können die Hilfedatei Ihrer Anwendung in folgenden Formaten liefern:

- **Datei ".CHM"**, das ist das Standardformat von Microsoft® Help für Windows. Dieser Dateityp lässt sich mit dem Programm Microsoft® HTML Help Executable (HH.EXE) öffnen, das im Betriebssystem integriert ist. Die Dateien sind mit Windows XP und Windows Vista kompatibel. Zum Erstellen und Kompilieren der .CHM-Dateien benötigen Sie das Programm HTML Help Workshop, das von Microsoft® kostenlos zur Verfügung gestellt wird.
- **Datei ".HTM"**, das ist das HTML-Format. Damit erstellen Sie eine für alle Plattformen identische Online-Hilfe, die dann über einen Web Browser angezeigt wird.

Hilfedatei der Datenbank zuweisen

Ist die Hilfedatei erstellt, müssen Sie diese einer Datenbank zuordnen. Sie wird dann automatisch aufgerufen, wenn der Benutzer die online Hilfe aufruft (siehe unteren Absatz [Online-Hilfe in einer 4D Datenbank aufrufen](#)) Dazu müssen folgende Bedingungen erfüllt sein:

- Der Name der Hilfedatei muss identisch sein mit dem Namen der Datenbankstruktur und je nach Format und Plattform die Erweiterung ".CHM" oder ".HTM" haben. Heißt die Strukturdatei z.B. „Rechnungen.4DB“, muss die Hilfedatei „Rechnungen.CHM“ oder „Rechnungen.HTM“ lauten.
- Die Hilfedatei wird neben die Strukturdatei der Datenbank oder in den Ordner **Resources** gelegt, der auf derselben Ebene wie die Strukturdatei liegt.

4D Server: Soll die Hilfedatei mit 4D Server auf allen Arbeitsstationen verfügbar sein, legen Sie die Datei in den Ordner **Resources**. Sie wird dann automatisch auf die Arbeitsstationen übertragen.

Auch Plug-Ins können eine Hilfedatei haben. Diese muss sowohl für Einzelplatz- als auch Client/Server-Anwendungen im Ordner **PlugIns** liegen. Die Plug-In Hilfedatei muss den Namen des Plug-In mit der Endung .CHM oder .HTM haben. Für Plug-Ins die als Ordner geliefert werden, muss die Hilfedatei an folgender Stelle liegen:

- Neben dem ausführbaren Plug-In, also entweder in *Contents/MacOS* oder *Contents/Windows*.
- oder im Ordner *Contents/Resources* des Plug-In.

Ebenso können Komponenten eine eigene Hilfedatei haben. Sie muss neben der Strukturdatei der Komponente liegen. Weitere Informationen dazu finden Sie im Abschnitt [Online Hilfe für Komponenten](#).

Kontextuelle Online-Hilfe zuweisen

Eine kontextuelle Online-Hilfe erstellen Sie, wenn Sie jedem Ihrer Formulare eine Nummer zuweisen. Ruft der Benutzer dann die Online-Hilfe von einem Formular aus auf, erscheint die Hilfeseite mit derselben Nummer wie das Formular.

Die kontextuelle Online-Hilfe ist verfügbar:

- Unter Windows, wenn sie im CHM-Format ist.
- Unter Windows und Mac OS, wenn sie im HTML-Format ist.

Eine Kennnummer weisen Sie dem 4D Formular in den Formulareigenschaften zu. Weitere Informationen dazu finden Sie im Abschnitt [Hilfe](#).

Der Hilfedatei müssen Sie dieselbe Nummer zuordnen wie dem Kapitel Ihres Formulars. Diese Operation variiert je nach dem verwendeten Format:

- Für CHM Dateien schlagen Sie in der Dokumentation zum Programm *HTML Help Workshop* nach.
- Für HTML Dateien legen Sie für jeden Abschnitt eine Marke fest und weisen ihm eine Nummer zu. Einen Abschnitt legen Sie mit folgender Marke an:

```
<A NAME="Section">
```

Zum Beispiel:

```
<A NAME="21">
```

Die URL des Abschnitts lautet:

```
<A HREF="#Section">...</A>
```

Zum Beispiel:

```
<A HREF="#21">...</A>
```

Ist die übergebene Nummer gleich 0 (Null), zeigt 4D die erste Seite der Hilfedatei an.

Online-Hilfe in einer 4D Datenbank aufrufen

In 4D rufen Sie die eigene Online-Hilfe einer Datenbank auf zwei Arten auf:

- Sie wählen im Menü **Hilfe>Datenbankname Hilfe**. In diesem Fall wird die erste Seite der Hilfedatei angezeigt.
- Sie drücken die Taste **F1** (Mac OS und Windows) oder die Taste **Hilfe** (nur Mac OS) wenn ein Formular im Anwendungsmodus angezeigt wird. Ist dem Formular eine Hilfedateinummer zugeordnet, erscheint die entsprechende Seite (kontextuelle Hilfe). Andernfalls erscheint die erste Seite der Hilfedatei.

