



4th Dimension バージョン 2004.1 へようこそ。本ドキュメントは、4th Dimension プロダクトラインのアプリケーションおよびプラグインの新機能と変更点について説明します。

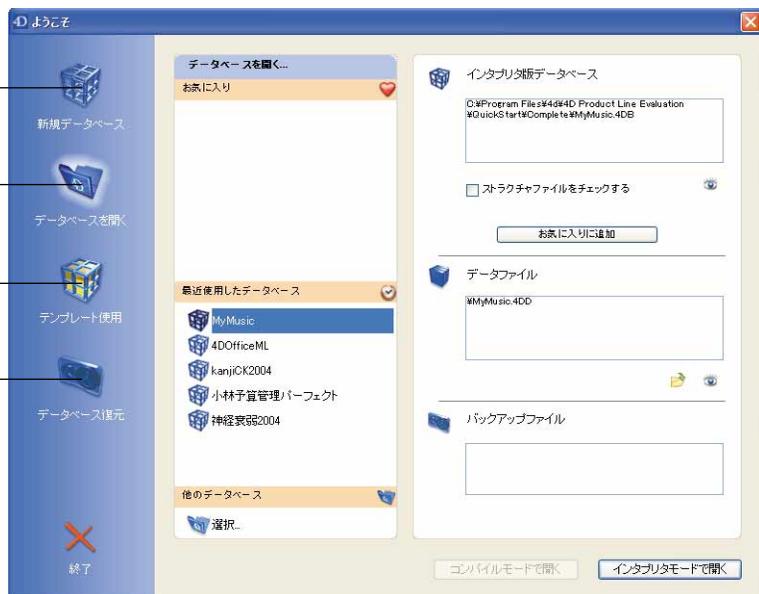
これらの新機能は、次のテーマ別に分けられています。

- 新しい「データベースを開く」ダイアログボックス
- リストボックスの管理
- 「デザイン」モード
- ランゲージ
- 4D Server
- 4D Tools
- 4D Write
- 4D View
- 4D Pack

新しい「データベースを開く」ダイアログボックス

データベースの作成、オープン、復元をより分かりやすくシンプルにするため、データベースを開く際のダイアログボックスがすべて作り変えられました。主要な機能（新規データベース、データベースを開く、テンプレート使用、データベース復元）へは、4つの異なるページを使用してアクセスします。

別のページへアクセスするボタン



このダイアログボックスにより提供されるオプションは、以前のバージョンと同じですが、「データベースを開く」ページで「お気に入り」を設定できるようになった点に注目してください（後述）。

お気に入り

「データベースを開く」ページでは、「お気に入り」を管理することができます。「お気に入り」には、選択したデータベースの名前がそのまま保存され、データベースへ簡単にアクセスすることができます。「最近使用したデータベース」と同じように、「お気に入り」にはそのデータベースの特性（パス名、各ファイル等）がすべて保存されます。

データベースを「お気に入り」に入れるには、そのデータベースが「最近使用したデータベース」エリアにリスト表示されていなければなりません。このリストからデータベースを選択し、ウインドウの上部右側にある「お気に入りに追加」ボタンをクリックします。また、このリストのコンテキストメニュー（データベース名の上で、Windowsなら右クリック、MacOSならControl+クリックする）の「お気に入りに追加」コマンドを使用することもできます。

「お気に入り」からデータベースを取り除くには、そのデータベースを選択し、「お気に入りから取り除く」ボタンをクリックするか、またはコンテキストメニューの「お気に入りから取り除く」コマンドを選択します。

リストボックスの管理

リストボックスオブジェクトの操作に関して、いくつかの変更が行われました。

- **SCROLL LINES** コマンドのサポート
- スクロールカーソルの位置の取得
- 3ステートチェックボックス
- スル日付の表示

SCROLL LINES

リストボックスオブジェクトに対して、**SCROLL LINES** コマンドを使用できるようになりました。このコマンドを使用すると、選択された最初の項目や特定の項目が表示されるように、リストの行がスクロールされます。

このコマンドをリストボックスに対して適用すると、コマンドは他のタイプのオブジェクトの場合と同じように動作します。ただ1つ異なる点は、引数<位置>が渡された場合に、コマンドはレコード番号ではなく、表示される行番号を示すところです。

▼ この命令は、100番目の行をリストボックスの先頭に表示します。

```
SCROLL LINES(*,"MyListbox";100;*)
```

スクロールカーソルの位置

Get listbox information 関数を使用して、スクロールバーの位置をプログラムから取得できるようになりました。

次の2つの定数を引数<情報>に渡すことができます。

- Position listbox hor scrollbar(6)
- Position listbox ver scrollbar(7)

この定数を渡すと、**Get listbox information** 関数はスクロールカーソルの元の位置に対しての位置、つまり、ウィンドウの隠れた部分のサイズ（ピクセル単位）を返します。デフォルトとして、この位置は0になります。

例えば、この値を行の高さに関する情報と組み合わせて使用すると、リストボックス上に表示された内容を調べることができます。

▼ あるリストボックスに、それぞれの高さが20ピクセルである行が納められているものとします。次の命令を実行します。

`$scroll:=Get listbox information(*;"リストボックス";Position listbox ver scrollbar)`

例えば、\$scrollに200が返された場合、11番目の行が現在リストボックスの最初の行として表示されていることがわかります（ $200/20=10$ 、従って10行分が隠れています）。

3 ステートチェックボックス

リストボックスオブジェクトにおいて数値を表わすために、3ステートチェックボックスを使用できるようになりました。

次の値を表示することができます。

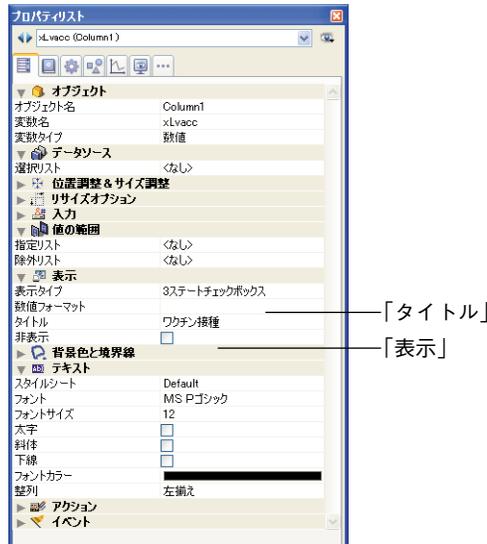
- 選択されていないチェックボックスは、値“0”を表わす。
- 選択されたチェックボックスは、値“1”を表わす。
- 塗りつぶされたチェックボックス（3番目の状態）は、他の値を表わす。

3ステートチェックボックスをデータ入力に使用することができます。クリックされるたびに、ボックスの状態（ステート）が変わります。この例では、塗りつぶされたチェックボックスには値“2”が割り当てられています。



リストボックスの列に3ステートチェックボックスを表示するには、プロパティリストの「表示タイプ」メニュー（「表示」テーマ）から「3ステートチェックボックス」オプションを選択します（列に関連付ける変数のタイプは数値タイプに設定しておかなければなりません）。

「タイトル」欄を使用し、チェックボックスに関連付けるラベルを指定することができます。



ヌル日付

以前のバージョンの4th Dimensionでは、リストボックス上でヌル日付は“00/00/00”と表示されました。今後、ヌル日付を含むセルには、空白が表示されます。

配列とリストボックスの同期化

リストボックスの内部管理が最適化され、列に関連付けた配列に関してプログラムで実行された変更が考慮されるようになりました（セレクション、データ、スタイル）。

今後、**INSERT LISTBOX ROW**や**DELETE LISTBOX ROW**などのコマンドを使用したり、関連付けた配列を変更する命令を実行すると、メソッド実行後にリストボックスオブジェクトが同期化（更新）されます。

「デザイン」モード

4th Dimension 2004.1の「デザイン」モードに次の変更が加えられました。

環境設定

データ入力時に自動トランザクションを使う

「データ入力時に自動トランザクションを使う」オプションは、今まで「環境設定」の「データベース／データ管理」ページに置かれていましたが、「アプリケーション／互換性」ページへ移されています。この陳腐化したオプションは、もう使用しないようにしてください。新規データベースにおいて、このオプションは表示されません。

読み込みのみモードでストラクチャファイルを開く際の警告を表示しない

現在のところは、データベースを開き、そのストラクチャファイルが読み込みのみ（リードオンリー）モードである場合に、4th Dimensionは警告ダイアログボックスを表示し、その旨を通知します。このデータベースを開くと、ストラクチャファイルを変更しても、その変更内容は保存されません。

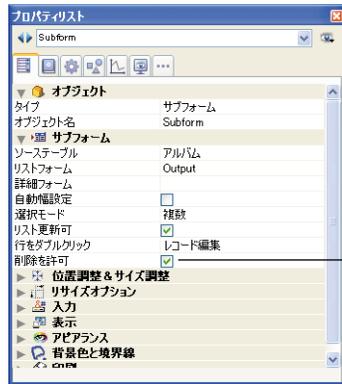
例えば、CD-ROMに保存された参照用データベースの場合など、特定の状況において、このダイアログボックスを表示したくない場合があります。これを実現するには、「環境設定」の「アプリケーション／アクセス」ページにある「読み込みのみモードでストラクチャファイルを開く際の警告を表示しない」オプションを選択します。

互換性上の理由から、バージョン2004.1より前の4th Dimensionで作成されたデータベースに対しては、このオプションがデフォルトで選択されています。

削除を許可（サブフォーム）

デフォルトとして、ユーザはDelキーまたはBackspaceキーを使用して、サブフォーム内のサブレコードを削除することができます。

しかし、特定のアプリケーション（例えば、ボタンをベースしたアプリケーション）では、この処理によりインタフェースの標準動作に支障が出る場合があるため、サブフォームのプロパティ（「サブフォーム」テーマ）に置かれた新しい「削除を許可」オプションを用いて、この問題を回避することができます。



