

# 4D 2004 Integrated Backup Module

By Yvan Ayaay, Technical Support Engineer, 4D, Inc.

TN 05-46

## Introduction

---

4D 2004(4th Dimension および 4D Server)には、バックアップと復元の操作が統合されており、手動、自動、そしてプログラム制御でそれぞれ実行することができるようになりました。バックアップモジュールが統合されたことにより、必要なファイルを揃えてデータベースを復元する際の手順が簡略化されました。このテクニカルノートでは、**4D 2004** の統合されたバックアップモジュールの基本操作について論じています。扱う内容には、基本設定方法、バックアップおよび復元の実行方法、そしてログファイルの使用が含まれます。

## Overview

---

以前のバージョンの 4D では、**4D Backup** プラグインによってバックアップと復元の操作が制御されていました。**4D 2004** では、同様の仕組みがアプリケーション本体に統合されています。バックアップは手動またはプログラム制御でいつでも開始することができ、スケジューラを使用して自動的に開始させることもできます。バックアップおよび復元のプリファレンスは環境設定で変更することができ、バックアップの対象には、ストラクチャファイル、データファイル、さらに他のファイルやフォルダを含めることができます。ログファイルが使用されていれば、ログファイルのバックアップも作成されます。不測の自体が生じた際には、ログファイルが使用され、バックアップによってデータベースが元の状態に復元されます。

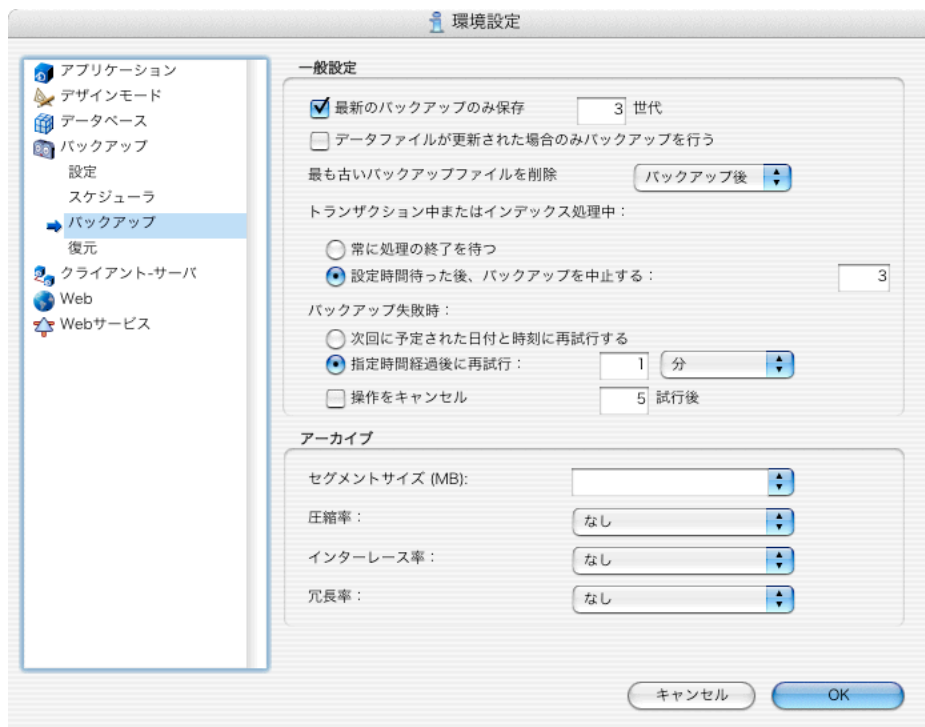
## Backup Settings and Configuration

---

バックアップが実行される際には、バックアップと設定のセッティングが使用されます。ここでは、それぞれの設定項目について概観しています。

### Backup Settings

バックアップのセッティングは、環境設定のバックアップテーマにある「バックアップ」の項目で設定します。次の図に示されているように、デフォルトの値は、バックアップ機能の標準的な用法に合わせてあり、値は自由に変更することができます。バックアップの設定項目は、一般設定とアーカイブに内容が分類されています。



## General Settings

一般設定の項目は、ディスクスペースが限られている場合に有効です。「最新のバックアップのみ保存-x 世代」のオプションを使用すれば、**x+1** 個目のバックアップが作成された場合、もっとも古いバックアップが削除されます。古いバックアップが削除されるタイミングについては、「最も古いバックアップファイルを削除」のオプションで切り替えることができます。スケジューラと併用して「データファイルが更新された場合のみバックアップを行う」を有効にしておけば、データファイルが更新されていないときにはバックアップを次回の予定に持ち越すように設定することができます。

