

# CONVERSION TO 4D 2004

By 4D, Inc./4D S.A.

TN 06-28

<b>Foreword</b> .....	2
<b><i>Help with Migration to 4D 2004</i></b> .....	3
<b>1 - Conversion</b> .....	3
Overview	3
Before you begin	4
Required elements and tools to be used for conversions	4
Handling of Externals and Plug-ins	4
Plug-ins for versions 3.5 to 2003	5
Conversion of very early versions (versions 2 and 3)	6
<b>2 – Steps to follow</b> .....	8
Handling possible data problems revealed by the conversion	8
<b>3 – 4D 2004</b> .....	12
Structure and data files: “resource fork” and “data fork” separation	12
New architecture and operating system	13
Compatibility of plug-in formats	13
New folders	13
New location of Preferences folder	14
New plug-in architecture	14
New locations	14
New folder for plug-ins	15
Downloading plug-ins in client/server mode	15
Loading priority	16
Activation of licenses	16
<b>4 – Converted databases: compatibilities</b> .....	17
Structure compatibilities	17
Web compatibilities	18
Menu bars	18
Platform (Preferences and Form)	19
Size of form window	20
System Highlight Color	21
ASCII filters	21
Conversion of plug-ins	21
<b>5 - Integrations</b> .....	22
Integrated 4D Insider functionalities	22
4D Customizer features integrated into 4D	22
Integrated 4D Customizer features in plug-ins	24
New parameters	25
Integration of 4D Backup and addition of new features	25
Integrated 4D ODBC	26
3 new plug-ins with 4D 2004	27
Users and Groups	27
<b>6 – New behavior</b> .....	27
Formula editor	27

Harmonization of list form operations in User Mode, Custom Menus mode and in subforms: Three selection modes for all lists	28	
New features concerning Output forms	28	
New events concerning lists	28	
New commands concerning lists	28	
Modified commands concerning lists	29	
Modified commands concerning Entry in list	29	
New features concerning subforms	29	
<b>7 – A few essential keyboard shortcuts .....</b>	<b>31</b>	
Modified shortcuts	31	
Added shortcuts	32	
Switching from Custom Menus mode to User mode	32	
<b>8 – New features in 4D 2004.....</b>	<b>32</b>	
New Language features	33	
New commands	33	
Modified commands	34	
Renamed commands	34	
New parameters	34	
New form event	35	
New Backup features	35	
New List Box features	35	
New 4D Tools feature	36	
New 4D Write features	36	
New 4D View features	37	
New 4D Pack features (2004 and 2004.1)	37	
Component	37	

## **Foreword**

This document is a compilation of various documents published by 4D SA: technical notes, tech-tips, product documentation, etc.

The migration to 4D 2004 is, in most cases, very simple: copy, compact, convert.

We will cover conversions of former databases, possible problems that might occur, compatibility issues, behavior modifications and new features.

With 4D 2004, the maximum has been done to preserve compatibility with previous functioning so that you can avoid having to make changes. Your migration should therefore take place smoothly.

# **Help with Migration to 4D 2004**

## **1 - Conversion**

### **Overview**

Upgrading is based on a few principles:

- Upgrading recent versions does not pose any problems:  
For example, if you use a version 6.5.x database and you want to upgrade to version 2004, it is not necessary to pass through versions 6.7, 6.8, 2003 and then to 2004. You can convert your database directly to version 2004 and use the intermediary stages only if you run into problems. Even if certain operations may have changed, there are several compatibility options available so that you can avoid having to modify everything manually.
- If you are using version 3 or earlier, you should use an intermediary version like 6.5 or 6.7. From there you can pass directly to 2003 or 2004.
- If you are both upgrading and changing platforms, you should first upgrade your version using the original platform and then change platforms when the upgrade to 2004 has been completed.  
For example, if you want to upgrade a database from version 3 under Mac to version 2004 under Windows, you should first upgrade the database to version 2004 under Mac and then, once the upgrade is completed, you can switch the database to Windows.
- Do not launch former versions of 4D on machines that have a recent OS.

Former versions like versions 2 and 3 are obviously not meant to operate correctly on a recent OS. For example, launching a version 2 under Mac OS 9 or higher can lead to problems at the data level. You can use the following table as a reference when launching a former version:

4D Version	Windows	Mac OS
2.x (*)	N.A.	8.6
3.0.x (**)	N.A.	8.6
3.5.x (**)	Windows 95/98/Millennium/NT3.5/NT4.0	8.6
6.0.5	Windows 95/98/Millennium/NT4.0	8.6
6.0.6	Windows 95/98/Millennium/NT4.0	9.0
6.5	Windows 98/98SE//Millennium/NT4.0	9.0, 9.1
6.7	98SE/Millennium/NT4/2000/XP	9.0 to 9.2.2
6.8	98SE/Millennium/2000/XP	9.1, 9.2.2 to 10.2.2
2003	98/Millennium/2000/XP/2003	9.2.2, 10.2.8, 10.3.9
2003.7	2000/XPsp2/Serv2003	9.2.2(***)/10.2
2004	2000sp4/XPsp1-sp2/Serv2003	10.3.5
2004.2	2000sp4/XPsp1-sp2/Serv2003sp1	10.3.9/10.4.1

(\*) Versions 2.x are US versions and correspond to European versions 4.x

(\*\*) Versions 3.x are US versions and correspond to European versions 5.x

(\*\*\*) CarbonLib 1.4 minimum

This table does not show all the different compatibilities between versions and OSs (certified OS versions); it merely indicates the principal ones.