標準の削除用キーを使用して削除するためのオプション

このオプションを選択しないと、キーボード上の削除キーを用いてサブレコードを削除できなくなります。

MacOS 上の「ファイル」メニューコマンド

MacOS Xのインタフェース標準に従い、ユーザがOptionキーを押した場合は、「ファイル」メニュー上に「すべて閉じる」と「すべて保存」コマンドが表示されるようになります。これらのコマンドにより、「閉じる：ウインドウ」と「保存：ウインドウ」コマンドが置き換えられます。



標準メニュー



Optionキーを使用した場合のメニュー

MacOS 上の F1 キー

開発者の要望に応えるために、MacOS版の4th Dimension 2004.1では、F1キーでオンラインヘルプを呼び出すための独自のシステムが変更されました。

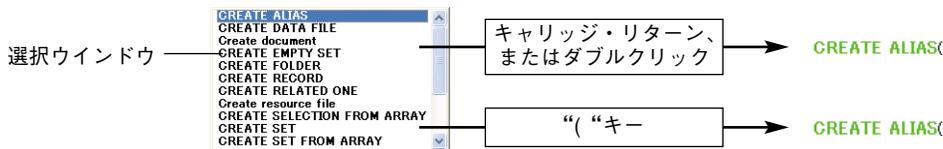
今後、「ユーザ」および「カスタム」モードにおいてユーザがF1キーを押すと、次のようになります。

- データベースストラクチャファイルと同じ階層に“データベース名.HTM”ファイルが存在する場合、4Dオンラインヘルプを管理する標準メカニズムが実装されます。このヘルプファイルは、デフォルトブラウザのウインドウ上に開かれます。

- 上記以外の場合、4th Dimensionは何も行わず、開発者は自由にこのイベントを処理することができます。つまり、独自のヘルプシステムを設置したり、または別の機能をF1キーに割り当てることができます。

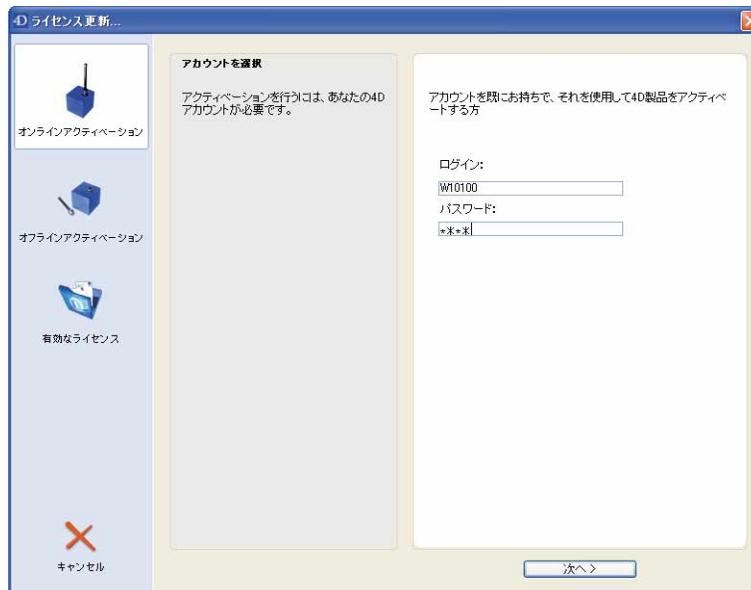
タイプahead機能

「メソッド」エディタのタイプahead機能を使用すると、コマンドの引数を手早く入力することができます。例えば、選択ウインドウの値を選んだ後で、“(”（開き括弧）や“;”（セミコロン）キーを押すと、挿入される値の後ろに開き括弧またはセミコロンが入力され、データを入力できる状態になります。



新しいライセンス管理ダイアログボックス

4th Dimension 2004.1では、ライセンスの管理や更新を行うダイアログボックスが新しくなりました。このダイアログボックスへは、「ヘルプ」メニューの「ライセンス更新...」コマンド（または、ランゲージコマンドの**CHANGE LICENSES**）を使用してアクセスすることができます。



このダイアログボックスを使用して、新しい4Dアプリケーションやプラグインのアクティベーション（有効化）を行い、有効なライセンスを表示することができます。このダイアログボックスに関する詳細は、4th Dimension 2004.1の『インストールガイド』を参照してください。

ランゲージ

ローカル変数へのポインタ

4th Dimension 2004.1では、ローカル変数へのポインタを使用することができます。次の命令文は、完全に有効となります。

```
$MyVar:="Hello world"  
MyPointer:=> $MyVar
```

ローカル変数へのポインタにより、ほとんどの場合でプロセス変数を保存できるようになります。

ローカル変数へのポインタは、同一プロセス内においてのみ使用可能です。

プロセス変数へのポインタと同じように、ポインタの使用時には、そのポインタの指す変数は必ず設定済みでなければなりません。ただし、ローカル変数を作成したメソッドの実行が完了すると、そのローカル変数は削除されてしまうということを覚えておいてください。

既に存在していない変数をポインタが呼び出すと、インタプリタモードではシンタックスエラー（変数が定義されていない）になりますが、コンパイルモードではより重大なエラーになる可能性があります。

デバッガ上でローカル変数へのポインタを表示し、このポインタが別のメソッドで宣言されたものである場合は、ポインタに続けてオリジナルのメソッド名が括弧内に示されます。

▼ 例えば、Method1で次のように記述した場合

```
$MyVar:="Hello world"  
Method2(->$MyVar)
```

Method2において、デバッガでは\$1が次のように表示されます。

```
$1->$MyVar(Method1)
```

\$1の値は次の通りです。

```
$MyVar(Method1) "Hello world"
```

注：RESOLVE POINTER コマンドは、ローカル変数へのポインタを取り扱いません。実際、同じ名前のローカル変数が別の場所に複数存在する可能性があるため、コマンドは正しい変数を見つけることができません。

SHOW ON DISK（新しいコマンド）

SHOW ON DISK(パス名{*}*)

引数	タイプ	説明
パス名	文字列	→ 表示する項目のパス名
*		→ 項目がフォルダである場合、その内容を表示する

テーマ：システムドキュメント

SHOW ON DISK コマンドは、引数<パス名>に渡されたパス名のファイルまたはフォルダをOSの標準ウインドウ上に表示します。

ユーザインタフェースでこのコマンドを使用すると、特定のファイルやフォルダの場所を指定することができます。

<パス名>にフォルダが指定された場合、デフォルトとしてコマンドはそのフォルダ自体の階層を表示します。任意の引数である<*>を渡すと、コマンドはそのフォルダを開き、ウインドウ上にフォルダ内容を表示します。<パス名>にファイルを指定した場合、引数<*>は無視されます。

コマンドが正常に実行されると、システム変数OKには1が代入されます。

▼ 次の例は、このコマンドの動作を表わしています。

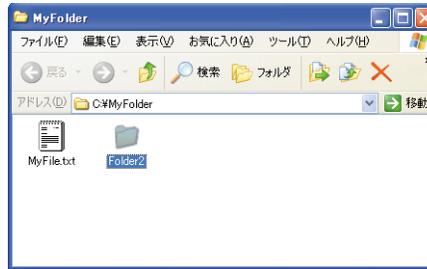
SHOW ON DISK("c:¥¥MyFolder¥¥MyFile.txt")

、指定されたファイルを表示する

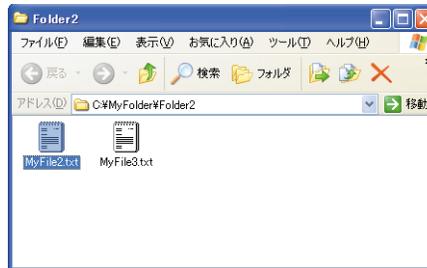


SHOW ON DISK("c:¥¥MyFolder¥¥Folder2")

、指定されたフォルダを表示する



SHOW ON DISK("c:¥¥MyFolder¥¥Folder2";*)
 `指定されたフォルダの内容を表示する



ピクチャ圧縮定数

より明確にするため、ピクチャの QuickTime 圧縮に関連する定数（QT Animation compressor、QT Video compressor など）は、新しいテーマ “Picture Compression（ピクチャ圧縮）” にまとめられました。今まで、これらの定数は “Standard System Signatures” テーマに置かれていました。

これらの定数は、**COMPRESS PICTURE FILE** コマンドなど、圧縮されたピクチャを生成する 4D コマンドで使用されます。

SET ENVIRONMENT VARIABLE

SET ENVIRONMENT VARIABLE(変数名; 変数値)

LAUNCH EXTERNAL PROCESS コマンドで使用するために、2つの新しい環境変数、**_4D_OPTION_CURRENT_DIRECTORY** と **_4D_OPTION_HIDE_CONSOLE** を利用できるようになりました。

カレントプロセスにおいて次に **LAUNCH EXTERNAL PROCESS** コマンドを呼び出す際に、これらの変数が有効になります。

■ **_4D_OPTION_CURRENT_DIRECTORY** : 起動する外部プロセスのカレントディレクトリを設定するために使用します。<変数名>には、このディレクトリのパス名を渡さなければなりません (MacOSではHFSタイプのシンタックス、WindowsではDOSタイプ)。

■ **_4D_OPTION_HIDE_CONSOLE** (Windowsのみ) : DOS コンソールのウインドウを隠すために使用します。<変数名>には、コンソールを隠す場合は“true”を、表示する場合は“false”を渡してください。

▼ Windowsでの使用例を次に示します。

```
SET ENVIRONMENT VARIABLE("_4D_OPTION_CURRENT_
                                DIRECTORY";"C:¥¥4D_VCS")
SET ENVIRONMENT VARIABLE("_4D_OPTION_HIDE_CONSOLE";"true")
LAUNCH EXTERNAL PROCESS("mycommand")
```