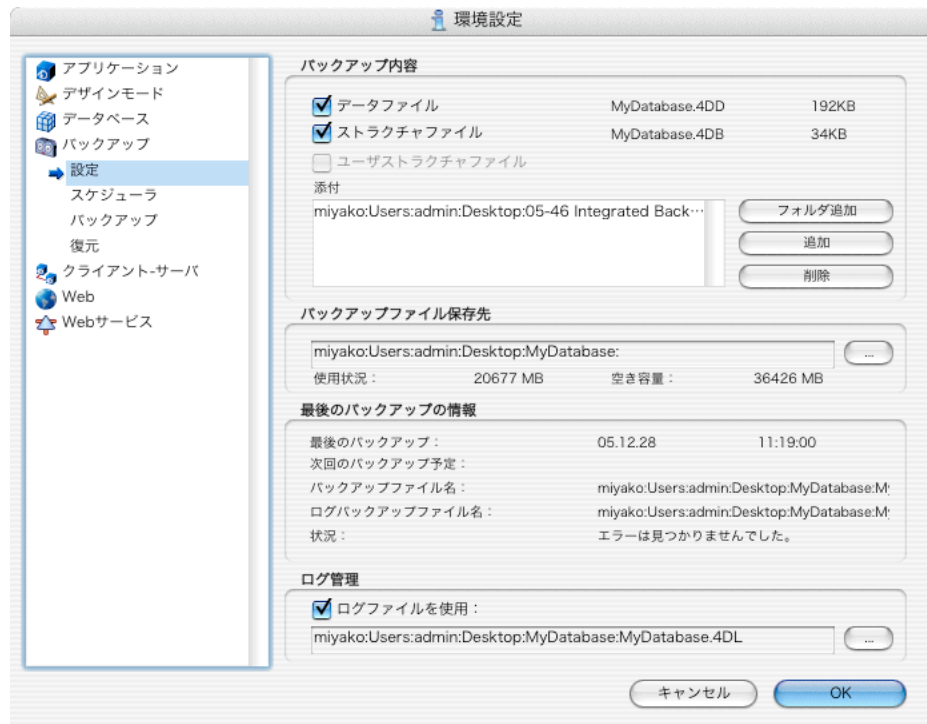
加えて、トランザクション中の場合、バックアップに失敗した場合の動作についても、トランザクションが終了するまで待つ、または一定時間待った後にバックアップを断念するように設定することができます。キャンセルされたバックアップは失敗とみなされ、失敗時のオプションにしたがい、次の予定されたバックアップまで持ち越す、一定時間経過後にリトライ、一定回数のリトライ後に中止のいずれかの仕組みが動作することになります。

## Archive

アーカイブの項目は、バックアップファイルの保存方法に関する設定をするために使用します。セグメント化すれば、アーカイブを別々の場所に保管することができ、圧縮すれば、バックアップの時間は長くなりますが、ディスクスペースを節約できます。

## Configuration Settings

バックアップを実行する際には、環境設定のバックアップテーマにある「設定」の項目も使用されます。。設定項目は、バックアップ内容、バックアップファイル保存先、最後のバックアップの情報、ログ管理に分類されています。



## Backup Contents

バックアップ内容には、バックアップしたいファイルを指定します。デフォルトでストラクチャファイルとデータファイルが選択されており、他のフォルダやファイルを追加することができます。

## Backup File Destination Folder

バックアップファイル保存先は、ブラウザボタン(...)で指定します。

## Last Backup Information

最後に実行されたバックアップに関する情報、つまり、実行された日時、次回のスケジュール、バックアップファイルとログバックアップファイルのファイル名、バックアップの結果について知ることができます。

## Log Management

ログファイルには、フルバックアップ以後、データファイルに加えられた更新の内容が記録されます。これには、レコードの追加、更新、削除、および単一の操作としてのトランザクションが含まれます。

## Ways to perform backup

4D 2004 では、バックアップを手動、プログラム制御、あるいはスケジュールで開始します。

### Manually

バックアップはいつでも **4D Server** のファイルメニューまたは **4th Dimension** のユーザモードのファイルメニューからバックアップの項目を選択することによって開始できます。