- Use demo versions of 4D single-user to upgrade your databases. It is not necessary to have a license for each intermediary version. The demo version is sufficient to pass the database through an intermediary version.
- Different intermediary versions of 4D and 4D Tools are available on the [www.4d.com](http://www.4d.com) site or on various CDs published by 4D like the CDs provided as part of the DevNet program.

## Before you begin

- Always make a conversion using a **copy** of your database since the procedure cannot be undone. Avoid simply making a copy of the folder next to your original database folder. It is even advisable to use a compression utility to backup the database beforehand. Simply copying the database into another folder is not recommended since the structure file may still point to the original data file and you may end up converting it instead of the copy. If the database has been compressed beforehand, a working version of the data file will always be available.
- Before converting it, compact the data of the former version of your database using the corresponding version of 4D Tools.

## Required elements and tools to be used for conversions

If you upgrade from version 2 or 3, under Mac OS, you must have **ResEdit** if your database used external packages or external routines.

- **ResEdit** is a Mac OS 9 application that runs on the Classic environment.

You can download it at:

[ftp://ftp.apple.com/developer/Tool\\_Chest/Developer\\_Uutilities/ResEdit\\_2.1.3/](ftp://ftp.apple.com/developer/Tool_Chest/Developer_Uutilities/ResEdit_2.1.3/)

- **4D Transporter** is required in order to transport a Mac database to Windows for versions prior to 4D 2004.
- In addition, to upgrade a database, you must have the **interpreted structure** and, if a password system has been set up, you must know the **Designer password**.

## Handling of Externals and Plug-ins

### Externals

Before version 3.5 of 4D, plug-ins were "external packages." If a database has never been upgraded from this version, or if it is in a version earlier than version 3.5, there may be externals inserted into the structure that must be deleted before carrying out the conversion.

Before converting a database from this version to the current version or an intermediary one, you must delete the external routines and the 4D modules.

- For the 4D modules, the "Installers" of the various externals can be used to delete their respective resources. If you no longer have the installer for the module, refer to the deletion procedure using ResEdit.
- For external routines, you just need to use "External Mover" or ResEdit.

**Note:** *If your database has been transferred under Windows and still contains externals in its resource file (extension .rsr), you must transfer the database back onto a Mac and delete the resources.*

## Deleting externals

### Using External Mover

If you intend to use **External Mover**, make sure that you are using a system on which it can operate. **External Mover** was never intended to work under Mac OS 9 or higher and must therefore not be launched under this OS.

**External Mover** can be used for databases in version 2, and **External Mover Plus** for databases up to version 3.5. These tools are provided on the 4D disks.

**External Mover Plus** can be launched under Mac OS 9. To open a structure file with **External Mover Plus**, hold down the **OPTION** key while selecting the structure file.

### Using ResEdit

**Important:** Before opening a database with ResEdit, make a copy of your database and work with this copy.

When you open the structure of a database containing externals, you should find different types of resources. The resources installed in the structure might be: "4DND," "4DNX," "4DEX," "4DPX," "4DTE," "4Dte," "4DTF," "FON#," and/or "THM#" used by external routines and packages.

All the resources can be deleted from the structure. You may want to keep the PICT or STR# type resources which can be converted without any problem.

## Plug-ins for versions 3.5 to 2003

In versions 3.5 to 2003, plug-ins are placed in the Win4DX and Mac4DX folders.

When you upgrade a database that uses plug-ins, the upgrade of the plug-ins is carried out separately from the upgrade of the structure and data file. In version 2004, the plug-in architecture has been completely modified. See the "New plug-in architecture" section below, as well as the addendum: [4D\\_2004\\_Upgrade.pdf](#).

Upgrading a plug-in essentially consists in replacing the plug-in with the current version. If a plug-in has been used discontinuously and does not work with the current version, you will lose its functionalities and must either delete any calls to these commands or, if possible, recode the functionalities in 4D or with a different plug-in. Keep in mind that you cannot compile a database that has a missing plug-in.

With plug-ins, you basically have two cases:

- **The plug-in is published by 4D (4D Draw, 4D Write, etc):** In this case, you just need to install its latest version. Certain plug-ins may no longer exist because their functionalities have been added to 4D and you will need to re-write any calls to these routines in native 4D.

- **The plug-in is a third-party plug-in:** In this case, you must contact the plug-in publisher to check its compatibility. In general, plug-ins created for version 6.7 or any earlier versions cannot be loaded by version 6.8 or subsequent versions.

## Conversion of very early versions (versions 2 and 3)

### The "TYPE" and "CREATOR" of various versions and their possible modification

The main problem with versions 2 or 3 (except for external problems) is that the structure will be dimmed in the opening dialog box when you try to open it with version 6.5 or a subsequent version. This is due to the difference of the Type and Creator codes between the various versions of 4D. Below you will find a list of these codes according to the different versions:

4D Version	Structure Type	Structure Creator	Data Type	Data Creator
2.x	BAS3	4D03	data	4D03
3.0	BAS5	4D05	DAT5	4D05
6.5	BAS5	4D06	DAT5	4D06
6.7/6.8	BAS5	4D06	DAT5	4D06
2003	BAS5	4D06	DAT5	4D06
2004	BAS5	4D06	DAT5	4D06