Get 4D folder

Get 4D folder{(フォルダ)} → 文字列

引数	タイプ	説明
フォルダ	倍長整数	→ フォルダタイプ (省略した場合=アクティブな4Dフォルダ)
戻り値	文字列	← 4Dフォルダのパス名

Get 4D folder 関数は、任意の引数<フォルダ>を受け入れるようになりました。この引数を使用し、取得しようとするパス名を持つ4D環境フォルダのタイプを指定します。

また、“4D環境” テーマにある次の定数のいずれかを引数<フォルダ>に渡すこともできます。

定数	値	説明
Active 4D Folder	0	アプリケーションのアクティブな4Dフォルダ (デフォルト値)
Licenses Folder	1	そのマシンのライセンスファイルを含むフォルダ
Extras Folder	2	各4D Clientマシンにダウンロードされた独自の内容を含むフォルダ
4D Client Database Folder	3	各4D Clientマシンにダウンロードされた4Dデータベースフォルダ

デフォルトとして (引数<パス名>を省略した場合)、コマンドはカレントアプリケーションのアクティブな4Dフォルダのパス名を返します (以前のバージョンの4th Dimensionと同じ)。

SET DATABASE PARAMETER、Get database parameter

SET DATABASE PARAMETER{テーブル;} セレクタ; 値)

Get database parameter{テーブル;} セレクタ)→倍長整数

これらのコマンドでは、いくつかの新しい定数を利用できるようになりました。

セレクタ	値	適用範囲
4D Server Log Recording	28	4D Server
Web Log Recording	29	4th Dimension、4D Server
Client Web Log Recording	30	すべての4D Clientマシン
Table Sequence Number	31	4D アプリケーション

■ セレクタ=28 (4D Server Log Recording : 4D Server ログを記録する)

■ 有効な値 : 0 または 1 (0=記録しない、1=記録する)

■ 説明 : 4D Server が受信する標準リクエスト (Web リクエストを除く) の記録を開始、または停止します。デフォルト値は 0 (リクエストを記録しない) です。

4D Server では、サーバマシンが受け取る各リクエストをログファイルに記録することができます。このメカニズムが有効になると、ログファイルがデータベースストラクチャファイルと同じ階層に作成されます。ログファイル名は“4DRequestsLog”で始まります。このテキストファイルには、各リクエストに関するさまざまな情報 (時刻、プロセス番号、ユーザ名、リクエストのサイズ、プロセスの継続時間など) が単純な表形式で保存されます。アプリケーションの微調整を行う場合や、統計を取る場合には、これらの情報が非常に役立ちます。この情報は、例えば、スプレッドシートソフトウェアへ書き出して処理することができます。

このファイルのサイズはすぐに大きくなる可能性があるため、一定期間に限ってこのオプションを有効にすることをお勧めします。

■ セレクタ=29 (Web Log Recording : Web ログを記録する)

■ 有効な値 : 0 または 1 (0=記録しない、1=記録する)

■ 説明 : 4th Dimension または 4D Server の Web サーバが受信する Web リクエストの記録を開始、または停止します。デフォルト値は 0 (リクエストを記録しない) です。

Web リクエストのログは、“logweb.txt” という名前のテキストファイルに保存され、自動的にデータベースストラクチャファイルと同じ階層に置かれます。このファイルのフォーマットは CLF (Common LogFile Format) または NCSA 形式ですが、これは大部分の Web サイト解析ツールにより認識されます。

また、4th Dimensionの「環境設定」の「Web／詳細」ページからこのファイルを有効にすることも可能です。

■ セレクタ=30 (Client Web Log Recording : クライアントのWebログを記録する)

■ 有効な値 : 0 または 1 (0=記録しない、1=記録する)

■ 説明 : すべてのクライアントマシンのWebサーバが受信するWebリクエストの記録を開始、または停止します。デフォルト値は0 (リクエストを記録しない) です。

このセレクタの働きはセレクタ29と同じですが、Webサーバとして使用されるすべての4D Clientマシンに対して適用されます。特定のクライアントマシンに対してのみ値を設定したい場合は、4D Clientの「環境設定」ダイアログボックスを使用してください。

■ セレクタ=31 (Table Sequence Number : テーブルの順次番号)

■ 有効な値 : あらゆる倍長整数値

■ 説明 : このセレクタは、引数として渡した<テーブル>のレコードのユニークなカレント番号を変更、または取得します。“カレント番号”とは、“最後に使用された番号”のことです。**SET DATABASE PARAMETER** コマンドでこの値を変更すると、その値に1を加算した番号で次のレコードが作成されます (Sequence number関数からは、この新しい番号が返されます)。

デフォルトでは、4th Dimensionによりこのユニークな番号が設定され、レコードの作成順に対応しています。詳細については、Sequence number関数の説明を参照してください。

BUILD APPLICATION

BUILD APPLICATION({プロジェクト名})

今後、任意の引数である<プロジェクト名>を省略すると、コマンドは標準のファイルオープン用ダイアログボックスを表示し、そこでプロジェクトファイルを指定できるようになります。

このダイアログボックスを確定すると、システム変数 Document には開かれたプロジェクトのフルパス名が代入されます。

32KB を超える SOAP 引数を返す

4th Dimensionにより公開されたWebサービスでBLOBを使用することにより、サイズが32KBを超えるSOAP引数を返すことが可能になりました。

以前のバージョンの4th Dimensionでは、テキスト引数しか使用できなかったため、SOAPサーバの応答は32KBに制限されていました。

この制限を大きくするには、**SOAP DECLARATION** コマンドをコールする前に、変数をBLOBタイプとして宣言します。

例えば

C_BLOB(\$0)

SOAP DECLARATION(\$0; Is Text; SOAP Output)

タイプが“Is Text (Is BLOBではない)”である点に注意してください。これにより、引数が正しくフォーマットされます。

注：クライアント側で、4th Dimensionを使用してこのタイプのWebサービスへサブスクライブした場合、当然ながらWebサービスウィザードによりテキストタイプの変数が生成されます。これを使用可能にするには、戻り値となるこの変数をCOMPILER_WEBメソッドでBLOBタイプとして再定義する必要があります。

“データフォーク”のリソースを管理する

マルチプラットフォーム環境での開発を円滑に行うため、リソースの保管に各ファイルのデータフォークを使用することをお勧めします。これは、リソースフォークがMacOS固有の機能であるためです。

リソースファイルの管理に用いられるコマンドは、リソースファイルのデータフォークを使用できるように変更されています。

Open resource file

Open resource file 関数には、次のメカニズムが導入されています。

引数として渡されたファイルのリソースフォークが空である場合、この関数はファイルのデータフォークをオープンし、そこに保管されたリソースへアクセスします。つまり、データフォークに保存されている任意のリソースを使用することができます。

Create resource file

Create resource file(リソースファイル名{; ファイルタイプ{; *})→文書参照番号

Create resource file 関数は、任意の引数であるアスタリスク (*) を受け入れるようになりました。

■ この引数を省略すると、関数はファイルのリソースフォークの作成とオープンを行います（以前のバージョンの4th Dimensionと同じ）。

■ この引数を渡した場合、関数はファイルのデータフォークの作成とオープンを行います。

XML 文書を開く

SAXから読み込まれる予定である文書は、Open document関数を使用して強制的に“リードオンリー（読み込みのみ）”モードでオープンしなければなりません。これは、“標準”文書とXML文書を両方同時に開く場合に、4th DimensionとXercesライブラリ間でのあらゆるコンフリクトを避ける必要があるからです。

リード/ライト（読み書き）モードでオープンされた文書を使用して、SAX解析コマンドを実行すると、警告が表示されて解析は行えません。

▼ SAX解析の対象となる文書のオープンは、次のように実行しなければなりません。

```
DocRef:=Open document("","XML";Read Mode)
`... SAX解析コマンド
```

SET DICTIONARY

SET DICTIONARY(辞書)

SET DICTIONARY コマンドは、引数<辞書>の値として、「Dictionaries」テーマの定数に指定された値以外にも、多様な数値を受け入れます。これらの値は、4つの主要言語それぞれの各種バリエーションに対応します。バリエーションをロードするには、直接その値を引数<辞書>に渡します。

辞書	値
English(United Kingdom)：英語（英国）	65536
English Irish(Ireland)：アイルランド英語（アイルランド）	65600
English Australian(Australia)：オーストラリア英語（オーストラリア）	65664
English of New Zealand：ニュージーランド英語	65680
English American(USA)：アメリカ英語（米国）	65792
English Canadian(Canada)：カナダ英語（カナダ）	65920
English South African(South Africa)：南アフリカ英語（南アフリカ）	66048
English West Indian(Caribbean)：西インド諸島英語（カリブ）	66176
English Jamaican(Jamaica)：ジャマイカ英語（ジャマイカ）	66192
<i>English(United Kingdom+ America)：英語（英国+米国）</i>	<i>69632</i>
German standard(Germany, old spelling)：標準独語（ドイツ、旧スペル）	131072