Hinweis: Auf Mac OS lässt sich die Funktionsweise der **F1** Taste anpassen. Drückt der Benutzer im Anwendungsmodus diese Taste, gilt folgendes:

- Gibt es eine Datei "DatenbankName.HTM" neben der Strukturdatei der Datenbank, wird die Standardoperation zum Verwalten von 4D Online-Hilfe Dateien integriert. Die Hilfedatei wird in einem standardmäßigen Browser-Fenster geöffnet.
- Andernfalls führt 4D nichts aus und der Entwickler muss die Operation selbst steuern. Auf diese Weise kann er ein eigenes Hilfesystem einrichten oder der **F1** Taste eine andere Funktion zuweisen.

☰ Anhang B: 4D über Befehlszeilen verwenden

Sie können Ihre 4D Programme über Befehlszeilen im Mac OS Terminal oder in der Windows Konsole steuern (4D und 4D Server).

Mit dieser Funktionalität können Sie eine Datenbank aus der Ferne öffnen oder schließen. Das ist insbesondere beim Verwalten von Web Servern hilfreich.

Grundlagen

Sie können Befehlszeilen für 4D Programme im Mac OS X Terminal oder in der Windows Konsole ausführen:

- auf OS X müssen Sie den Befehl **Öffnen** verwenden
- unter Windows können Sie die Argumente direkt übergeben

Hinweis: Auf OS X können Sie die Parameter direkt übergeben, wenn Sie zum Ordner gehen, in dem das Programm innerhalb des Pakets abgelegt ist (Contents/MacOS Pfad). Liegt das 4D Paket z.B. im Ordner *MeinOrdner*, schreiben Sie die Befehlszeile wie folgt: `/MeinOrdner/4D.app/Contents/MacOS/4D`. Wir empfehlen jedoch, den Befehl **Öffnen** zu verwenden.

Befehle und Parameter

Es folgt die Beschreibung der Befehlszeilen und Parameter, die 4D Programme unterstützen. Diese Beispiele gehen davon aus, das Ihr 4D Programm auf dem Desktop gespeichert ist und die zu öffnende Anwendung im Ordner *Documents* liegt.

Hinweis: Der aktuelle Ordner des Benutzers wird auf OS X über den Befehl `"~"` aufgerufen, unter Windows über `"%HOMEPATH%"`

- **Programm starten**

Syntax: ApplikationPfad

Beispiel für OS X:

```
open ~/Desktop/4D.app
```

Beispiel für Windows:

```
%HOMEPATH%\Desktop\4D\4D.exe
```

Dieser Befehl führt dasselbe aus wie bei Doppelklick auf die 4D Applikation. Das Programm startet und auf dem Bildschirm erscheint das Dialogfenster zum Auswählen der Datenbank.

- **Programm mit Strukturdatei starten**

Syntax: ApplikationPfad[--Struktur] StrukturPfad | PackagePfad

Beispiel für OS X:

```
open ~/Desktop/4D.app --args ~/Documents/myDB.4dbase
```

Beispiel für Windows:

```
%HOMEPATH%\Desktop\4D\4D.exe %HOMEPATH%\Documents\myDB.4dbase\myDB.4db
```

Dieser Befehl startet das Programm und öffnet die zugewiesene Strukturdatei mit der aktuellen Datendatei. Es erscheint kein Dialogfenster.

- **Programm mit Strukturdatei und Datendatei starten**

Syntax: applicationPath --Struktur StrukturPfad | PackagePfad --Data DataPfad

Beispiel für OS X:

```
open ~/Desktop/4D.app --args --structure ~/Documents/myBase.4dbase --data ~/Documents/data/myData.4DD
```

Beispiel für Windows:

```
%HOMEPATH%\Desktop\4D\4D.exe --structure %HOMEPATH%\Documents\myBase.4dbase\myBase.4db --data %HOMEPATH%\Documents\data\myData.4DD
```

Dieser Befehl startet das Programm und öffnet die zugewiesene Strukturdatei mit der gewählten Datendatei. Es erscheint kein Dialogfenster.

- **Programm beenden** (nur OS X)

Syntax: kill -s INT process_ID

Beispiel:

```
Kill -s INT 323
```

Dieser Befehl führt dasselbe aus wie der Menübefehl **Beenden**. Die Prozesse werden der Reihe nach beendet; der Cache-Speicher wird gesichert und dann das Programm beendet.

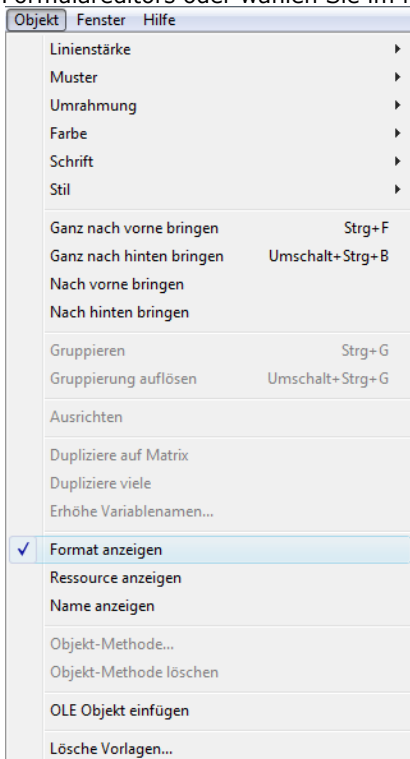
Hinweis: Die Prozessnummer des 4D Programms (*process_ID*) erhalten Sie über die Befehle **ps** oder **top**.