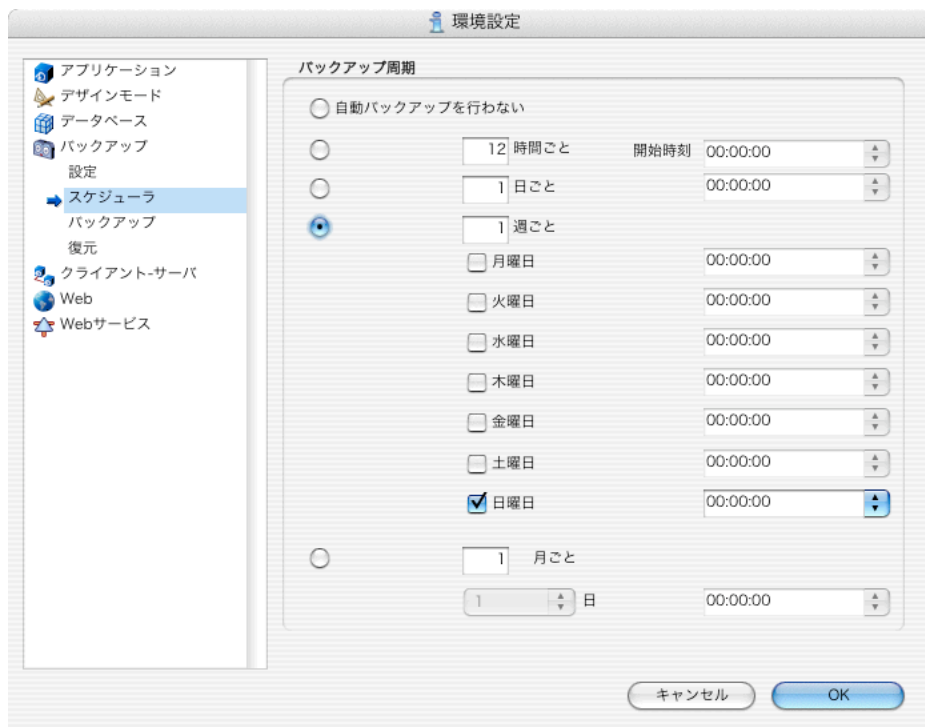
項目を選択するとバックアップウィンドウが表示されます。



バックアップボタンをクリックすると、現在の設定を使用してバックアップが実行されます。環境設定ボタンをクリックすれば、バックアップを実行する前にバックアップの環境設定を変更することができます。

### Automatically

次の図に示されているように、バックアップの環境設定でスケジュールを作成することにより、ユーザによる操作を必要としない自動バックアップを実行することができます。



バックアップの周期は、時間毎、日毎、週毎の特定曜日あるいは月毎に設定することができます。バックアップは、スケジュールされた時刻に実行されます。スケジュールされたバックアップを無効にするには、「自動バックアップを行わない」オプションを選択します。バックアップの一般設定で、データファイルが更新されていないときにはバックアップを次のスケジュールまで繰り返すように設定することもできます。

はじめてデータベースを作成するとき、次の図に示されているように自動バックアップの設定がデフォルトで選択されています。この場合、デフォルトのバックアップ設定が使用されます。自動バックアップを使用しないのであれば、このオプションを外してから新しいデータベースを作成してください。



データベースオプション

☒ 自動バックアップ

このオプションが選択された場合、定期的にデータベースのバックアップが作成されます。デフォルトで、このファイルはストラクチャと同じ階層に置かれます。バックアップの設定は環境設定で変更することができます。

## Programmatically

**BACKUP** コマンドを使用することにより、プログラム制御でバックアップを実行することができます。バックアッププロセスは、新しいデータベースメソッドである **On Backup Startup** と **On Backup Shutdown** でコントロールすることができます。**BACKUP** コマンドが実行されると、そのときのバックアップおよび設定の内容が使用されます。**On Backup Startup** はバックアッププロセスが開始するとき、**On Backup Shutdown** はバックアッププロセスが終了するときに実行されるデータベースメソッドです。

新しいデータベースメソッドは、バックアップに特定の条件を設けたい場合に使用します。たとえば、特定のユーザだけが実行できるようにしたり、同じ日にバックアップの実績があればキャンセルしたりすることができます。あるいはバックアップの保存先を検証して、同じボリュームであれば、警告を表示するようにプログラムすることもできるでしょう。これらは **On Backup Startup** で処理され、**\$0** に **0** を返すことによってバックアップが許可されます。その他の値はすべてバックアップをキャンセルします。**\$0** に渡された値は **On Backup Shutdown** で読み取ることができるので、その値に基づいてエラーハンドリングをすることができます。