辞書	値
German of Luxembourg : ルクセンブルグ独語	131073
German of Austria : オーストリア独語	131088
German of Liechtenstein : リヒテンシュタイン独語	131089
German of Switzerland(old spelling) : スイス独語 (旧スペル)	131104
German of South Tyrol : 南チロル独語	131120
German New spelling : 独語新スペル	131328
German of Switzerland New spelling : スイス独語新スペル	131360
<i>German Old and New spelling</i> : 独語新旧スペル	131584
German of Switzerland Old and New spelling : スイス独語新旧スペル	131616
<i>Spanish standard(Spain)</i> : 標準スペイン語	196608
Spanish of Latin America standard : 中南米標準スペイン語	196864
Spanish Argentinean(Argentina) : アルゼンチンス페인語 (アルゼンチン)	196865
Spanish Bolivian(Bolivia) : ボリビアスペイン語 (ボリビア)	196866
Spanish Chilean(Chile) : チリスペイン語 (チリ)	196867
Spanish Columbian(Columbia) : コロンビアスペイン語 (コロンビア)	196868
Spanish Cuban(Cuba) : キューバス페인語 (キューバ)	196869
Spanish Costa Rican(Costa Rica) : コスタリカスペイン語 (コスタリカ)	196870
Spanish Dominican(Dominican Rep.) : ドミニカスペイン語 (ドミニカ共和国)	196871
Spanish Ecuadorian(Ecuador) : エクアドルスペイン語 (エクアドル)	196872
Spanish Guatemalan(Guatemala) : グアテマラス페인語 : (グアテマラ)	196873
Spanish Honduran(Honduras) : ホンジュラススペイン語 (ホンジュラス)	196874
Spanish Mexican(Mexico) : メキシコス페인語 (メキシコ)	196875
Spanish Nicaraguan(Nicaragua) : ニカラグアスペイン語 (ニカラグア)	196876
Spanish Panamanian(Panama) : パナマスペイン語 (パナマ)	196877
Spanish Paraguayan(Paraguay) : パラグアイスペイン語 (パラグアイ)	196878
Spanish Peruvian(Peru) : ペルース페인語 (ペルー)	196879
Spanish Puerto Rican(Puerto Rico) : プエルトリコス페인語 (プエルトリコ)	196880
Spanish Salvadorian(El Salvador) : エルサルバドルスペイン語 (エルサルバドル)	196881
Spanish Uruguayan(Uruguay) : ウルグアイスペイン語 (ウルグアイ)	196882
Spanish Venezuelan(Venezuela) : ベネズエラス페인語 (ベネズエラ)	196883
Spanish Guinean(Equatorial Guinea) : ギニアスペイン語 (ギニア)	197121
<i>France, Monaco, Valle d'Aosta</i> : フランス、モナコ、ヴァッレ・ダオスタ州	262144
Canada : カナダ	262160
Louisiana : ルイジアナ	262161
Belgium : ベルギー	262176
Luxembourg : ルクセンブルグ	262177
Switzerland : スイス	262192
Martinique, Guadeloupe, Haiti, Guyana : マルチニク、グアドループ、ハイチ、ギアナ	262208

辞書	値
Reunion, Seychelles, Comoro, Mauritius : レユニオン、セイシェル、コモロ、モーリシャス	262224
Tahiti, New Caledonia, Vanuatu, etc. : タヒチ、ニューカレドニア、バヌアツ等	262240
Morocco, Algeria, Tunisia : モロッコ、アルジェリア、チュニジア	262256
French African standard : アフリカ標準仏語	262272
Benin : ベニン	262273
Burkina Faso : ブルキナファソ	262274
Burundi : ブルンジ	262275
Cameroon : カメルーン	262276
Central African Republics : 中央アフリカ共和国	262277
Congo(Brazzaville) : コンゴ (ブラザビル)	262278
Democratic Republic of Congo(ex-Zaire) : コンゴ民主共和国 (前ザイール)	262279
Ivory Coast : コートジボワール	262280
Djibouti : ジブチ	262281
Gabon : ガボン	262282
Guinea : ギニア	262283
Mauritania : モーリタニア	262284
Niger : ニジェール共和国	262285
Rwanda : ルワンダ	262286
Senegal : セネガル	262287
Chad : チャド	262288
Togo : トーゴ	262289

注：イタリック体で表わされた名称は、定数使用時にインストールされる標準辞書です。

SET LIST PROPERTIES、GET LIST PROPERTIES (誤り)

SET LIST PROPERTIES (リスト; 表示形式{; アイコン{; 線の高さ{; ダブルクリック{; 複数選択可{; 編集可}}}}))

GET LIST PROPERTIES (リスト; 表示形式{; アイコン{; 線の高さ{; ダブルクリック{; 複数選択可{; 編集可}}}}))

以前のバージョンのドキュメントにおいて、新しい引数である<編集可>の説明が正しくありませんでした。

正しい説明は、次の通りです。

“0= リスト編集不可、1= リスト編集可 (デフォルト値)”

この逆の説明が行われていました。『ランゲージリファレンス』マニュアルではこの誤りが訂正されています。

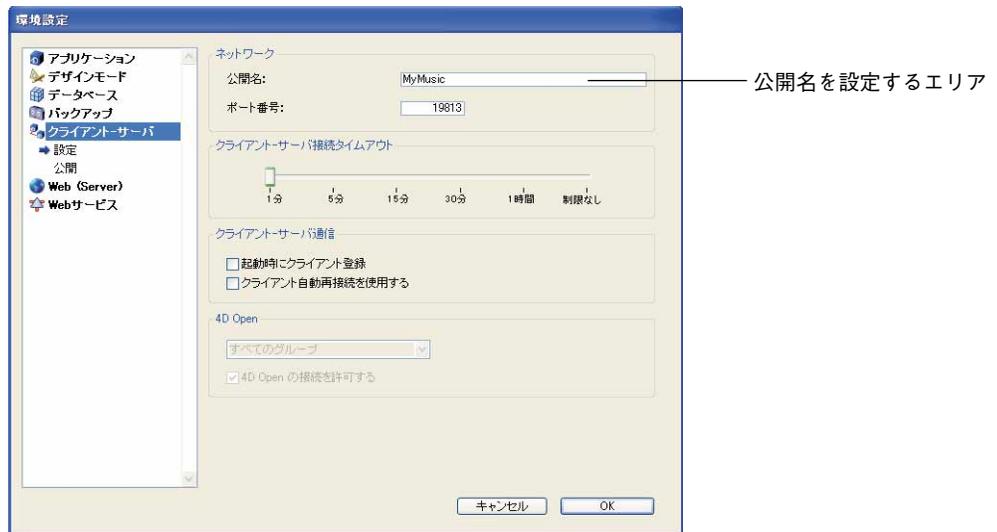
4D Server

2つの新機能である公開名と「Extras」フォルダは、特にクライアント／サーバ版の4th Dimension に関係します。

公開名

4D Serverが公開するデータベースの公開名を変更できるようになりました。これは、接続用ダイアログボックスのダイナミックTCP/IP公開ページ上に表示される名前です。

このパラメータは、アプリケーションの「環境設定」ダイアログボックスの「クライアント／サーバ／設定」ページに置かれています。



デフォルトとして、4D Serverはデータベースのストラクチャファイル名を使用します。独自の名前を自由に入力することが可能です。

注：独自のクライアント／サーバアプリケーションの範囲内では、このパラメータが考慮されません。原則として、クライアントアプリケーションは直接サーバアプリケーションへ接続し、接続用ダイアログボックスを使用しません。しかし、エラーが発生した場合は、このダイアログボックスが表示されます。この際、サーバアプリケーションの公開名はコンパイル済みデータベースの名前になります。

「Extras」フォルダ

クライアント／サーバ構成において、サーバのカスタムアイテム（リソースファイル、テキストドキュメント、XML参照ファイル等）をクライアントマシンへ転送するために、特定のフォルダを使用できるようになりました。「Extras」フォルダと呼ばれるこのフォルダは、ストラクチャファイルと同じ階層に配置しなければなりません。

4D Serverはこのフォルダへの変更を自動管理し、必要な項目だけを転送します。さらに、ネットワーク上のコピー時間を最適化するため、フォルダ内容が圧縮されます。

重要：クライアントマシン側の接続初期化フェーズに著しく時間を要するため、大量のデータを転送しないようにしてください。

クライアント側では、他のストラクチャ要素と同じ場所へ「Extras」フォルダがダウンロードされます。この場所は次の通りです。

システム	クライアントマシン上にダウンロードされたフォルダの場所
Mac OS X	{Disk}:Library:Application support:4D:データベース名_アドレス:Extras
Windows 2000 Windows XP	{Disk}:¥Documents and Settings¥ユーザ名¥Application Data¥4D¥データベース名_アドレス:Extras

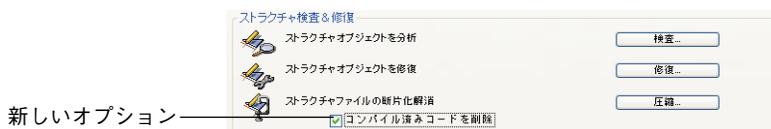
各クライアントマシン上で、元の階層構造が再作成されます。

4D Tools

コンパイル済みコードを削除

4D Toolsの「ストラクチャ」ページにおいて、ストラクチャの圧縮機能に関連する新しいオプション（「コンパイル済みコードを削除」）が利用できるようになりました。

このオプションはストラクチャファイルにコンパイル済みコードが含まれている場合のみ有効になります。このオプションを選択すると、ファイル内のコンパイル済みコードの削除とともに圧縮が行われます。



場合によっては、このオプションにより、かなりの量のスペースが得られます。

4D Write

4D Write エリアのタイマー

4D Write 2004.1では、“タイマー”タイプのイベントに関して特定の処理を行えるようになります。これにより、4D Writeエリアで定期的にアクションを実行することができます。

より具体的に言うと、タイマーは、編集されるドキュメントに対して自動バックアップセキュリティメカニズムを導入するために使用されます。

このイベントは、次のコマンドを用いて処理することができます。

■ **WR ON EVENT** (新しいイベントである wr on timer)

■ **WR SET AREA PROPERTY** (新しいエリアプロパティである wr timer frequency)

「wr on timer」イベントのインストール

WR ON EVENT コマンドを使用して、プロジェクトメソッドを「wr on timer」イベントに基づいて実行できるようになりました。その名が示す通りに、一度このイベントをインストールすると、ユーザのアクションとは関係なく、X tick 毎にこのイベントが生成されます (1 tick は 60/1 秒です)。

次の命令を実行した場合、

```
WR ON EVENT(myArea; wr on timer;"myMethod")
```

myArea がオープンされている間は、メソッド myMethod が一定間隔で実行されます (デフォルトでは 60 秒ごと)。myArea に 0 を渡すと、このメカニズムがすべての 4D Write エリアに適用されます。**WR SET AREA PROPERTY** コマンドを使用すると、メソッドを呼び出す間隔を変更することができます。