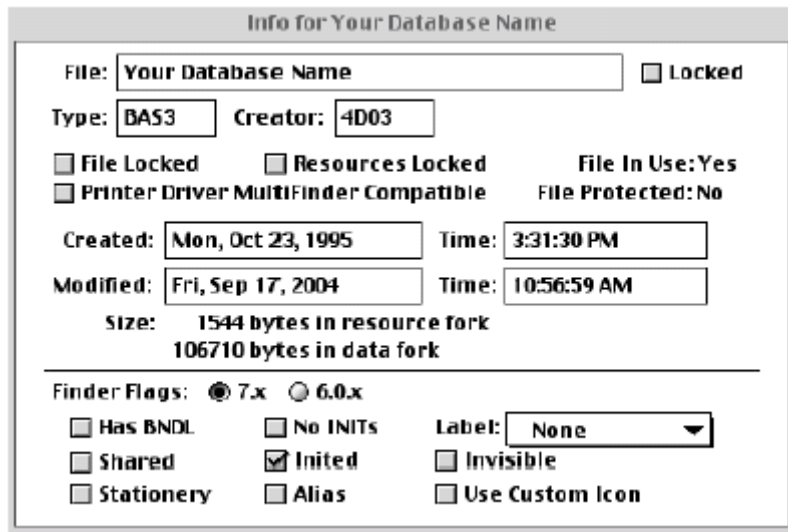
### Changing the "Creator" code

The "Creator" code of earlier databases must match the code of the version with which you want to open it. In order to change it, you need to use ResEdit.

To change the "Creator" code:

- 1** - Launch ResEdit
- 2** - Choose "Get File/Folder info" in the File menu.
- 3** - Select the structure or data file where you want to change the "Creator" code.

The following window appears:

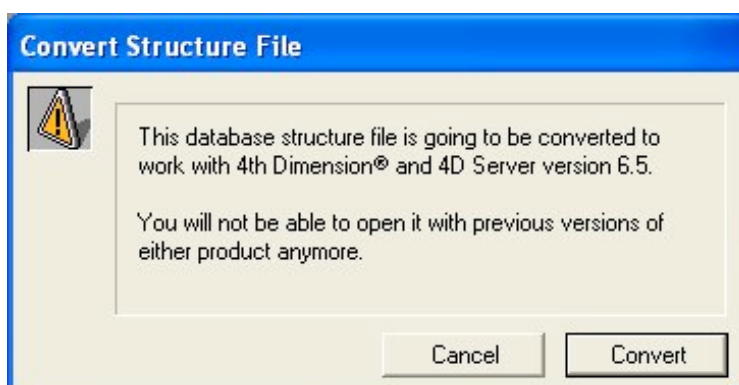


- 4 - To convert to version 6.5 or 6.7 for example, change the Type to "BAS5" and the Creator to "4D06" for the structure file. For the data file, the Type becomes "DAT5" and the Creator becomes "4D06."
- 5 - Save the changes and exit ResEdit.

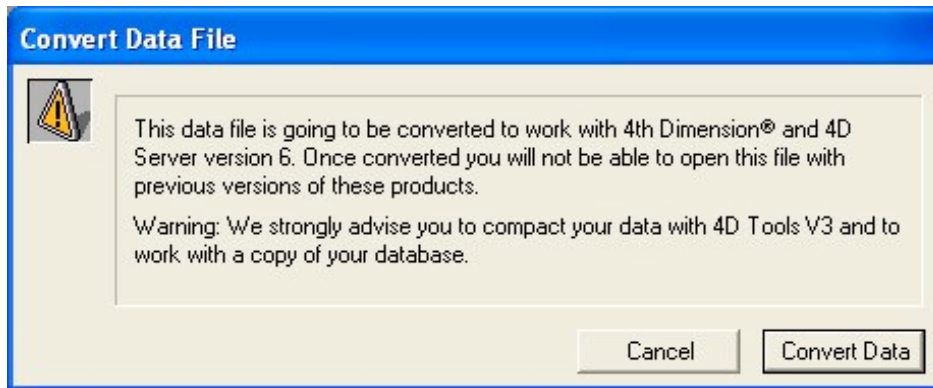
Once the Creator and Type have been changed for the data and structure file, you can open the database using a demo version of 4D. In the present case, we are attempting to open a database with version 6.7, but it "crashes" when opening. We then try to open it with version 6.5. If this doesn't work any better than with version 6.7, we will then try with version 6.0.6r8.

### Opening version 2 with version 6.5

- 1 - Launch version 6.5 of 4D single-user.
- 2 - Choose demo mode.
- 3 - Open the database. The dialog box concerning the upgrade of the structure file is shown.



4 - Click on "Convert." The dialog box concerning data conversion is then shown:



Your database is then converted to 6.5.

Conversion to 2004 should no longer be a problem. If the message:

"This database uses an earlier compatibility mode (BBOX resource) that is no longer used" appears, it is just for information. You can simply validate the dialog box and continue with the conversion.

The BBOX resource was the "compatibility box" that was used to set whether or not the new operation of 4D in version 3 would be used (ON EVENT CALL, ON SERIAL PORT CALL, START TRANSACTION, semaphore locking, flush of transactions, Activation/Deactivation, and so on).

## **2 – Steps to follow**

- Always work with copies;
- Compact your data with the highest version of 4D Tools available (for the starting version);
- Convert.

## **Handling possible data problems revealed by the conversion**

### **Common corruption problems**

Generally speaking, conversion cannot actually corrupt the data or the structure but may reveal a pre-existing problem.

There are two types of corruption problems: data corruption and structure corruption. The causes are usually difficult to pinpoint.