バックアップが終了すると **On Backup Shutdown** が実行されます。バックアップのステータスは **Get backup information** 関数で取得することができます。前述のように **On Backup Startup** で **\$0** に渡された値に基づいて特定の処理をすることができます。値は整数タイプの **\$1** に代入されます。このパラメータはメソッドの中で宣言する必要があります。

以下は新しいバックアップコマンドを使用してプログラム制御でバックアップを実行する場合のサンプルのコードです。**BACKUP** コマンドを実行すると **On Backup Startup** でバックアップを開始するための条件が調べられます。満たされない条件があれば、**0** 以外の値が **\$0** に渡され、**On Backup Shutdown** で処理されるようになっています。

```

` Database Method: On Backup Startup Database Method.
` Conditions be met to execute the backup:
` -Only the Administrator can perform the backup.
` -Backup will not be performed when a backup has already been made the same day.
` -Destination of the backup is checked. If the destination is on the same volume as
the current
` location of the database, the user is notified. The user can then choose to proceed or
to cancel
` the backup.

```

```

C_INTEGER(selector;$0)
C_TEXT(Vol_Structure;Vol_BackupDestination;dest_path)
selector:=0
$0:=0

```

```

If (Current user="Administrator") ` allow backup only if the user is Administrator.
GET BACKUP INFORMATION(0;last_backupDate;last_backupTime)
If (last_BackupDate=Current date) ` only perform backup on a different day
    $0:=3 ` backup has already been done on the same day.
Else
    Vol_Structure:=ExtractVolume(Structure f i l e)
    ` Get the current volume of the structure file.
    Vol_BackupDestination:=ExtractVolume(CheckDestination)
    ` Get volume where backup will be saved.
    If (Vol_Structure=Vol_BackupDestination)
        ` if backup destination is on the same volume as the current database
        CONFIRM("Are you sure you want to perform backup on the same
volume?";"Yes";"No")
        ` confirm from user to perform backup
        If (OK#1)
            ` If user does not want to save backup on the same volume as the database.
            $0:=2
            ` client does not want to perform backup on the same volume
        End if
    End if
End if
Else
    $0:=1 ` user is not administrator
End if

```

On Backup Startup の整数タイプパラメータである\$0 に渡される値によってバックアップを実行するかどうかが決まります。\$0 に 0 が代入されればバックアップが実行され、そうでなければ実行されません。上記のメソッドでは、はじめにカレントユーザが調べられます。カレントユーザが Administrator であれば、現在の日付と最後のバックアップ日付が比較されます。両者が一致しなければ、バックアップの保存先とストラクチャファイルのボリュームが比較され、ボリュームが一致しなければバックアップが開始されます。同じボリュームが選択されている場合は、そのことを知らせる警告が表示されます。\$0 には満たされない条件に応じた値が代入されます。

On Backup Startup で\$0 に渡された値は、On Backup Shutdown の整数タイプパラメータである\$1 に代入されるので、この値や GET BACKUP INFORMATION コマンドを使用して、カスタマイズされたエラーハンドリング処理をすることができます。サンプルコードでは、On Backup Startup で満たされなかった条件については固有のエラーメッセージを表示し、その他のエラーについては該当するエラー番号とエラーメッセージを表示しています。

```
`Method: On Backup Shutdown Database Method
C_INTEGER(error_num;$1)
C_TEXT(error_desc)
Case of
  ¥($1=1)
    ALERT("Backup was not performed because the current user is not the administrator. ")
  ¥($1=2)
    ALERT("Backup was not performed because you prefer not to save the backup on the same volume as your database.")
  ¥($1=3)
    ALERT("Backup was not performed because a backup was just performed today. ")
  ¥ ($1=0)
    ALERT("Backup Successful.")
Else ` An error occurred during backup.
  GET BACKUP INFORMATION(2;error_num;error_desc)
  ALERT("Error: "+String(error_num)+" - "+error_desc)
End case
```