注：この定数は、WR Get on event method 関数で使用することができます。

呼び出されるメソッドでは、あまり大量の処理を実行しないでください。このような処理を繰り返し実行すると、アプリケーション処理速度が著しく低下する可能性があるため注意が必要です。

呼び出し間隔を変更する

新しい「wr timer frequency」定数を使用して、任意のエリアで「wr on timer even」イベントが生成される間隔を変更することができます。この定数は、**WR SET AREA PROPERTY** コマンドとともに使用しなくてはなりません。

タイマーの間隔を変更するには、次の命令を実行します。

WR SET AREA PROPERTY(myArea;wr timer frequency;myvalue)

倍長整数タイプである myvalue の値により、そのエリアに関する呼び出し間隔を tick 単位で指定することができます (1 tick は 60/1 秒です)。デフォルトの呼び出し間隔は 3600 ticks、つまり 60 秒です。

注：この定数は、WR GET AREA PROPERTY コマンドで使用することができます。

アプリケーションの処理速度を著しく低下させないように、このイベントを頻繁に生成しすぎてはならないという点に注意してください。

4D View

バージョン 2004.1 の 4D View には、いくつかの新機能が導入されています。

- 新しいイベント “contextual click”
- 印刷用の新しいプロパティとコマンド (**PV Print settings to blob** および **PV BLOB TO PRINT SETTINGS**)
- **PV SET COMMAND STATUS** コマンドの値 “0”
- ユーザが利用可能な、新しいセルフフォーマットオプション
- 新しい「詳細プロパティ」ダイアログボックス

pv on contextual click イベント

4D View 2004.1 は、新しいイベントである 「pv on contextual click」 を処理します。このイベントは、ユーザがマウスの右ボタンをクリック (Windows) するか、または Control+クリック (MacOS) すると生成されます。

「pv on contextual click」 イベントは、従来の 「pv on right clicked」 イベントと似ていますが、このイベントはユーザがマウスボタンを放した時点で呼び出されるのに対して、「pv on right clicked」 イベントはボタンが押された時点で呼び出されます。

これら2つのイベントを使用して、ポップアップ形式のコンテキストメニューを設置することができます。「pv on contextual click」 イベントは Windows の操作により適合し、「pv on right clicked」 イベントは MacOS での操作に対応しています。2つのイベントは、同時に使用することができます。

イベントの処理は、**PV ON EVENT** コマンドを使用して実行します。このコマンドは、引数<イベント>に新しい「pv on contextual click (値15)」定数を受け入れます。

このイベントが生成されると、**PV ON EVENT** コマンドにより呼び出されたメソッドでは、\$4と\$5へ次の値が返されます。

- セルでコンテキストクリックが行われると、\$4にはカラム番号が、\$5には行番号が返されます。
- 行ヘッダでコンテキストクリックが行われると、\$4には0が、\$5には行番号が返されます。
- カラムヘッダでコンテキストクリックが行われると、\$4にはカラム番号が、\$5には0が返されます。
- エリアの左上端でコンテキストクリックが行われると、\$4と\$5には0が返されます。

新しい印刷プロパティ

PV SET PRINT PROPERTY コマンドと **PV Get print property** 関数において、新しい印刷プロパティを使用することができます。これらのプロパティを用いて、ローレベルの印刷プロパティを設定します。

PV SET PRINT PROPERTY、**PV Get print property**

PV SET PRINT PROPERTY(エリア; プロパティ; 値{; 値2})

引数	タイプ	説明
エリア	倍長整数	→ 4D View エリア
プロパティ	倍長整数	→ プロパティ番号
値	倍長整数	→ プロパティの値1
値2	文字列	→ プロパティの値2

PV Get print property(エリア; プロパティ{; 値2})→倍長整数

引数	タイプ	説明
エリア	倍長整数	→ 4D View エリア
プロパティ	倍長整数	→ プロパティ番号
値2	文字列	← プロパティの値2
戻り値	倍長整数	← プロパティの値

PV SET PRINT PROPERTY コマンドおよび **PV Get print property** 関数には、文字列タイプの引数 (任意) が追加され、後述する特定の新しい印刷プロパティとともにこの引数を使用することができます。

新しい定数は、「PV Print properties」テーマに納められています。また、新しい値定数が「PV Print values」テーマに追加されました。

pv print paper width、pv print paper height

これらの定数は以前のバージョンの4D Viewにもありましたが、読み込み目的にのみ使用可能でした（**PV Get print property** 関数）。バージョン2004.1以降、これらの定数は**PV SET PRINT PROPERTY** コマンドを用いて書き込む目的にも使用できるようになります。

関連する値：幅と高さ（ピクセル単位）

pv print scale

現行の印刷倍率の設定や取得を行うために使用します。ただし、プリンタのなかには、倍率を変更できないものがある点に留意してください。無効な値を渡すと、印刷時にこのプロパティが100%にリセットされます。

関連する値：印刷倍率

pv print number copies

印刷部数の設定や取得を行うために使用します。

関連する値：部数（デフォルト値は1）

pv print paper source

使用する用紙トレイの設定や取得を行うために使用します。関連する値：使用する用紙トレイのインデックスに相当する数値であり、4Dの**PRINT OPTION VALUES** コマンドから返されるトレイ配列に納められた値。

注：このプロパティはWindows上でのみ使用可能です。

pv print color

カラー処理用のモードの設定や取得を行うために使用します。このプロパティは、カラープリンタにのみ有効です。関連する値：「PV Print values」テーマの定数。

■ **pv black and white**：黑白印刷（モノクロ）。

■ **pv color**：カラー印刷。

注：このプロパティはWindows上でのみ使用可能です。

pv print destination

印刷先の設定や取得を行うために使用します。

関連する値：“pv destination” で始まる「PV Print values」テーマの定数

- **pv destination printer**：印刷ジョブはプリンタへ送信されます。
- **pv destination file (Windowsのみ)**：印刷ジョブはファイルへ送信されます。この定数を使用すると、<値2>には出力結果であるドキュメントのパス名が代入されます。<値2>に空の文字列を渡すか、または省略した場合は、印刷時にファイル保存用のダイアログボックスが表示されます。
- **pv destination PDF file (MacOSのみ)**：印刷ジョブはPDFファイルへ送信されます。この定数を使用すると、<値2>には出力結果であるPDFファイルのパス名が代入されます。<値2>に空の文字列を渡すか、または省略した場合は、印刷時にファイル保存用のダイアログボックスが表示されます。
- **pv destination EPS file (MacOSのみ)**：印刷ジョブはEPSファイルへ送信されます。この定数を使用すると、<値2>には出力結果であるEPSファイルのパス名が代入されます。<値2>に空の文字列を渡すか、または省略した場合は、印刷時にファイル保存用のダイアログボックスが表示されます。

pv print double sided

片面印刷、または両面印刷を行うために使用します。

関連する値：pv value on または pv value off 定数

- **pv value on**：両面印刷
- **pv value off**：片面印刷（デフォルト値）

注：このプロパティはWindows上でのみ使用可能です。

pv print binding

印刷が両面印刷モード（前述）で行われる場合に、とじしろ位置の設定や取得を行うために使用します。

関連する値：「PV Print values」テーマの定数

- **pv left binding**：左とじ（デフォルト値）
- **pv top binding**：上とじ

注：このプロパティはWindows上でのみ使用可能です。

pv print document name

スプーラ文書の一覧に表示しなければならない印刷文書名の設定や取得を行うために使用します。この定数を使用すると、<値2>には印刷文書の名前が代入されます。<値>には0を渡してください。

標準操作（“4D View” という名前を使用する）を使用、または復帰するには、<値2>に空の文字列を渡します。

pv print pages from

印刷を開始するページ番号の設定や取得を行うために使用します。

関連する値：ページ番号

pv print pages to

印刷を終了するページ番号の設定や取得を行うために使用します。

関連する値：ページ番号

例題

Windows上で用紙トレイ（用紙ソース）の選択を行う例を次に示します。

```
ARRAY TEXT($arrNames;0)
ARRAY LONGINT($arrInfo1;0)
    `使用可能なトレイ一覧を取得する
PRINT OPTION VALUES(Paper source;$arrNames;$arrInfo1)
```

例えば、配列 \$arrNames と \$arrInfo1 に取り出すことができる値を次に示します。

\$arrNames	\$arrInfo1	インデックス
自動選択	15	1
トレイ1	257	2
トレイ1（手動）	258	3
トレイ2	259	4
トレイ3	260	5
トレイ4	261	6
封筒フィーダー	262	7

“トレイ1（手動）”を使用したい場合は、このトレイに対応するインデックスを渡します。

PV SET PRINT PROPERTY(area; pv print paper source;3)

新しい印刷コマンド

4D Viewにおいて、プログラムから印刷設定を完全に管理するため、2つの新しいコマンド、**PV Print settings to blob**と**PV BLOB TO PRINT SETTINGS**（「PV Printing」テーマ）を使用できるようになりました。

PV Print settings to blob

PV Print settings to blob(エリア)→BLOB

引数	タイプ	説明
エリア	倍長整数	→ 4D View エリア
戻り値	BLOB	← 印刷設定を含むBLOB

PV Print settings to blob関数は、4D View<エリア>の現在の印刷設定をBLOBに格納します。

このBLOBには、印刷に使用される設定がすべて納められます。

- レイアウトパラメータ（用紙、方向、倍率）
- 印刷パラメータ（部数、用紙ソース等）

一方、この関数では、「印刷オプション」ダイアログボックス上に現れる4D Viewプラグイン固有の印刷オプションは格納されません（ページヘッダ、行の繰り返し等）。

この関数を使用すると、プリンタの機種や利用可能な印刷設定を問わず、4D Viewエリアの印刷設定を保存することができます。返されるBLOBの内容をプログラムから変更してはいけません。このBLOBは**PV BLOB TO PRINT SETTINGS**コマンドでのみ使用することができます。