### **Data corruption**

Imagine that we converted an earlier version of a database to 4D 6.5 and got an error message. A dialog box stating that a record cannot be read indicates the most common case of data corruption after conversion.

This warning will be displayed for each corrupted record. Click OK each time. At this point, you must use 4D Tools. Carry out a "Data Check" and "Recover by Tags" using the most recent version of 4D Tools, in our case 6.5, since the most recent versions of 4D Tools are always the most efficient. If this attempt fails, try using the 4D Tools



version corresponding to the starting version of the database and then reattempt the conversion.

When opening the database with 4D Tools, the following type of message may appear: "Your database appears to be severely damaged. You need to carry out a Recover by Tags operation, but you must first select all the data segments if there is more than one."

If you click on OK, you then have the possibility of choosing the data segments.

The different steps to follow are:

- Use 4D Tools;
- Reindex the data file;
- Resave the records;
- Export the data into a new data file.

### Using 4D Tools

4D Tools is a utility program provided with 4<sup>th</sup> Dimension. 4D Tools contains several diagnostic functions that detect and repair damage that may have occurred in a database. Depending on the degree of data corruption, 4D Tools may be able to repair it. Detecting and fixing problems is a simple procedure.

- 1 - Launch 4D Tools with the structure file of your database.
- 2 - Select the associated (damaged) data file.
- 3 - Then select the "Maintenance" tab.

There are three options used to check the data: "Check All," "Check Records..." and "Check Indexes..."

When you use the "Check All" option, 4D Tools immediately begins to check the data file. It checks the integrity of all the records and all the indexes. This is the most thorough option but it is also slower than the other two options.

We recommend choosing this option for regular maintenance of your database, or when 4D has detected a problem in your database that it cannot repair.

If one of the previous options returns an error, the next step is to repair the data file. Select the "Repair" tab. There are three options that can be used to repair data: "Repair All," "Repair Records..." and "Repair Indexes..."

The option you choose will depend on the problem encountered.

#### *Recovery by tags*

Before proceeding with a "Recover by Tags," you must first try to reindex your database and load and resave the records. For further information about reindexing, see the "Re-indexing the data file" section of this document. If reindexing does not solve the problem then proceed with a "Recover by Tags."

When your data file has been severely damaged and cannot be repaired using the Quick Repair options or by reindexing your database, it is then necessary to use the "Recover by Tags" option. When you choose this option, a new data file will be created rather than repairing the former one. Therefore, when using this option, you need to make sure that you have enough disk space. What does recovery by tags consist of? Since records in 4<sup>th</sup> Dimension are of variable lengths, in order to locate them it is necessary to store their "address" in an address table. This way, when the address table is assigned, 4D Tools uses the tag stored in the "header" of each record. This tag is like a summary of the record containing essential information that can be used to locate it in the address table.

After entering a new name for the data file, click on the "Save" button and 4D Tools will begin the recovery by tags. In extreme cases, when using 4D Tools does not solve the problem, a more aggressive approach will be required.

### Re-indexing the data file

The various steps for reindexing the data file are:

- 1 - Make a copy of the structure file.
- 2 - Open the copy of the structure file with a new data file.
- 3 - Delete all the relations then go to each table and uncheck all the indexes,
- 4 - Close the database.
- 5 - Relaunch the duplicated structure with the original data file then close the database again.

This way you will obtain a non-indexed data file.

- 6 - Launch your original structure file with the original data file. By doing this, 4D will reindex your data file.

### Loading and resaving the records

It may happen that you launch your database and that performing queries on your data causes a "crash." If you use 4D Tools, it may tell you that your database is OK. What can you do?

The fastest and most efficient way to know whether your data file is damaged and to find out which records are at the origin of the problem is to load each record and resave it. When a record is damaged, loading it will trigger unexpected behavior. There is a method that loops through all the records of the database and checks whether the record is corrupted. This method will create a log file next to the structure file of the database. The "Journal" file will indicate all the records that have been resaved successfully. This method is available with the French technical note "Récupération d'un fichier de données endommagé" 4D-200506-17-FR (Recovery of a damaged data file) and can be imported into your database for use (Design Mode, New Method, Method Menu, Import method... menu command).

### Exporting data into a new data file

When your data file is severely damaged and the previously described procedures have not been effective, the final solution is to export all your data into a new file. The export may take a lot of time depending on the size of your data. There is a "good" and "bad" way to export data. You may be tempted to use the **SEND RECORD** and **RECEIVE RECORD** commands to export and import all your data; but when you use these commands, since each record is exported as a complete entity, most of the time the problem found in the record will be exported and then re-imported. You can use the standard import/export function in 4D. When you export your data, make sure that you do so with the Tab delimiter and as a Text or XML file. You can export your data as text or XML from the User Mode.

Select Export... in the File menu. When the export dialog box appears, choose the table and fields that you want to export. Select the format in which you want to export them. You can choose to export all or part of the records in the selection. Then click on the Export button.

The advantage of this method is that, for each field, 4D converts the value into text. By doing this, if the value can be converted, the corruption will be eliminated.

### Damaged address table

Depending on the symptom (corrupted index, crashes when trying to access data, etc.), the first thing to do remains the same: try and launch a **Recover by tags** on the converted database. If recovery by tags fails or does not allow you to recover an adequate amount of records, you must return to the previous version of your database and try a recovery by tags with the pre-conversion database. In any case, we recommend this step before conversion when you do not have the correct version of the OS corresponding to the version of 4D Tools. If recovery by tags does not solve the problem, refer to the French technical note "Détecter des erreurs dans les index en cours d'utilisation (4D 2004) " 4D-200410-29-FR (Detecting index errors during use).