バックアップの設定は **backup.xml** に保存されています。環境設定に表示される項目のほとんどはこのファイルから取られており、4D の XML コマンドを使用すれば、これらの値をプログラム制御で変更することができます。紹介した On Backup Startup では、*CheckDestination* メソッドでバックアップの保存先を取得しています。メソッドでは、はじめに **backup.xml** ファイルを解析し、保存先フォルダを示している要素を特定することによって値を返しています。

```
`Method: CheckDestination
` Description: It will check and return the backup destination folder.
C_TEXT(Path_Structure;$0)
C_TEXT(Path_Prefs;Path_Structure;Path_BackupXML)
C_TEXT($elementRef1;$elementRef2)
C_INTEGER($separator;$StrPtr)
C_INTEGER($myPlatform;$mySystem;$myMachine)
PLATFORM PROPERTIES($myPlatform;$mySystem;$myMachine)
If ($myPlatform=Windows ) ` check if platform is Windows or Mac
  $separator:=Ascii("¥¥")
Else
  $separator:=Ascii(":")
End if
Path_Structure:=Structure file ` get the pathname of Structure file.
$StrPtr:=Length(Path_Structure) ` get length of string
```



```

While (Path_Structure[[${StrPtr}]]#Char($separator)) ` starting from the end of
string, find first
  occurrence of the separator
  $StrPtr:=$StrPtr-1
End while
$var1:=Substring(Path_Structure;1;$StrPtr) ` extract path to preferences
Path_Prefs:=$var1+"Preferences"
Path_BackupXML:=Path_Prefs+Char($separator)+"Backup"+Char($separator)+"Ba
ckup.xml" ` path to
backup.xml file
$ref1:=DOM Parse XML source(Path_BackupXML)
$elementRef1:=DOM Find XML
element($ref1;"/Preferences4D/Backup/Settings/General")
$elementRef2:=DOM Find XML
element($elementRef1;"/General/DestinationFolder") ` find Destination
folder element
DOM GET XML ELEMENT VALUE($elementRef2;value) ` get element value for
destination folder
DOM CLOSE XML($ref1)
$0:=value ` return destination folder of backup.

```

注記: 4D Server の場合、クライアントマシンのバックアップコマンドは、ストアドプロシー  
ジャとみなされ、自動的にサーバマシンで実行されます。

## Using a Log File

データベースを最新の状態に戻したい場合、復元だけでは不十分な場合があります。最後のバ  
ックアップを実行した後に加えられた変更は、復元によっては元に戻らないからです。加えて、  
4D はパフォーマンスを向上するためにデータキャッシュを使用しており、変更は即座にディス  
クに書き込まれません。事故が起こると、ディスクに書き込まれていない情報は失われてしま  
います。そのような状況においては、ログファイルが役に立ちます。

ログファイルには、更新、追加、削除やトランザクションなどのデータ操作が記録されていま  
す。レコードの変更は、データファイルやキャッシュに書き込まれると同時にログファイルに  
も時系列にそって書き込まれます。ログファイルの情報は、最後のバックアップが実行された  
時点からはじまっています。バックアップを実行すると、ログファイル自体もバックアップさ  
れ、新しいログファイルが作られます。

ログファイルを使用すれば、最後のバックアップが実行された後に加えられた変更を統合して  
データベースを最新の状態に戻すことができるだけでなく、キャッシュがディスクに書き込ま  
れる前に問題が発生した場合についても、次回、データベースを開く際に足りない操作を自動  
的に統合するようになります。(「データベースが完全でない場合、最新のログを統合する」)

ログファイルを使用するためには、まずバックアップを実行する必要があります。続いて、バックアップの環境設定におけるログ管理のオプションで、ログファイルを使用するように設定することができます。デフォルトでこのオプションは選択されており、いつでも無効にすることができます。