例えば、**PV SET PRINT PROPERTY**コマンドを使用してオプションを一時的に変更する前に、**PV Print settings to blob**関数を用いて現在の印刷設定を保存しておくことができます。印刷が終了した後に、**PV BLOB TO PRINT SETTINGS**コマンドを使用して、現在のパラメータへ戻すことができます。

参照：PV BLOB TO PRINT SETTINGS

PV BLOB TO PRINT SETTINGS

PV BLOB TO PRINT SETTINGS(エリア; 印刷設定)

引数	タイプ	説明
エリア	倍長整数	→ 4D View エリア
印刷設定	BLOB →	印刷設定を含む BLOB

PV BLOB TO PRINT SETTINGS コマンドは、BLOBである<印刷設定>に納められた値で、4D View <エリア>の現在の印刷設定を置き換えます。このBLOBは、**PV Print settings to blob** 関数を用いて生成しておかなければなりません。

引数<印刷設定>には、印刷に使用されるすべての設定が納められます。

■ レイアウトパラメータ（用紙、方向、倍率）

■ 印刷パラメータ（部数、用紙ソース等）

注：Windows と MacOS では、印刷設定のフォーマット方法が異なります。この結果、2つのプラットフォーム間においてBLOB <印刷設定>の互換性が保証されません。

BLOB <印刷設定>に有効な印刷設定が含まれない場合、コマンドはエラーを返します。

PV SET CELL NAME

PV SET CELL NAME(エリア; カラム; 行; 名前{; モード})

引数	タイプ	説明
エリア	倍長整数	→ 4D View エリア
カラム	倍長整数	→ セルのカラム番号
行	倍長整数	→ セルの行番号
名前	文字列	→ セルの名前
モード	倍長整数	→ 0 または省略=名前を追加 → 1=名前を置換

PV SET CELL NAME コマンドには、任意の引数である<モード>が追加されました。この引数を使用し、セルに1つ以上の名前が既に付けられている場合に、新しいセル名の設定方法を指定します。「PV Cell properties」テーマの定数を使用することができます。

■ <モード>を省略するか、または「pv add name」定数（値“0”）を渡した場合、そのセルに対して既に設定されている名前へ新しい名前が追加されます。

■ <モード>に「pv replace name」定数（値“1”）を渡した場合、そのセルに対して既に設定されている名前は新しい名前です置き換えられます。

セルに関連付けられた名前をすべて削除するには、<名前>に空の文字列（""）を、<モード>には「pv replace name」定数を渡してください。

PV SET COMMAND STATUS の値 0

PV SET COMMAND STATUS コマンドは、引数<コマンド>（コマンド番号）の値として0を受け入れるようになりました。これを指定すると、このコマンドは4D Viewエリアの全コマンドのステータスを変更します。

“自動ワードラップ” と “強制テキスト” オプション

「フォーマット」と「スタイルシート」ダイアログボックス上で、2つのオプションが新たに利用できるようになります。

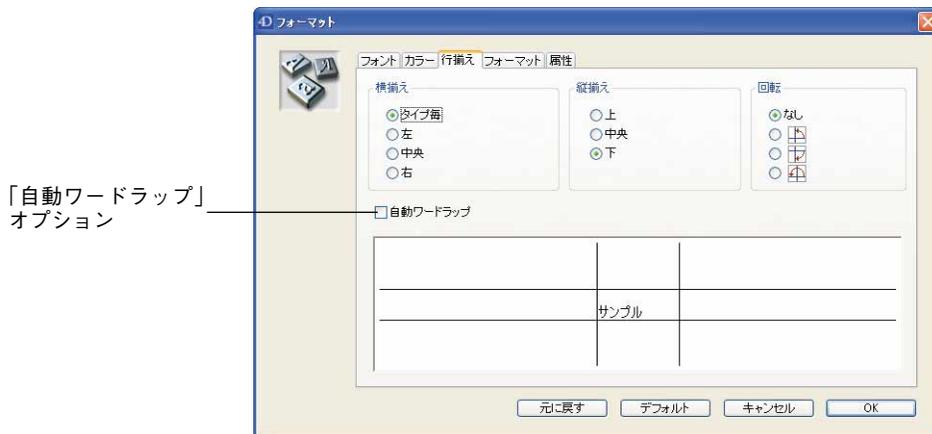
■ 自動ワードラップ

■ 強制テキスト

以前のバージョンの4D Viewにおいて、これら2つのオプションはプログラム上でのみ利用可能でした。

自動ワードラップ

このオプションにより、セル内容がセルの幅に収まらない場合に、その内容が自動的にワードラップされます。



デフォルトとして、この機能が無効である場合はセルの内容が隣のセルへはみ出します。

注：プログラム上で、このオプションは「pv style automatic word wrap」定数により利用することができます。また、このオプションはPV SET CELL PROPERTY、PV SET RANGE PROPERTY、PV SET STYLE PROPERTY コマンドで使用できます。

強制テキスト

このオプションを使用し、強制的にセルを生テキストとして表示します。生テキストとは、セル内容に応じたフォーマット（数値、日付、テキストなど）が4D Viewにより行なわれていないテキストのことです。



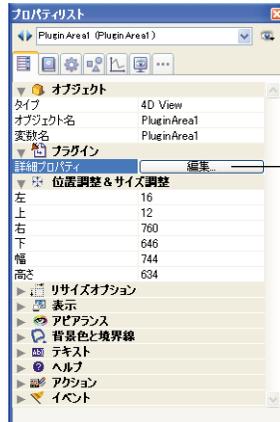
このオプションを選択すると、すべての表示フォーマット用ポップアップメニューが無効になります（テキストタイプデータを除く）。

注：プログラム上で、このオプションは、「pv style format forced text」定数により利用することができます。また、このオプションはPV SET CELL PROPERTY、PV SET RANGE PROPERTY、PV SET STYLE PROPERTY コマンドで使用できます。

「詳細プロパティ」ダイアログボックス

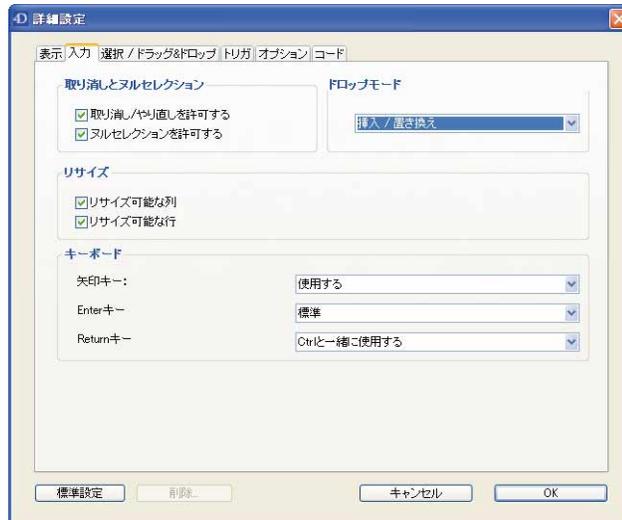
4D View エリアに組み込まれているプロパティであり、**PV SET AREA PROPERTY** コマンドや **PV Get area property** 関数を用いて設定可能なプロパティはすべて、「フォーム」エディタの「詳細プロパティ」ダイアログボックスを使用して直接アクセスできるようになりました。

このダイアログボックスを表示するには、「フォーム」エディタ上で4D View エリアを選択し、「プロパティリスト」の「プラグイン」テーマにある「編集...」ボタンをクリックします。



「詳細プロパティ」ダイアログボックス
へアクセスするボタン

次のダイアログボックスが表示されます。



各オプションや入力エリアを使用し、エリアのプロパティを指定することができます。このダイアログボックスは複数ページで構成されています。最後のページを使用すると、このダイアログボックスの設定内容に対応する4D Viewコードを作成することができます。

- 「標準設定」ボタンを使用すると、現在のページのプロパティがすべてデフォルト値にリセットされます。
- 「削除...」ボタンを使用すると、このダイアログボックスを用いてエリアに設定した詳細プロパティがすべて削除されます。このボタンは、プログラムから設定されたプロパティには影響を与えないという点に留意してください。

「詳細プロパティ」ダイアログボックスが少なくとも一度は確定されていないと、このボタンはグレー表示されます。

■「キャンセル」ボタンにより、すべての変更がキャンセルされ、ダイアログボックスが閉じられます。

■「OK」ボタンにより、すべての変更が有効になり、ダイアログボックスが閉じられます。

このダイアログボックスのパラメータについては、次の説で説明します。特定のメカニズムに関する詳細は、4D View『ランゲージリファレンス』マニュアルの **PV SET AREA PROPERTY** コマンドの説明を参照してください。

「表示」ページ

このページのオプションを使用して、エリアのインタフェース要素の表示を決定します。



■ ツールバー、スクロールバー、グリッド：このオプションを使用し、エリアの対応する項目を表示（選択されたオプション）または非表示（選択されていないオプション）に設定します。

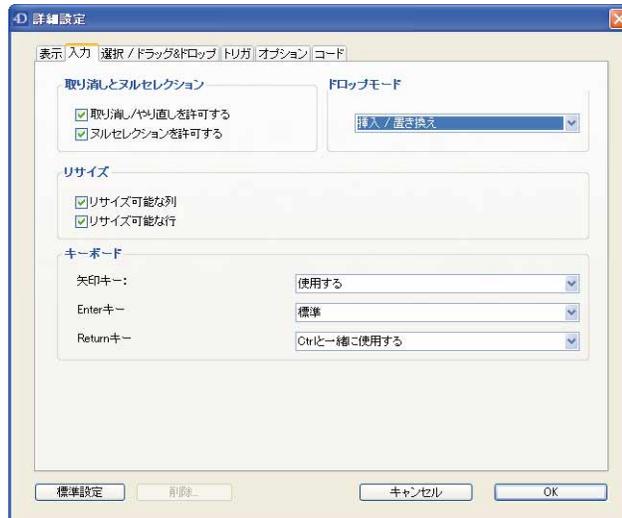
■ ヘッダ：このオプションを使用し、行または列のヘッダエリアを表示または非表示に設定します。