**Note:** Keep in mind that while upgrading a file is a relatively simple operation which does not require complete compatibility with the OS you are working on, more complex operations like recovery of records by tags demand total compatibility with the OS.

### Structure corruption

Problems with structure corruption rarely occur during an upgrade. To check whether you have a problem with the structure or the data, simply create a new data file with your upgraded structure. If the problem continues, it's a structure problem.

Version 6.5 and higher of 4D Tools can be used to repair a structure file and this should be your first step in attempting to solve the problem. Keep in mind that the capacities of 4D Tools for solving database problems increase with each new version so make sure that you are using the latest version of 4D Tools (in other words, 2003.7 for 2003 and 2004.3 for 2004).

### Corruption of forms and objects

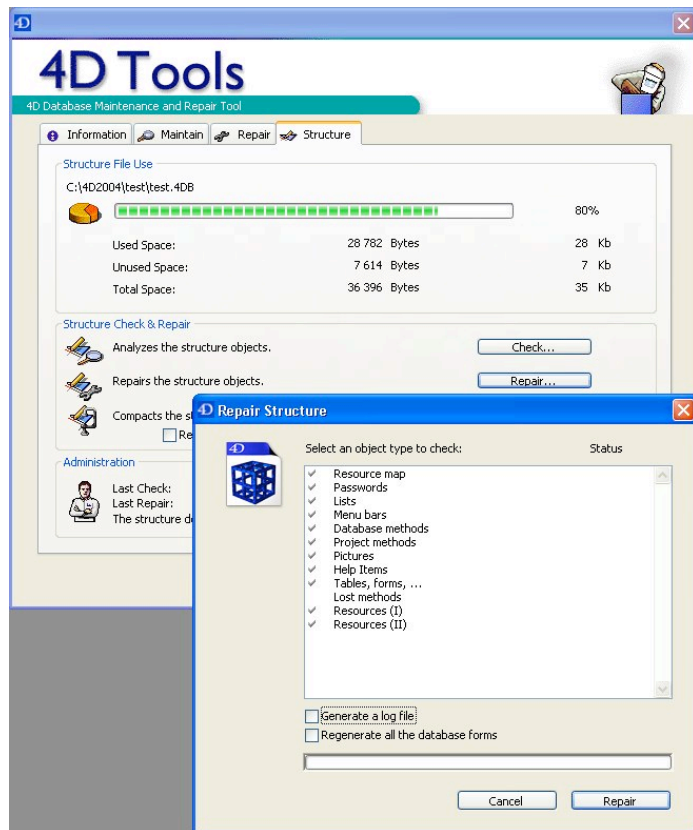
Occasionally, a database can show signs of corruption involving forms or form objects; a form or form object may not be displayed or printed correctly. This may occur in particular for forms built with earlier versions of 4D, which were left untouched subsequently in the structure.

In this case, the solution is to save the form after adding a new object to it; then close and reopen it, and delete the added object. You can also select all the objects of the form and paste them into a new form.

When an object is corrupted, simply copying and pasting an object in the same form resets all the objects.

This should only be done if you do not have the latest version of 4D since version 2004.3 of 4D Tools provides the following option:

- Structure, Structure Check & Repair, Repair – Regenerate all the database forms:



### **3 – 4D 2004**

## **Structure and data files: “resource fork” and “data fork” separation**

From now on, all 4D databases created in 4D 2004, under Mac OS or Windows, include 4 files:

.4DB  
.RSR  
.4DD  
.4DR

Mac databases that are converted to 4D 2004 will still have only two files: the structure file and the data file.

To change them into four separate files, you have several possibilities:

- Compacting the data and structure file using 4D Tools 2004.3.
- Using 4D Transporter (only under Classic environment)
- Using the Mac OS X Terminal

**mkdir** .../Win\_Folder

**cp** MyDatabase Win\_Folder / MyDatabase.4DB

**cp** MyDatabase /..**namedfork/rsrc** Win\_Folder / MyDatabase.RSR  
**cp** MyDatabase.data Win\_Folder / MyDatabase.4DD  
**cp** MyDatabase.data/..**namedfork/rsrc** Win\_Folder / MyDatabase.4DR

## **New architecture and operating system**

- Integration with new generation systems (Windows XP and Mac OS X)
- Support of interface specific characteristics
- Support of Mach-O format under Mac OS X

The 2004 product line does not operate under:

- Mac OS 9
- Mac OS X in Classic mode
- Windows 98 SE
- Windows Millennium

## **Compatibility of plug-in formats**

- Mach-O (Mach Object-file format) was the format of native executables under Mac OS X.
- "PEF" (Preferred Executable Format) was the format of native executables for Mac OS Power PC systems prior to Mac OS X. With a few adaptations, "PEF" executables can run under Mac OS X as well as on earlier systems.
- "PEFs" are sometimes called "CFMs" because the Code Fragment Manager is the mechanism used for the preparation and execution of these types of files.

## **New folders**

"Extras": Contains files to be copied onto 4D Client (resources, text, prefs, object library, and so on). It must be located next to the structure file. It is copied onto the client machine into:

- Mac OS X: Library: ApplicationSupport:4D:DatabaseName\_192.168.xxx.xxx.Extras
- Windows 2000/XP: Documents and Settings\UserName\Application Data\4D\DatabaseName\_address\Extras.