4D unterstützt den XLIFF Standard zum Lokalisieren von Oberflächentext und Titeln. Diese Technologie wird für 4D Anwendungen intern verwendet. 4D Entwickler sowie Plug-In Entwickler können die Vorteile dieser neuen Funktionalität für ihre eigenen Anwendungen und Plug-Ins nutzen.

Die Vorgehensweise für Oberflächen von 4D Anwendungen, die dynamisch lokalisiert werden, ist folgende: Alle Elemente, die auf Sprache basieren, wie Text, Bezeichnungen und Bilder, werden als Dateien außerhalb der Anwendung im Ordner mit Namen **Resources** gespeichert. Die Bezeichnungen und Bilder werden in Formularen, Menüs, etc. als Referenzen gespeichert. Diese Elemente werden beim Ausführen der Anwendung je nach Sprache dynamisch über die externen Dateien angezeigt. Der XLIFF Standard vereinheitlicht die Referenzierung und Anzeige von Bezeichnungen.

Hinweis: Sie können in XLIFF Dateien gespeicherte Strings auch direkt über die Funktion **Get localized string** lesen.

Im Formulareditor können Sie den tatsächlichen Inhalt der statischen Bereiche sehen. Klicken Sie dazu in den Bereich des Formulareditors oder wählen Sie im Menü **Objekt>Format anzeigen** oder **Name anzeigen**:



Hinweis zur Kompatibilität: Zur Wahrung der Kompatibilität unterstützt 4D weiterhin das vorige auf Ressourcen basierende System (vom Typ STR#). Beide Systeme lassen sich gleichzeitig verwenden, insbesondere in konvertierten Anwendungen. Weitere Informationen dazu finden Sie auf den folgenden Seiten.

Was ist XLIFF?

XLIFF (*XML Localization Interchange File Format*) ist ein Standard speziell für den Lokalisierungsprozess. Damit lässt sich die Entsprechung zwischen Quellsprache und Zielsprache in einer XML-Datei beschreiben.

Heutzutage ist XLIFF Standard eine neue Alternative zu Lokalisierungssystemen, die auf Ressourcen basieren. Zum Verwalten solcher Dateien gibt es auch verschiedene Tools, einige davon auch als Freeware.

Weitere Informationen dazu finden Sie in der offiziellen XLIFF 1.1 Spezifikation, die Sie im Internet laden können unter: <http://www.oasis-open.org/committees/xliff/documents/xliff-specification.htm>

Warnung: XLIFF Standard unterscheidet zwischen Groß- und Kleinschreibung.

XLIFF Strings aus 4D Datenbank aufrufen

Es gibt zwei Arten, XLIFFStrings von 4D aufzurufen:

- Über die Funktion **Get localized string**
- Über das dynamische Übersetzungssystem via Referenzen

Über die Programmiersprache

Über die Funktion **Get localized string** können Sie auf die Übersetzung eines String direkt in die aktuelle Sprache der Anwendung von überall in der Datenbank zugreifen. Über die Funktion **Get database localization** erfahren Sie die aktuelle Sprache.

Hinweis zur Kompatibilität: Die Befehle **Get indexed string** und **STRING LIST TO ARRAY** im Kapitel "Ressourcen" können auch die Vorteile von XLIFF Dokumenten nutzen. Ihre Verwendung wird jedoch nicht mehr empfohlen, sie werden

nur zur Wahrung der Kompatibilität beibehalten.

Stellen für Referenzen

In 4D lassen sich automatische XLIFF Referenzen an folgenden Stellen verwenden:

- Feld "Objektname" in der Eigenschaftenliste des Formulareditors
- Feld "Fenstertitel" im Dialogfenster Formulareigenschaften, das sich über den Explorer aufrufen lässt
- Statischer Text, wie Bezeichnungen für Schaltflächen, Optionsfelder, etc.
- Menüeditor (außer Syntax *Objektname*, siehe unten)
- Hilfetippeditor (außer Syntax *Objektname*, siehe unten).

Hinweise:

- Damit XLIFF Referenzen als solche erkannt werden, müssen sie am Anfang des Textbereichs stehen.
- XLIFF Referenzen lassen sich nicht im Hilfetippeditor anzeigen. Um auf XLIFF basierende Tipps zu verwenden, geben Sie die XLIFF Referenzen direkt im Feld **Hilfemeldungen** der Eigenschaftenliste ein.