関連する入力エリアを使用して、行ヘッダの幅と列ヘッダの高さをそれぞれピクセル単位で指定します。

- **選択**：このオプションを使用し、選択範囲のハイライトモードを指定します。「ハイライト」オプションを選択すると、エリアの選択範囲が強調表示されて区別されます。この場合、「セルをハイライト」オプションにより、選択範囲内のアクティブセルを視覚的に識別するかどうかを指定します。「Keep in inactive area：非アクティブエリアのハイライトを維持する」オプションにより、エリアがアクティブではない場合もハイライト表示したままにするかどうかを指定します。

「入力」ページ

このページを使用し、エリアのデータ入力オプションを決定します。



- **取り消しとヌルセレクション**：このオプションを使用し、アクションの取り消し（「編集」メニューの「取り消し」コマンドを使用する）、およびエリアでのヌルセレクションを許可（または禁止）します。
- **ドロップモード**：このメニューを使用し、ドラッグ&ドロップ操作時に、エリアのセルで許可されるデータドロップのタイプを設定します。このプロパティは、そのエリア内で実行可能となるアクションを設定するだけである点に留意してください。データのコピーはこれとは別に管理しなければなりません。
- **リサイズ**：このオプションを使用し、ユーザがエリアの行や列のサイズを手動で変更できるかどうかを指定します。
- **キーボード**：このメニューを使用し、矢印キー、データ入力時のEnterキー、Returnキーの使用を定義または禁止します。

- **矢印キー**：入力確定用として矢印キーの使用を設定します（データ入力の確定およびセルの移動）。カーソルがセル内容の先頭または終わりに置かれている場合にのみ、確定処理が行われます。
- **入力Enterキーモード**：データ入力時のEnterキー（数値キーパッド）の効果を設定します。
 - ・ “標準” を指定すると、このキーは入力を確定してから同じセルを入力モードに選択／切り替えます。
 - ・ “Tabとして扱う” を指定すると、このキーは入力を確定してから右隣のセルを入力モードに選択／切り替えます。Shift+Enterを同時に押すと、左側のセルに対して同じ動作を行います。
 - ・ “Returnとして扱う” を指定すると、このキーは入力を確定してから真下にあるセルを入力モードに選択／切り替えます。Shift+Enterを同時に押すと、真上にあるセルに対して同じ動作を行います。
- **Returnキー**：同一セル内に新しい行を作成する時に、Returnキーを使用する条件を設定します。
 - ・ “使用しない”：複数行のセルの作成を禁止します。
 - ・ “使用する”：Returnキーにより新しい行が作成されます。
 - ・ “Ctrlと一緒に使用する”：Ctrl+Returnキー（Windows）またはCommand+Returnキー（MacOS）を同時に使用すると、新しい行が作成されます。
 - ・ “Shiftと一緒に使用する”：Shift+Returnキーを同時に使用すると、新しい行が作成されます。

「選択／ドラッグ&ドロップ」ページ

このページを使用し、使用可能なセル選択のタイプ、およびドラッグ&ドロップが可能であるかどうかを設定します。



- 「選択」欄：エリアに作成可能なセル選択範囲のタイプを設定するために使用します。“複数のセル”タイプが最も自由に操作できます。
- 「ドラッグ」欄：エリア内でドラッグできるセル選択範囲のタイプを設定するために使用します。複数のオプションを選択すると、さまざまなタイプの選択範囲をドラッグすることができます。
「なし」オプションを選択すると、選択範囲はドラッグできません。この他のオプションが選択されていても、すべて自動的に選択解除されます。
- 「ドロップ」欄：エリア内でデータをドロップできるセル選択範囲のタイプを設定するために使用します。複数のオプションを選択すると、さまざまなタイプの選択範囲へデータをドロップすることができます。
「なし」オプションを選択すると、いずれのタイプの選択範囲へもデータをドロップできません。この他のオプションが選択されていても、すべて自動的に選択解除されます。
- 署名：エリアの“ドラッグ”と“ドロップ”署名を設定するために使用します。署名は英数字タイプの文字列（最大32桁）であり、その内容は自由に設定することができます。

ドラッグ&ドロップが許可されると、2つのエリアに共通する署名が少なくとも1つは存在し、その署名がソースエリアに対しては“ドラッグ”タイプであり、ターゲットエリアに対しては“ドロップ”タイプである場合に、これら2つのエリア間（または同一エリア内）でドラッグ&ドロップを行うことができます。署名の追加や削除を行うには[+]と[-]ボタンを使用しなければなりません。

詳細については、4D Viewの『ランゲージリファレンス』マニュアルの「ドラッグ&ドロップ、はじめに」の節を参照してください。

「トリガ」ページ

このページのオプションを使用し、そのエリアの選択や入力、およびドラッグ操作に関するトリガを設定します。



トリガは“イベントの起動”を行います。このページを使用して、エリア内の選択、入力、ドラッグ操作を起動するイベントを指定することができます。

選択と入力の双方に対して同じトリガを設定した場合は、入力トリガが優先されます。選択とドラッグの双方に同じトリガを設定すると、ドラッグトリガが優先されます。

注：MacOSの場合、Ctrlキーの代わりにCommandキーが使用されます。

■「選択」欄：エリア内での選択を起動するモードを設定するために使用します。

「使用しない」オプションを選択すると、選択範囲を設定することはできません。この他のオプションが選択されていても、すべて自動的に選択解除されます。

■「入力」欄：エリア内での入力を起動するモードを設定するために使用します。入力はアクティブセルでのみ実行することができます。

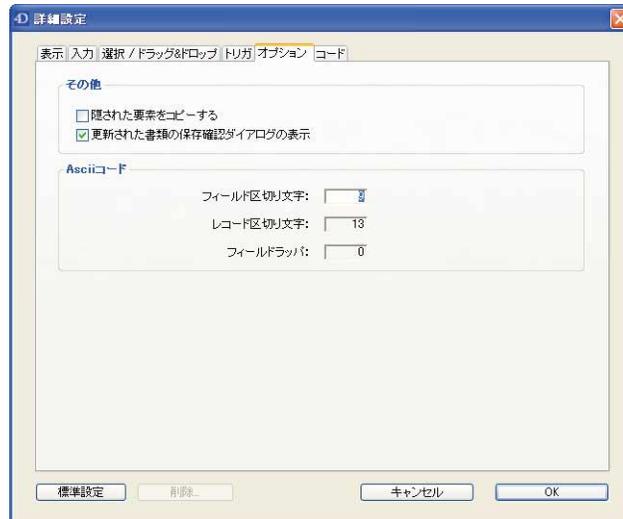
「使用しない」オプションを選択すると、入力は無効になります（入力を起動するイベントはありません）。この他のオプションが選択されていても、すべて自動的に選択解除されます。この場合でも、フォーミュラバーでは入力を行うことができます。

- 「ドラッグ」欄：エリア内でのドラッグを起動するモードを設定するために使用します。ドロップ用のトリガは存在しません。

「使用しない」オプションを選択すると、エリアではドロップ操作を行えません。この他のオプションが選択されていても、すべて自動的に選択解除されます。

「オプション」ページ

このページを使用し、エリア操作に関する各種オプションを設定します。



- 隠された要素をコピーする：コピーする際、隠された項目を考慮するかどうかを指定するために使用します。
- 更新された書類の保存確認ダイアログの表示：4D View 文書が更新された場合に、文書の保存を確認するメッセージを表示するかどうかを指定するために使用します。
- Ascii コード：このエリアのオプションを使用し、フィールドやレコードの区切り文字、およびフィールドドラッパ（囲み）文字の Ascii コードを指定します。

これらの文字は、データの読み込み / 書き出し時にのみ使用されます。

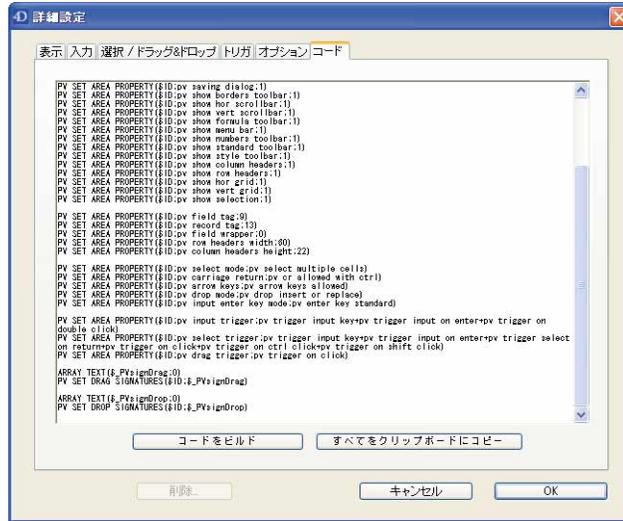
次の例を検証します。

"E1Field1", "E1Field2", "E1Field3"; "E2Field4", "E2Field5";

- コンマはフィールド区切り文字です。
- セミコロンはレコード区切り文字です（E1 と E2 の 2 レコード）。
- 引用符はフィールドドラッパです。

「コード」ページ

このページを使用し、「詳細プロパティ」ダイアログボックスで設定した各オプションに対応する4Dコードを構築します。



コードを作成するには、「コードをビルド」ボタンをクリックします。すると、コードは命令文のリストとしてウインドウ上に表示されます。この命令文のすべて、または一部を選択したりコピーすることができます。

「すべてをクリップボードへコピー」ボタンを使用すると、このコードを直接クリップボードへコピーすることができます。その後、例えば、エリアの初期化用メソッドにコードをペーストすることができます。エリアは変数\$!Dにより参照され、この変数は\$!1の値を受け取ります。