"Licenses": Contains the licenses files that are automatically placed there during activation or that are added manually. Located in the active folder of 4D.

- Mac OS X:Library:Application Support: 4D:Licences
- Windows 2000/XP:Documents and Settings\All Users\Application Data\4D\Licences

"Macros": This new folder contains the various macros that you create.

## New location of Preferences folder

In 4<sup>th</sup> Dimension 2004.3, the locations of these files were standardized. This means that 4D applications can be used on systems where the "All Users" folders have been locked.

By default, these locations are now the following:

	Windows 2000/XP	Mac OS X
All applications except 4D Client	Documents and Settings\All Users\Application Data\4D - > If the All Users folder is locked: Documents and Settings\ User name \Application Data\4D	(All users) Library:Application Support:4D -> If the All Users folder is locked: (Current user folder):Library: Application Support:4D
4D Client	Documents and Settings\ User name \Application Data\4D	(Current user folder):Library:Application Support:4D

Before 4D 2004

- All applications except 4D Client:

Windows 2000/XP: Documents and Settings\All Users\Applications Data\4D

Mac OS X: (All users) Library:Application Support:4D

- 4D Client:

Windows 2000/XP: Documents and Settings\User name\Applications Data\4D

Mac OS X: (Current user folder): Library:Application Support:4D

## New plug-in architecture

In order to operate, plug-ins must be in Carbon Mach-O format or in Carbon CFM (Code Fragment Manager) format (the former plug-in library).

The new plug-ins in Mach-O format are: 4D View, 4D Write, 4D Draw, 4D for OCI, 4D ODBC Pro, 4D Open for 4D, 4D Pack, 4D Internet Commands, 4D for MySQL, 4D for PostgreSQL, and 4D for ADO.

The plug-ins that no longer work with the 4D 2004 product line are: 4D Calc, 4D List, 4D ODBC (earlier version).

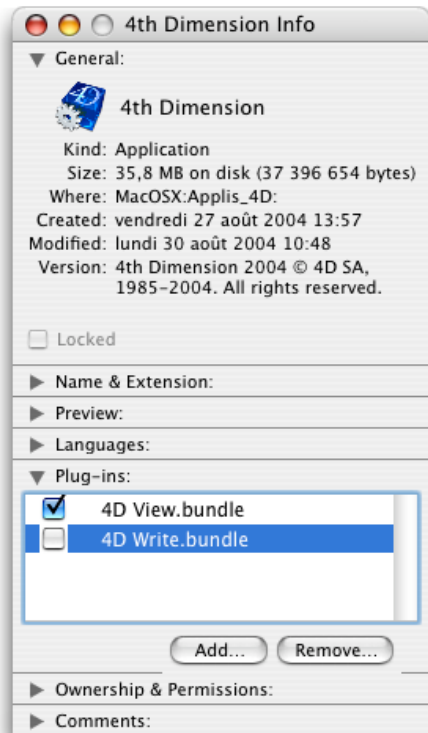
## New locations

Plug-ins can no longer be placed in the active 4D folder. They must be placed either at the database structure level or at the application level.

## New folder for plug-ins

This new folder is named "PlugIns." Both Mach-O or CFM packages can be placed in it since the ".bundle" files can contain both Mac OS and Windows versions. This folder can co-exist with the "Mac4DX" and "Win4DX" folders and can still be placed at the structure level.

Under Mac OS, the installation of plug-ins can be carried out at the application level either by copying files or, more simply, via the "Plug-ins" sections of the Get info dialog box.



In this context, a plug-in may be installed but inactive.

## Downloading plug-ins in client/server mode

When a 4D Client connects to a 4D Server, the plug-ins installed on the server are sent locally to the client machine. Depending on the server configuration, the plug-ins that are downloaded are either placed in a PlugIns folder and/or the Mac4DX/Win4DX folder on the client machine, as described below.

**Note:** With 4D Server 2004, the element hierarchy (folders and files) placed in the downloaded folder is preserved and reconstructed on the client machines. Only the elements corresponding to the client machine platform are downloaded.

**Note:** Additional folders entitled Cache, Server and Database are also created on the client machines for 4D Server internal operation requirements.

## Location of folders downloaded onto the client machine

- If the PlugIns and/or Mac4DX/Win4DX folders are installed on the server machine **at the same level as the database structure:**

Mac OS X

{Disk}:Library:Application Support:4D:DatabaseName:PlugIns (and/or Mac4DX or Win4DX)

Windows 2000 Windows XP

{Disk }:\Documents and Settings\User name\Application Data\4D\DatabaseName\PlugIns (and/or Mac4DX or Win4DX)

- If the PlugIns folder is installed on the server machine **at the same level as the 4D Server application:**

Mac OS X

{Disk}:Library:Application Support:4D:DatabaseName:4D:PlugIns

Windows 2000 Windows XP

{ Disk }:\Documents and Settings\User name\Application Data\4D\DatabaseName\4D\PlugIns

## Loading priority

- Same plug-in at the structure and application level  
-> Only the one next to the structure is loaded
- Same plug-in in the Mac/Win4DX and PlugIns folder  
-> Error message (the database cannot be started)

## Activation of licenses

### On-line activation

In 4<sup>th</sup> Dimension 2004, product registration is essentially carried out via Internet.

**Step 1:** You need to have an Internet connection and an e-mail account in order to register.

You create an account on line and then receive an e-mail reply that needs to be kept carefully.