Beispiel für eine XLIFF Datei:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?> <!DOCTYPE xliiff PUBLIC "-//XLIFF//DTD XLIFF//EN"
"http://www.oasis-open.org/committees/xliiff/documents/xliiff.dtd"><xliiff version="1.0">
  <file datatype="x-STR#" original="undefined" source-language="en-US" target-language="de">
    <body>
      <group id="20000" resname="Buttontitles">
        <trans-unit id="1" resname="CancelButton">
          <source>Cancel</source>
          <target>Abbrechen</target>
        </trans-unit>
        <trans-unit id="2">
          <source>Done</source>
          <target>Fertig</target>
        </trans-unit>
        <trans-unit id="3" resname="OKButton" >
          <source>Ok</source>
          <target>OK</target>
        </trans-unit>
      </group>
    </body>
  </file>
</xliiff>
```

Syntax der Referenzen

In 4D können Sie automatische XLIFF Referenzen mit verschiedenen Syntaxtypen verwenden:

- **:20000,3** Syntax
Verwenden Sie XLIFF Dateien mit dieser Syntax, gibt der erste Wert (im Beispiel 20000) das **id** Attribut des **group**-Elements an.
Der zweite Wert (im Beispiel 3) gibt das **id** Attribut des **trans-unit** Elements an.
Hinweis zur Kompatibilität: Diese Syntax ist identisch mit der früheren Generation der STR# Ressourcen vom Typ ":xxxx,yyy", wobei xxxx die Kennnummer der STR# Ressource und yyy die Elementnummer ist. So bedeutet :20000,3, dass 4D das 3. Element der STR# Ressource 20000 verwenden soll.
Auf diese Weise können Sie für konvertierte Datenbanken das auf Ressourcen basierende System beibehalten, andererseits das neue auf XLIFF basierende System einführen. 4D versucht zuerst, die Werte, welche zu 20000,3 gehören, in allen geladenen XLIFF Dateien zu finden und zu verwenden; wird der String nicht gefunden, sucht 4D in allen geladenen Ressource-Dateien danach.
- **:xliiff:OKButton** Syntax
Diese alternative Syntax lässt sich nur mit geladenen XLIFF Dateien verwenden. In diesem Fall bestimmt der referenzierte Wert (im Beispiel Schaltfläche OK) das Attribut **resname** des **trans-unit** Elements.
- **Objektname** Syntax (basiert auf dem Feld "Name" in der Eigenschaftenliste)
Diese Syntax wurde eingerichtet, um die Übersetzung von Formularen zu vereinfachen. Sie ist nur für Formulare verwendbar. Sie umgibt das trans-unit Element in der XLIFF-Datei mit zwei group Elementen, deren Attribute resname jeweils den Tabellennamen und den Formularnamen enthalten.
 - Beispiel (Tabellenformular)
Um den Titel der Schaltfläche mit dem Objektname "SaveButton" in Formular1 der Tabelle [Clients] zu übersetzen, fügen Sie in die XLIFF Datei folgende Zeilen ein:

```
<group resname="[Clients]"> <group resname="Form1"> <trans-unit resname="SaveButton">
<source>Save</source> <target>Sichern</target> </trans-unit> </group> </group>
```

- Beispiel (Projektformular)
Bei einem Projektformular wird der Tabellenname ersetzt durch [ProjectForm]. Der Code lautet dann:

```
<group resname="[ProjectForm]"> <group resname="Form1"> <trans-unit resname="SaveButton">
<source>Save</source> <target>Sichern</target> </trans-unit> </group> </group>
```


Verwenden Sie in Ihrer Datenbank verschiedene Syntaxarten, gilt beim Suchen nach einer gültigen Übersetzung in den XLIFF Dateien die Priorität:

1. ":20000,3" Syntax
2. ":xliff:OKButton" Syntax
3. "MyTitle" Syntax

Im Formulareditor können Sie zwischen der XLIFF Referenz und ihrer Übersetzung in die aktuelle Sprache wechseln. Wählen Sie dazu im Menü **Objekt>Name anzeigen** bzw. **Ressource anzeigen**. Weitere Informationen dazu finden Sie im Abschnitt **Referenzen in statischem Text verwenden**.

Angepasste XLIFF Dateien installieren

Um eine XLIFF Architektur für Ihre lokalisierte Anwendung einzurichten, müssen Sie lediglich ein bzw. mehrere gültige XLIFF Dateien erstellen und diese im Ordner **Resources** der Datenbank speichern (Paket auf Mac OS).

Weitere Informationen zum Ordner **Resources** finden Sie im Abschnitt **Ordner Resources verwalten**.

Der vollständige Pfadname lautet:

- Windows:
MeineDatenbank\Resources\Lang.lproj\MyEnLoc.xlf
- Mac OS:
MeineDatenbank:Resources:Lang.lproj:MyEnLoc.xlf

Es gilt folgendes:

- *MeineDatenbank* ist der Ordner mit den Dateien der Datenbank. *MeinPaket* ist das Datenbankpaket auf Mac OS.
- *Lang.lproj* ist ein Ordner mit den XLIFF Dateien für die Sprache Lang. Sein Name muss dem internationalen Standard entsprechen (siehe nächster Abschnitt). Beispiel: Für englische Versionen sollte er *en.lproj* lauten.

4D lädt die XLIFF Dateien automatisch in den Ordner mit der aktuellen Sprache der Datenbank. Um die Sprache der Datenbank einzurichten, führt 4D aufeinanderfolgende Suchläufe im Ordner **Resources** der Datenbank durch, um die passende Sprache zu finden (in der Priorität 1-4):

1. Sprache des Betriebssystems (auf Mac OS lassen sich mehrere Sprachen mit Prioritätenskala definieren. 4D verwendet diese Einstellungen)
2. Sprache der 4D Anwendung
3. Englisch
4. Sind diese Suchläufe nicht erfolgreich, wird die zuerst im Ordner **Resources** gefundene Sprache geladen.

Die 4D Funktion **Get database localization** findet heraus, welche aktuelle Sprache für die Datenbank festgelegt wurde. Wird eine Sprachvariante verwendet, die in den XLIFF Dateien nicht verfügbar ist, wird die am nächsten liegende Sprachvariante verwendet.

Sie können den XLIFF Dateien beliebige Namen geben, sie müssen nur die Endung ".xlf." haben. Sie können mehrere XLIFF Dateien in denselben Sprachenordner legen; sie werden in alphabetischer Reihenfolge der Dateinamen geladen.

Ordnername .lproj

Der Name des Ordners ".lproj" muss einen der nachfolgenden Standards berücksichtigen. 4D sucht nach einem gültigen Ordnernamen in folgender Reihenfolge:

1. **Language-Regional Codes**
In dieser Kombination wird eine Sprache beschrieben durch einen Ländercode (ISO639-1) + Bindestrich + Regionalcode (ISO3166) oder ein Skript Code mit 4 Buchstaben.
Zum Beispiel, "fr-ca" (*also fr-ca.lproj*) für kanadisches Französisch.
2. **ISO639-1**
Dieser Standard definiert jede Sprache mit zwei Buchstaben, zum Beispiel, "en" (*also en.lproj*) für die englische Sprache, "de" für die deutsche Sprache.
Ref: http://www.loc.gov/standards/iso639-2/php/English_list.php
3. **Legacy name**
Hier wird der Sprachename in der englischen Schreibweise komplett ausgeschrieben, zum Beispiel "english" (*also english.lproj*) für die englische Sprache.
Hinweis: Die beiden ersten Konventionen gelten für Mac OS ab Version 10.4. In früheren Versionen kann nur der komplett ausgeschriebene Name, also der "Legacy" Name verwendet werden.

Eine Übersicht der von 4D unterstützten Sprach-Codes finden Sie im Abschnitt .

Werden mehrere Sprachdefinitionen gefunden, sucht 4D immer nach der am nächsten liegenden Übersetzung. Ist die aktuelle Spracheinstellung des Systems z.B. kanadisches Französisch, sucht 4D erst unter "fr-ca" nach Übersetzungen, falls hier nichts gefunden wird, dann unter "fr".

Hinweis: Das gleiche Prinzip gilt für XLIFF Dateien im Tag "target language". Verwenden Sie dieses Attribut innerhalb der XLIFF Dateien mit Bedacht, da eine Datei im Ordner "fr-ca.lproj" mit dem Tag "target language="fr" als eine Übersetzung in "fr" und nicht "fr-ca" gewertet wird.

Eigene XLIFF Dateien erneut laden

XLIFF Dateien werden während der Lokalisierung Ihrer Anwendung dynamisch geladen, so dass Sie sehen können, ob die lokalisierten Wörter oder Sätze in die Objektrahmen von Schaltflächen, PopUp-Menüs, etc. hineinpassen. Das erneute Laden erfolgt, wenn 4D wieder in den Vordergrund kommt und sich Änderungsdatum oder -zeit seit dem letzten Laden geändert haben. Das aktuelle Formular wird gleichzeitig erneut geladen.

Programmiersprache Codes

Nachfolgende Tabelle zeigt die Programmiersprache Codes, die 4D zum Verwalten von XLIFF Dateien unterstützt.

Sprache	ISO639-1	"Legacy"	ISO3166
AFRIKAANS	af	afrikaans	
ALBANIAN	sq	albanian	
ARABIC_SAUDI_ARABIA	ar	arabic	sa
ARABIC_IRAQ	ar	arabic	iq
ARABIC_EGYPT	ar	arabic	eg
ARABIC_LIBYA	ar	arabic	ly
ARABIC_ALGERIA	ar	arabic	dz