ログファイルの内容は、ファイルメニューで「ログファイルをチェック」を選択することによって確かめることができます。

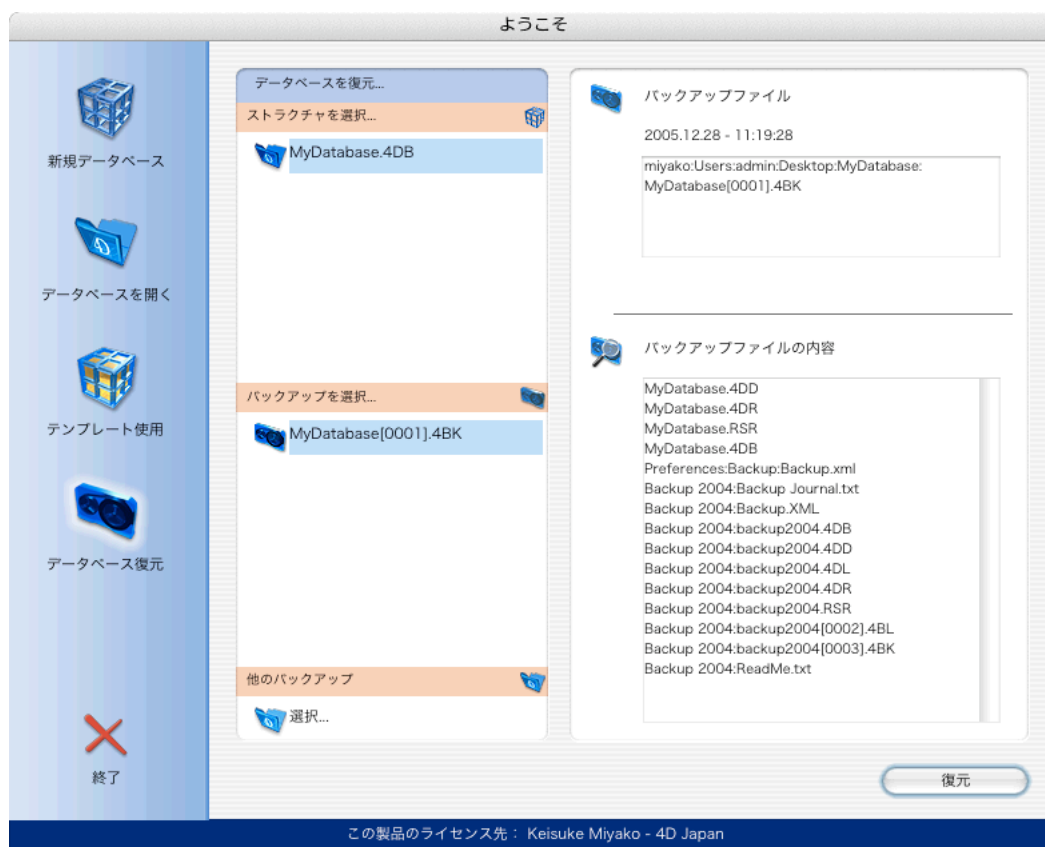


## Restoring a Backup

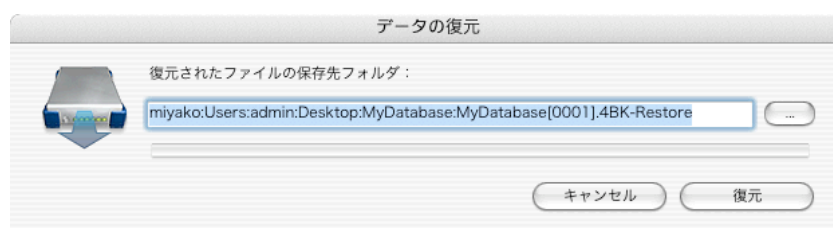
データの消失、クラッシュ、停電などの事故によってデータベースが破損した場合、手動またはプログラム制御によってバックアップを復元することができます。同じようにログファイルも、手動またはプログラム制御によって統合することができます。

## Manual Restore

手動でバックアップを復元するには、ようこそダイアログ画面の復元ページを使用します。**4D Server** または **4th Dimension** を起動し、復元ボタンをクリックすることによって、このページを表示することができます。メソッドの中で **RESTORE** コマンドを実行することによっても表示することができます。



復元するファイルには、バックアップファイルまたはログバックアップファイルを選択します。バックアップファイルのパスと中身は、画面の右側に表示されます。他のファイルを選択するには、選択ボタンをクリックします。復元ボタンをクリックすると、データの復元ダイアログが表示され、復元されたファイルの保存先を変更することができます。



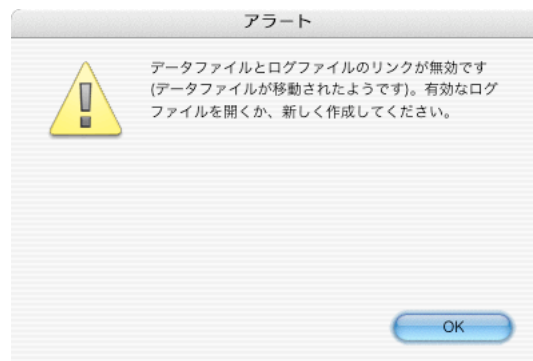
注記: バックアップファイルの名前は **DatabaseName[000x].4BK**、ログバックアップファイルの名前は **DatabaseName[000(x-1)].4BL** です。ログバックアップファイルの番号は、常にバックアップファイルの番号よりもひとつ少ないものになります。