別のエリアでこのコードを使用するため、例えばPV_Init_areaという名前のプロジェクトメソッドにコードをペーストし、PV_Init_area(MyArea)という命令を用いてこのメソッドを呼び出すことができます。

「詳細プロパティ」ダイアログボックスで変更を行った場合は、「コードをビルド」ボタンをクリックして、いつでもコードの再構築を行えます。

プロパティの優先順位

4D Viewエリアを設定するため、「詳細プロパティ」ダイアログボックスまたはプログラム (PV SET AREA PROPERTY コマンド) のいずれかを使用することができます。これら2つのソリューションは同時に、しかも相反する方法で使用可能なため、4D Viewは優先順位を次のように割り当てます。

- 1 フォームメソッドで実行される PV SET AREA PROPERTY コマンド
- 2 「詳細プロパティ」ダイアログボックス。特定のエリアへプロパティを適用します。
- 3 PV SET AREA PROPERTY(0;...)。例えば、「On Startup」データベースメソッドにおいて、すべての4D View エリアへプロパティを適用します。

「ユーザ」モードのダイアログボックスを用いて変更を行っても、「詳細プロパティ」の設定は影響を受けません。

注：エリアを複製すると、その詳細プロパティも複製されます。

4D Pack

4D Pack 2004.1 では、4D データベースストラクチャの管理に関するいくつかの新機能が利用されています（“ユーティリティ” テーマ）。

AP Add table and fields

AP Add table and fields(テーブル名; フィールド名配列; フィールドタイプ配列; フィールド長配列{; リストフォームテンプレート{; 詳細フォームテンプレート})
→ 倍長整数

引数	タイプ	説明
テーブル名	文字列	→ 作成されるテーブルの名前
フィールド名配列	文字列配列	→ フィールド名の配列
フィールドタイプ配列	倍長整数配列	→ フィールドタイプの配列
フィールド長配列	倍長整数配列	→ 文字フィールドのフィールド長の配列 (文字フィールド以外は0)
リストフォーム テンプレート	文字列	→ デフォルトのリストフォーム作成時に 使用するテンプレート
詳細フォーム テンプレート	文字列	→ デフォルトの詳細フォーム作成時に 使用するテンプレート
戻り値	倍長整数	← 作成されたテーブルの番号

AP Add table and fields 関数には2つの引数（任意）が追加され、<テーブル名>のデフォルトフォーム作成時に使用するテンプレートを定義できるようになりました。

フォームのテンプレートは、「フォームウィザード」ダイアログボックスで使用することができます。新しい4D Pack コマンドである **AP Get templates** 関数（後述）を使用すると、4D アプリケーションで利用できるフォームテンプレートの一覧を取得することができます。

<リストフォームテンプレート>には、デフォルトのリスト（出力）フォーム用に使用したいテンプレートの名前を渡します。また、<詳細フォームテンプレート>には、デフォルトの詳細（入力）フォーム用に使用したいテンプレートの名前を渡します。

▼ このメソッドは、2つのフィールドがあるテーブルを作成します。デフォルトフォームとして、2番目と3番目のテンプレートが使用されます。

```

C_STRING(25;TableName)
C_STRING(255;$FormTempList;$FormTempDetail)
ARRAY STRING(32;FieldsNameArray;2)
ARRAY LONGINT(FieldsTypeArray;2)
ARRAY LONGINT(FieldsLengthArray;2)
ARRAY STRING(255;$ArrTemplates;0)
TableName:="住所録"
FieldsNameArray{1}:="名前"
FieldsTypeArray{1}:="Is Alpha Field"
FieldsLengthArray{1}:="20"

FieldsNameArray{2}:="年齢"
FieldsTypeArray{2}:="Is LongInt"

$error:="AP Get templates($ArrTemplates)
If($error=0)
    $FormTempList:="$ArrTemplates{3}
    $FormTempDetail:="$ArrTemplates{2}
    $NumTable:="AP Add table and fields"(TableName;FieldsNameArray;
        FieldsTypeArray;FieldsLengthArray;$FormTempList;$FormTempDetail)
Else
    $NumTable:="AP Add table and fields"(TableName;FieldsNameArray;
        FieldsTypeArray;FieldsLengthArray)
End if

```

参照：AP Get templates

AP Get templates

AP Get templates(テンプレート名配列)→倍長整数

引数	タイプ	説明
テンプレート名配列	文字列配列	→ テンプレート名の配列
戻り値	倍長整数	← エラーコード (0=エラーなし)

テーマ：ユーティリティ

新しい関数、**AP Get templates** は、現行の4th Dimensionアプリケーションで利用可能なフォームテンプレートの一覧を配列<テンプレート名配列>に代入します。

フォームテンプレートは、「フォームウィザード」ダイアログボックスで使うことができます。利用可能なデフォルトテンプレートとしては、例えば、ノスタルジア、OSX、OSX（ラベルなし）、フィラデルフィア等があります。「フォームウィザード」の「詳細設定」ページで作成した独自のフォームをテンプレートとして追加することができます。

AP Create table and fields関数の使用時に、この関数を使用すると、デフォルトフォームに利用するテンプレートを決定できるので便利です。

コマンドが正常に実行されると、0が返されます。それ以外の場合はエラーコードが返されます。

▼ この例題では、利用可能なすべてのテンプレートを納めたウインドウを作成します。

```
$ref:=Open window(<>WLeft;<>WTop;450;400;8;"テンプレートリスト")
ARRAY STRING(255;ArrTemplates;0)
$error:=AP Get templates(ArrTemplates)
DIALOG([Table 1];"ArrTemplates")
CLOSE WINDOW($ref)
```

AP Create method

AP Create method(メソッド名; プロパティ配列; メソッドコード{; フォルダ名})
→倍長整数

引数	タイプ	説明
メソッド名	文字列 (31)	→ メソッドの名前 (31桁)
プロパティ配列	倍長整数配列	→ プロパティの配列 (4項目)
メソッドコード	BLOB	→ メソッドテキストを含むBLOB
フォルダ名	文字列	→ メソッドが作成される「エクスプローラ」フォルダ
戻り値	倍長整数	← エラーコード (0=エラーなし)

テーマ：ユーティリティ

新しい関数である**AP Create method**を使用すると、4th Dimension データベースストラクチャ（インタプリタ版データベースのみ）にプロジェクトメソッドを追加することができます。

<メソッド名>には、メソッドの名前を渡します。この名前の長さは31桁までです。4D オプションの命名規約に従い、文字、数値、スペース、下線を組み合わせて指定することができます。この名前が31桁を超える場合は、31桁に切り捨てられます。

引数<プロパティ配列>を使用すると、メソッドのプロパティを定義することができます。この引数は、事前に倍長整数タイプの配列として定義し、4つの項目を納めなければなりません。

- メソッドを表示しなければならない場合、プロパティ配列{1}には1を渡し、表示しない場合は0を渡します。
- メソッドを**4DACTION**、**4DMETHOD**、**4DSCRIPT**で利用可能にしなければならない場合、プロパティ配列{2}には1を渡し、それ以外の場合は0を渡します。
- メソッドをWebサービスとして提供しなければならない場合、プロパティ配列{3}には1を渡し、それ以外の場合は0を渡します。
- プロパティ配列{3}の値が1であり、メソッドを**WSDL**に公開しなければならない場合は、プロパティ配列{4}に1を渡し、それ以外の場合は0を渡します。

<メソッドコード>には、メソッドのテキストを納めた**BLOB**を渡します。**TEXT TO BLOB**コマンドを使用して**BLOB**への代入を行う場合は、このコマンドの3番目の引数にC string（または0）を渡してください。

<フォルダ名>には、メソッドが作成される「エクスプローラ」フォルダの名前を渡します。これらのフォルダは「エクスプローラ」ウインドウの「ホーム」ページ上で管理され、このフォルダを用いて各オブジェクトを独自の方法で整理することができます。

これは任意の引数ですが、この引数を省略すると、メソッドは“デフォルトプロジェクトメソッド”フォルダ内に作成されます。

処理が正常に完了すると関数は0を返し、それ以外の場合はエラーコードを返します。

- 1= 無効な引数タイプが渡されました。
- 2= データベースはコンパイルモードで動作しています。
- 3= メソッド名が空の文字列です。

AP Create relation

AP Create relation(リレート元テーブル番号; リレート元フィールド番号; リレート先テーブル番号; リレート先フィールド番号)→ 倍長整数

引数	タイプ	説明
リレート元 テーブル番号	倍長整数	→ リレートが開始するテーブルの番号
リレート元 フィールド番号	倍長整数	→ リレートが開始するフィールドの番号

リレート先 テーブル番号	倍長整数	→	リレート先となるテーブルの番号
リレート先 フィールド番号	倍長整数	→	リレート先となるフィールドの番号
戻り値	倍長整数	←	エラーコード (0= エラーなし)

テーマ：ユーティリティ

新しい関数である **AP Create relation** を使用して、2つのテーブル間にリレートを設定することができます。<リレート元テーブル番号>と<リレート元フィールド番号>には、リレート元となるテーブルやフィールドの番号を渡し、<リレート先テーブル番号>と<リレート先フィールド番号>には、リレート先となるテーブルやフィールドの番号を渡します。

リレートは標準のプロパティを用いて設定されます。4Dのリレート管理コマンドを使用して、これらのプロパティの取得や設定を行うことができます。

処理が正常に完了すると、関数は0を返し、それ以外の場合はエラーコードを返します。

▼ この例題は、テーブル1からテーブル2へのリレートを設定します。

```

C_LONGINT($SourceTable;$SourceField;$DestTable;$DestField)
$SourceTable:=1
$SourceField:=2
$DestTable:=2
$DestField:=2
$error:=AP Create relation($SourceTable;$SourceField; $DestTable;$DestField)

```