**Step 2:** Enter your product number.

The server will automatically send you a license file that will be installed in the "Licenses" folder.

An e-mail message will be sent to you with a copy of the license file to be used in case the automatically installed file is lost.



## Off-line activation

**Step 1:** After entering your product number and e-mail address in the off-line activation dialog box, you will generate a seed ID file (Reg.txt). This file can then be placed on another machine that has Internet access.

**Step 2:** Connect to the 4D web site. If you do not already have an account, you will need to create one. Register your product: the "Reg.txt" file will be requested. In return, you will receive a license file by e-mail.

**Step 3:** Copy this file onto your disk and, in the off-line activation dialog box, choose step 3: "Choose file" then proceed with file integration.

## Emergency activation

- This can only be used once for a given product number,
- Only certain product numbers can be used,
- The application will be activated for a period of 5 days maximum.

## Adding extensions

- Help Menu: Updating licenses.
- With a compiled database, you can also use the **CHANGE LICENSES** command.

If you purchased an application directly with extensions (for example, 4D Server with Web server), it is not necessary to add an expansion number: in this case, the extensions are included in the main product number.

You can view the contents of the license file using a browser.

## 4 – Converted databases: compatibilities

Compatibility functionalities are only shown in order to enable compatibility with the operation used in former versions. Former operation is obviously no longer recommended. Certain compatibility functionalities are only present in the Preferences dialog box for converted databases.

### Structure compatibilities

- "Use V3.x.x Startup Method Scheme" option: Activation of the call to the "Startup" procedure for versions prior to 6.0.  
*Tip:* Do not check the option and copy the contents into the "On Startup" database method.
- "Use V3.x.x File Procedure Scheme" option: Triggers are only executed for "Input" forms when entering data in a field or clicking on a button.  
*Tip:* Do not check the option and then use Triggers.
- "Fields are not enterable in dialog boxes" option (checked in **converted databases** and unchecked for newly created databases).
- "6.8 compatibility for text rendering" option: Used to disable the 100% Quartz display under Mac OS X. In version 6.8, the fields, text variables and plug-ins of 4D were displayed in QuickDraw mode (antialiasing, character spacing). Quartz is only used for display (not printing) under Mac OS 10.2 minimum and only on the "Mac Theme" platform.

- “Radio buttons grouped by name” option: You are no longer required to begin radio-button variables with the same string in order to associate them. You can now simply group them. This option only exists in **converted databases**. It is checked by default. Databases created in 2004 use the new operation.
- “Reload form for each record during **PRINT SELECTION**” option: This option is only shown for former **databases converted** to version 2004. It is checked by default. This means that you can automatically reset all the object parameters that the developer might have modified by language in the **On Printing Detail** form event. In order to optimize performance, this mechanism has been removed from 4<sup>th</sup> Dimension 2004. The 4D developer must now reset the desired parameters in the form method himself — this operation is identical to that of list forms using the **On Display Detail** form event. However, you can keep the former mechanism using the “Reload form for each record during **PRINT SELECTION**” option.  
Databases created in version 2004 use this new operation.
- “Automatic Transactions during Data Entry” option: This option only appears for former **databases converted** to version 2004. It is unchecked by default. It was used to automatically start a transaction when an input form with a subform was opened. This option was designed for users of 4D First that switched to 4D. It was already no longer recommended in version 2003.

## Web compatibilities

- “Use 4DVAR Comments instead of Brackets” option: In previous versions of the Web server, the following syntax was used: [MyVar], a proprietary solution to be replaced by <!--4DVAR MyVar-->
- “Use new context referencing mode” option: Used to avoid sending the context number for each element of the page to the browser by the 4D web server.
- “Remove "/" on unknown URLs” option: For example, <http://127.0.0.1/Contacts/Add> returned in \$1: “Contacts/Add” in previous versions. In 4D 2004, all the URLs begin with "/" (except when this option is checked).
- “Use Keep-Alive Connections” option: If an HTML page includes 10 pictures, this previously led to 10+ 1 connections in http 1.0. This option can be used to maintain a single connection.

## Menu bars

### Old Edit Menu Mechanism (option in 4D 2003)

In databases created with former versions of 4<sup>th</sup> Dimension and opened in 4D 2003, an additional option, “Old Edit Menu Mechanism,” was available. In 4D 2004, this same option is entitled “v 6.8” since it applies to databases created with versions prior to 2003 (in other words, versions up to and including version 6.8). This option is checked by default for existing menu bars. It can be used to keep the previous operation of the Edit menu. When this option is checked, 4<sup>th</sup> Dimension automatically adds an Edit menu to the menu bar. In this case, this menu is managed by the system and cannot be modified, unlike the operation of versions starting with 4D 2003.

## Execute without validating

Select the "Execute without validating" option. When this option is checked, 4<sup>th</sup> Dimension will no longer "validate" the field where the cursor is located before executing the associated standard action. This option is mainly intended for Edit menu commands. By default, 4<sup>th</sup> Dimension processes and "validates" the contents of a field before executing a standard action (via a menu command or a shortcut), which then generates an **On Data Change** form event. This may hinder the operation of copy or paste type commands since the **On Data Change** form event is generated unexpectedly when they are called. In this case, it is useful to check the "Execute without validating" option.

## Platform (Preferences and Form)

In the Preferences dialog box, this area only appears in **converted** databases. In previous versions of 4<sup>th</sup> Dimension, it was possible to "force" this appearance at the level of the database, forms and/or objects — since each level was inherited from the previous one by default. This mechanism is still kept for compatibility reasons.