データベースを復元した後、手動でログファイルを統合して、バックアップが実行された後に加えられた変更を統合することができます。ログファイルは復元したデータベースに対応している必要があります。バックアップが実行されると、カレントログファイルが閉じられてバックアップされます。以後の変更はリセットされたカレントログファイルに記録されます。ログバックアップファイルは、最後のバックアップ以前の状態を取り戻すために使用できます。以下は、ログファイルによる統合を実行できるシチュエーションの例です。

#### Scenario 1:

バックアップは毎日実行されていましたが、水曜日にデータベースがクラッシュしました。火曜日のバックアップを使用して、最新の状態にするには次の手順を踏みます。

1. 火曜日に作成されたバックアップを復元します。
2. 復元すると、次のようなダイアログが表示されます。



3. OK ボタンをクリックすると、ログファイルを選択するためのダイアログが表示されるので、カレントログファイルを選択します。これにより、火曜日にバックアップが実行されてから、データベースがクラッシュするまでのログが統合されます。

#### Scenario 2:

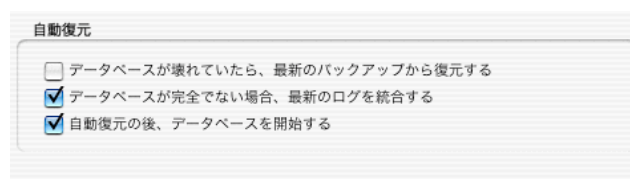
最後のバックアップよりも古いバックアップを使用する場合、復元を連続して行う必要があります。たとえば、**Scenario 1** と同じようにバックアップが毎日実行されているデータベースで、月曜日のバックアップから水曜日のクラッシュ直前の状態に復元するには次の手順を踏みます。

1. 月曜日のバックアップを復元します。
2. 火曜日のログバックアップファイルを復元します。
3. 復元されたデータベースを起動して復元されたログファイルを選択します。これによって火曜日に加えられたデータの変更が統合されます。
4. 環境設定のバックアップ設定で、ログファイルの使用を中止します。
5. 再びログファイルの使用を有効にして水曜日に使用されていたログファイルを選択します。

これによって水曜日にクラッシュするまでに加えられたデータの変更が統合されます。(環境設定にアクセスできない場合は、復元されたデータベースのログファイルを別の場所に移動します。データベースを再起動すると、ログファイルを選択するダイアログが表示されるので、カレントログファイルを選択すれば、最後に加えられた変更を統合することができます。)

## Automatic Restore

復元に関する環境設定の項目を変更することによって、バックアップの自動復元を実施することができます。



「データベースが完全でない場合、最新のログを統合する」と「自動復元の後、データベースを開始する」はデフォルトで選択されています。最新のバックアップから復元するためのオプションは任意で有効にすることができます。

「データベースが壊れていたら、最新のバックアップから復元する」を有効にすると、データベース起動時に問題が検出されると最後のバックアップからの復元が自動的に実行されます。合わせて「自動復元の後、データベースを開始する」を選択することにより、ユーザの介入を必要としないバックアップからの自動復元を有効にすることができます。

加えて「データベースが完全でない場合、最新のログを統合する」も選択されていれば、事故後には自動的に復元が実行され、復元されたデータベースが自動的に再起動し、カレントログファイルが統合されることになります。実行されたバックアップに関する情報は、ストラクチャファイルと同じ階層にある **Backup Journal.xml** ファイルに記録されます。

## Summary

---

4D 2004 には、バックアップモジュールが統合されています。バックアップおよび復元は、手動による操作、プログラムによる制御、または自動化して実行することができます。バックアップが実行される時には、バックアップの環境設定が採用されます。バックアップは、ファイルメニューから選択して実行することができます。バックアッププロセスを開始また制御するには、新しく設けられたバックアップコマンドを使用します。ログファイルを使用することもできます。データファイルに加えられた変更は、トランザクションを含め、ログファイルに記録されます。万一の場合は、ログファイルをデータベースに統合することによって、データベースを最新の状態に戻すことができます。4D 2004 のバックアップモジュールは、極力簡単な操作でデータベースのバックアップおよび復元ができるようになっています。