ARABIC_MOROCCO	ar	arabic	ma
ARABIC_TUNISIA	ar	arabic	tn
ARABIC_OMAN	ar	arabic	om
ARABIC_YEMEN	ar	arabic	ye
ARABIC_SYRIA	ar	arabic	sy
ARABIC_JORDAN	ar	arabic	jo
ARABIC_LEBANON	ar	arabic	lb
ARABIC_KUWAIT	ar	arabic	kw
ARABIC_UAE	ar	arabic	ae
ARABIC_BAHRAIN	ar	arabic	bh
ARABIC_QATAR	ar	arabic	qa
BASQUE	eu	basque	
BELARUSIAN	be	belarusian	
BULGARIAN	bg	bulgarian	
CATALAN	ca	catalan	
CHINESE_TRADITIONAL	zh	chinese	hant
CHINESE_SIMPLIFIED	zh	chinese	hans
CHINESE_HONGKONG	zh	chinese	hk
CHINESE_SINGAPORE	zh	chinese	sg
CROATIAN	hr	croatian	
CZECH	cs	czech	
DANISH	da	danish	
DUTCH	nl	dutch	
DUTCH_BELGIAN	nl	dutch	be
ENGLISH_US	en	english	
ENGLISH_UK	en	english	gb
ENGLISH_AUSTRALIA	en	english	au
ENGLISH_CANADA	en	english	ca
ENGLISH_NEWZEALAND	en	english	nz
ENGLISH_EIRE	en	english	ie
ENGLISH_SOUTH_AFRICA	en	english	za
ENGLISH_JAMAICA	en	english	jm
ENGLISH_BELIZE	en	english	bz
ENGLISH_TRINIDAD	en	english	tt
ESTONIAN	et	estonian	
FAEROESE	fo	faorese	
FARSI	fa	persian	
FINNISH	fi	finnish	
FRENCH	fr	french	
FRENCH_BELGIAN	fr	french	be
FRENCH_CANADIAN	fr	french	ca
FRENCH_SWISS	fr	french	ch
FRENCH_LUXEMBOURG	fr	french	lu
GERMAN	de	german	
GERMAN_SWISS	de	german	ch
GERMAN_AUSTRIAN	de	german	at
GERMAN_LUXEMBOURG	de	german	lu
GERMAN_LIECHTENSTEIN	de	german	li
GREEK	el	greek	
HEBREW	he	hebrew	
HUNGARIAN	hu	hungarian	
ICELANDIC	is	iceland	
INDONESIAN	id	indonesian	
ITALIAN	it	italian	
ITALIAN_SWISS	it	italian	ch
JAPANESE	ja	japanese	
KOREAN_WANSUNG	ko	korean	
KOREAN_JOHAB	ko	korean	

LATVIAN	lv	latvian	
LITHUANIAN	lt	lithuanian	
NORWEGIAN	no	norwegian	
NORWEGIAN_NYNORSK	nn	nynorsk	no
POLISH	pl	polish	
PORTUGUESE	pt	portuguese	
PORTUGUESE_BRAZILIAN	pt	portuguese	br
ROMANIAN	ro	romanian	
RUSSIAN	ru	russian	
SERBIAN_LATIN	sr	serbian	latn
SERBIAN_CYRILLIC	sr	serbian	cyr
SLOVAK	sk	slovak	
SLOVENIAN	sl	slovenian	
SPANISH_CASTILLAN	es	spanish	
SPANISH_MEXICAN	es	spanish	mx
SPANISH_MODERN	es	spanish	
SPANISH_GUATEMALA	es	spanish	gt
SPANISH_COSTA_RICA	es	spanish	cr
SPANISH_PANAMA	es	spanish	pa
SPANISH_DOMINICAN_REPUBLIC	es	spanish	do
SPANISH_VENEZUELA	es	spanish	ve
SPANISH_COLOMBIA	es	spanish	co
SPANISH_PERU	es	spanish	pe
SPANISH_ARGENTINA	es	spanish	ar
SPANISH_ECUADOR	es	spanish	ec
SPANISH_CHILE	es	spanish	cl
SPANISH_URUGUAY	es	spanish	uy
SPANISH_PARAGUAY	es	spanish	py
SPANISH_BOLIVIA	es	spanish	bo
SPANISH_EL_SALVADOR	es	spanish	sv
SPANISH_HONDURAS	es	spanish	hn
SPANISH_NICARAGUA	es	spanish	ni
SPANISH_PUERTO_RICO	es	spanish	pr
SWEDISH	sv	swedish	
SWEDISH_FINLAND	sv	swedish	fi
THAI	th	thai	
TURKISHLITHUANIAN	lt	lithuanian	
NORWEGIAN	no	norwegian	
NORWEGIAN_NYNORSK	nn	nynorsk	no
POLISH	pl	polish	
PORTUGUESE	pt	portuguese	
PORTUGUESE_BRAZILIAN	pt	portuguese	br
ROMANIAN	ro	romanian	
RUSSIAN	ru	russian	
SERBIAN_LATIN	sr	serbian	latn
SERBIAN_CYRILLIC	sr	serbian	cyr
SLOVAK	sk	slovak	
SLOVENIAN	sl	slovenian	
SPANISH_CASTILLAN	es	spanish	
SPANISH_MEXICAN	es	spanish	mx
SPANISH_MODERN	es	spanish	
SPANISH_GUATEMALA	es	spanish	gt
SPANISH_COSTA_RICA	es	spanish	cr
SPANISH_PANAMA	es	spanish	pa
SPANISH_DOMINICAN_REPUBLIC	es	spanish	do
SPANISH_VENEZUELA	es	spanish	ve
SPANISH_COLOMBIA	es	spanish	co
SPANISH_PERU	es	spanish	pe
SPANISH_ARGENTINA	es	spanish	ar
SPANISH_ECUADOR	es	spanish	ec
SPANISH_CHILE	es	spanish	cl
SPANISH_URUGUAY	es	spanish	uy
SPANISH_PARAGUAY	es	spanish	py
SPANISH_BOLIVIA	es	spanish	bo
SPANISH_EL_SALVADOR	es	spanish	sv
SPANISH_HONDURAS	es	spanish	hn

SPANISH_NICARAGUA	es	spanish	ni
SPANISH_PUERTO_RICO	es	spanish	pr
SWEDISH	sv	swedish	
SWEDISH_FINLAND	sv	swedish	fi
THAI	th	thai	
TURKISH	tr	turkish	
UKRAINIAN	uk	ukrainian	
VIETNAMESE	vi	vietnamese	

III Anhang D: Beschreibung der Protokolldateien

4D Applikationen können verschiedene Protokolldateien/Logbücher erstellen, die hilfreich sind, um das Ausführen von Operationen zu optimieren oder Fehler zu beheben. Protokolle werden in der Regel über Selektoren der Befehle **SET DATABASE PARAMETER** oder **WEB SET OPTION** gestartet und gestoppt und im Ordner **Logs** der Anwendung gespeichert (siehe Abschnitt **Beschreibung der 4D Dateien**).

Die protokollierten Daten müssen analysiert werden, um Probleme zu finden und zu beheben. Dieser Anhang beschreibt ausführlich den Inhalt folgender Logbücher:

- 4DRequestsLog.txt
- 4DRequestsLog_ProcessInfo.txt
- HTTPDebugLog.txt
- 4DDebugLog.txt

Die Dateien nutzen einige Felder gemeinsam, so dass Sie eine Chronologie erstellen und beim Debuggen Einträge miteinander verbinden können:

- *sequence_number*: Diese Nummer ist einmalig in allen Debug Logs und wird für jeden neuen Eintrag unabhängig vom Logbuch erhöht, so dass Sie die exakte Sequenz der Operationen erkennen können.
- *connection_uuid*: Für jeden 4D Prozess, der auf einem 4D Client erstellt wurde, welcher an einen Server angemeldet ist. Diese UUID der Verbindung wird auf Server- und Client-Seite geführt, so dass Sie leicht den remote Client identifizieren können, welcher den Prozess gestartet hat.

4DRequestsLog.txt

Dieses Logbuch protokolliert Standardanfragen jeweils auf dem Rechner mit 4D Server oder 4D Remote, der den Befehl ausgeführt hat (außer Web Anfragen).

Logbuch starten:

- auf dem Server:

```
SET DATABASE PARAMETER(4D Server log recording;1) //Server Seite
```

- auf einem Client:

```
SET DATABASE PARAMETER(Client Log Recording;1) //Remote Seite
```

Hinweis: Diese Anweisung startet auch das Logbuch 4DRequestsLog_ProcessInfo.txt (siehe nächster Abschnitt).

Kopfzeile

Diese Datei startet mit folgenden Kopfzeilen:

- Identifier der protokollierten Sitzung
- Hostname des Server, der die Applikation hostet
- User Login Name: Login auf dem Betriebssystem des Benutzers, der die 4D Applikation auf dem Server gestartet hat.

Inhalt

Für jede Anfrage werden folgende Felder protokolliert:

Feldname	Beschreibung
sequence_number	Einmalige und sequentielle Nummer der Operation in der protokollierten Sitzung
time	Datum und Zeit im Format 'MM/DD/YY, HH:MM:SS'
systemid	System ID
component	Signatur der Komponente (z.B., '4SQLS' oder 'dbmg')
process_info_index	Entspricht dem "index" Feld im Logbuch 4DRequestsLog_ProcessInfo.txt und ermöglicht, eine Anfrage an einen Prozess zu binden.
request	Anfrage ID in C/S oder Meldungsstring für SQL Anfragen oder LOG EVENT Meldungen
bytes_in	Anzahl der empfangenen Bytes
bytes_out]Anzahl der gesendeten Bytes
duration	Beanspruchte Zeit in Millisekunden zum Durchführen einer Aktion
task_kind	Preemptive oder kooperativ (jeweils 'p' oder 'c')

4DRequestsLog_ProcessInfo.txt

Dieses Logbuch protokolliert Angaben zu jedem Prozess, ausgeführt auf dem 4D Server Rechner oder dem 4D Remote Rechner, welcher den Befehl ausgeführt hat (außer Web Anfragen).

Logbuch starten:

- auf dem Server:

```
SET DATABASE PARAMETER(4D Server log recording;1) //Server Seite
```

- auf einem Client:

```
SET DATABASE PARAMETER(Client_Log_Recording;1) //Remote Seite
```

Hinweis: Diese Anweisung startet auch das Logbuch 4DRequestsLog.txt (siehe voriger Absatz).

Kopfzeile

Diese Datei startet mit folgenden Kopfzeilen:

- Identifier der protokollierten Sitzung
- Hostname des Server, der die Applikation hostet
- User Login Name: Login auf dem Betriebssystem des Benutzers, der die 4D Applikation auf dem Server gestartet hat.

Inhalt

Für jeden Prozess werden folgende Felder protokolliert:

Feldname	Beschreibung
sequence_number	Einmalige und sequentielle Nummer der Operation in der protokollierten Sitzung
time	Datum und Zeit im Format 'MM/DD/YY, HH:MM:SS'
process_info_index	Einmalige und sequentielle Prozessnummer
CDB4DBaseContext	Kontext UUID der DB4D Komponente der Datenbank
systemid	System ID
server_process_id	Prozess ID auf dem Server
remote_process_id	Prozess ID auf dem Client
process_name	Prozessname
cID	Identifier der 4D Verbindung
uID	Identifier von 4D Client
IP	IPv4/IPv6 Adresse des Client
host_name	Hostname des Client
user_name	Name des Benutzer Login auf dem Client
connection_uuid	UUID Identifier der Prozessverbindung
server_process_unique_id	Einmalige Prozess ID auf dem Server

HTTPDebugLog.txt

Dieses Logbuch protokolliert jede HTTP Anfrage und jede Antwort in informativer Form. Protokolliert werden die Anfragen, inkl. Kopfzeile; optional lassen sich auch Hauptteile protokollieren.

Logbuch starten:

```
WEB SET OPTION(Web debug log;wdl enable without body) //andere Werte sind verfügbar
```

Folgende Felder werden für Anfrage und Antwort protokolliert:

Feldname	Beschreibung
SocketID	ID des Socket, der zur Kommunikation verwendet wird
PeerIP	IPv4 Adresse des Host (Client)
PeerPort	Vom Host verwendeter Port (Client)
TimeStamp	Zeitstempel in Millisekunden (seit Systemstart)
ConnectionID	UUID der Verbindung (UUID des VTCPsocket, der zur Kommunikation verwendet wird)
SequenceNumber	Einmalige und sequentielle Nummer der Operation in der protokollierten Sitzung

4DDebugLog.txt (Standard)

Dieses Logbuch protokolliert jedes Ereignis, das auf 4D Programmiererebene auftritt. Der Standardmodus zeigt eine einfache Ansicht der Ereignisse.

Logbuch starten:

```
SET DATABASE PARAMETER(Debug_Log_Recording;2) //Standard
```

Für jedes Ereignis werden folgende Felder protokolliert:

Spalte #	Beschreibung
1	Einmalige und sequentielle Nummer der Operation in der protokollierten Sitzung
2	Vergangene Zeit in Millisekunden seit Protokollstart
3	Prozess ID (p=xx) und einmalige Prozess ID (puid=xx)
4	Stapelebene
5	Kann Befehlsname/ Methodenname/Meldung/ Task Start Stop Info/Plug-In Name, Ereignis oder Callback/Connection UUID sein
6	Für die protokollierte Operation beanspruchte Zeit in Millisekunden (unterscheidet sich von der 2. Spalte)

4DDebugLog.txt (tabular)

Dieses Logbuch protokolliert jedes Ereignis, das auf der 4D Programmierzebene auftritt in einem kompakten Format mit Tabulatoren, das zusätzliche Angaben enthält (verglichen mit dem Standardformat).

Logbuch starten:

```
SET DATABASE PARAMETER(Debug_Log_Recording;2+4) //erweitertes Format mit Tabulatoren
```

Folgende Felder werden für jedes Ereignis protokolliert:

Spalte #	Beschreibung
1	Einmalige und sequentielle Nummer der Operation in der protokollierten Sitzung
2	Vergangene Zeit seit Protokollstart im Format "hh:mm:ss:ms" (bei Bedarf wird ein Zähler für Tage vorangestellt. Wurde das Protokoll z.B. vor 3 Tagen gestartet, könnte das so aussehen "3+11:58:23:163")
3	Prozess ID
4	Einmalige Prozess ID
5	Stapelebene
6	Kann folgendes anzeigen (je nach Art der protokollierten Eingabe in der 8. Spalte): <ul style="list-style-type: none"> • ID eines Befehls der Programmiersprache (wenn Typ=1) • Ein Methodenname (wenn Typ=2) • Eine Kombination von PluginIndex;PluginBefehl (wenn Typ=4, 5, 6 oder 7). Kann etwas enthalten wie '3;2' • Eine Task Connection UUID (wenn Typ=8) • oder eine 'startende Sequenznummer' beim Schließen einer Stapelebene (dies sollte der Sequenznummer der gestarteten aktuellen Aktion entsprechen) <pre>121 15:16:50:777 5 8 0 Methodenaufruf 2 0 122 15:16:50:777 5 8 1 283 1 0 123 15:16:50:777 5 8 1 122 -1 0 3 124 15:16:50:777 5 8 0 121 -2 0 61</pre> <p>Hier ist in der Zeile (124) der Wert '121' der 6. Spalte gleich der Sequenznummer der ersten Zeile (Stapelebene 0). In der Zeile (123) ist der Wert '122' der 6. Spalte gleich der Sequenznummer der oberen Zeile (Stapelebene 1) etc.</p>
7	Parameter, die Befehlen, Methoden oder Plug-ins übergeben wurden
8	Art der protokollierten Operation. Dieser Wert kann absolut sein: <ul style="list-style-type: none"> 1: Befehl 2: Methode 3: Meldung (nur von Befehl LOG EVENT gesendet) 4: PluginMeldung 5: PluginEreignis 6: PluginBefehl 7: PluginCallback 8: Task <p>Ein negativer Wert bedeutet lediglich, dass es sich um das schließende Gegenstück der Stapelebene handelt (siehe 8. Spalten in den Zeilen 123 und 124 des obigen Protokolls).</p>
9	Formularereignis, wenn vorhanden; ist sonst leer (die Spalte wird benützt, wenn Code in einer Formularmethode oder einem Skript ausgeführt wird)
10	Verbrauchte Zeit in Mikrosekunden für die aktuell protokollierte Aktion; nur für schließende Stapelebenen (siehe 10. Spalten in Zeilen 123 und 124 im Protokoll oben)