In databases created with 4<sup>th</sup> Dimension version 2004 and higher, the interface is automatically chosen by 4<sup>th</sup> Dimension according to the current platform. The forms and objects used are drawn in a native manner. The interface can be configured only at the level of the forms and objects and only two options are available: **System** and **Printing** for forms, **Inherited from Form** for objects.

- **System**: Regardless of the platform set at the form level, the object is drawn according to the platform on which the application is running.
- **Printing**: Regardless of the platform set at the form level, the object is drawn in a manner that is suitable for printing. This option can be used when certain objects on your form do not have the desired results.
- **Inherited from Form**: The platform interface of the object is the same as the one selected for the form. The platform interface is configured in the form properties.

In databases that have been converted from previous versions of 4<sup>th</sup> Dimension, additional options are available (Automatic, Mac OS 7, Windows 3.11, NT 3.51, Windows 95/98/2000, NT 4, Mac OS 9 and Mac Theme). These options correspond to the former platform interface support mode in 4<sup>th</sup> Dimension and it is no longer recommended to use them. The same goes for the **Get platform interface** and **SET PLATFORM INTERFACE** commands, both of which have been kept for compatibility reasons and can only be used in converted databases. These commands are ignored in databases created with 4D 2004, since the platform interface is handled automatically.

In the **Preferences** of converted databases, it is highly recommended to select the **System** option in order to implement this operation. The two other pop-up menus will then be disabled and the new mechanism will be enabled. Note that it may be necessary to modify the "Platform" property of your forms in order to benefit from it, by choosing for example the **Inherited from Database** option.

- **Inherited from Database:** The form interface is the same as the one selected in the Preferences.

## Size of form window

### Window Size: With Constraints (databases converted to 4D 2004.2 and higher)

Unchecked: form **Resizable** without moving the objects (corresponds to the former operation with "Resizable" unchecked)

Checked: standard mode.

**4D 2004.0: Dynamic adjustment.** The "Dynamic adjustment" property is available in the "Form Size" theme for forms **converted** from a previous version of 4<sup>th</sup> Dimension, when the "**Automatic size**" property is selected.

In previous versions of 4<sup>th</sup> Dimension, when a form had the **Automatic size** property, the size of the form was only calculated when the form was opened. Thus if any modification had been made subsequently using commands such as **MOVE OBJECT**, the form size was not updated. The Dynamic adjustment option ensured this update (unchecked by default in converted databases).

### 4D 2004.2 and higher: With Constraints

For forms in databases of version 2003 that are converted to version 2004.2: This option can be used to reproduce the operation of the former "**Resizable**" form property when it is **not checked** (the user can freely resize the form window but the objects that it contains are neither resized nor moved). The min/max or fixed size properties as well as the object resizing properties are ignored.

This operation corresponds to former versions of 4<sup>th</sup> Dimension and should only be used in specific interfaces for compatibility purposes.

Choosing this option will prevent, in particular, the deformation of buttons in footers when the window is resized.

4D databases created with version 2004 correspond to the **Resizable** option.

## System Highlight Color

A new option appears in the form Property list: System Highlight Color: the row(s) selected will be highlighted according to the color set in the system preferences (the system highlight contrast has no effect on background pictures).

If this option is not checked, the contrast color will be reverse video (black background and white text).

## ASCII filters

The ASCII input and output filters available in User mode in previous versions can now be found in the 4D Preferences, on the Web/Options page.

## Conversion of plug-ins

### 4D Write

After migration from version 6.x.x to version 2003 and 2004, the 4D Write documents saved to disk will have print settings that no longer correspond: US format to 1440% for instance, instead of A4 at 100%. If we decrease the percentage, the resulting format is too small. The fault lies with the "Print record" resource, which differs from Mac OS 9 to Mac OS X.

Here is a method in version 4D 2004 that replaces the faulty print settings with the correct ones in 4D Write documents saved to disk:

```
C_BLOB(MyBlob)
MyArea:=WR New offscreen area
  `stores new default settings in a Blob
MyBlob:=WR Print settings to BLOB (MyArea)
  `open the document in question
WR OPEN DOCUMENT (MyArea)
bOK:=OK
While (bOK=1)
  `replace the settings
  WR BLOB TO PRINT SETTINGS (MyArea; MyBlob;1)
  WR SAVE DOCUMENT (MyArea)
  `and so on
  WR OPEN DOCUMENT (MyArea)
  bOK:=OK
End while
WR DELETE OFFSCREEN AREA (MyArea)
```

**Warning:** The format of 4D Write areas changed in 2004.2. This means that Write documents saved to disk with version 2004.2 and higher cannot be opened with earlier versions.

## **5 - Integrations**

### **Integrated 4D Insider functionalities**

#### **Explorer Folders and 4D Insider Groups**

4D 2004 lets you create folders and subfolders in Design mode (Home page of the Explorer). These folders can be used to gather Tables, Forms, and Project methods that can be organized according to their functions, etc.

The folders are available as a hierarchical list in the Method editor.

They are compatible with 4D Insider Groups and can be moved from one database to another.

4D Insider Group	4D Folder
Tables	Tables
Forms	Forms
Project Methods	Project Methods
Menu bar	Not shown
Lists	Not shown
Style sheets	Not shown
Formats and filters	Not shown
Help tips	Not shown
Database methods	Not shown
STR# Resources	Not shown

#### **4D Object Libraries and 4D Insider Libraries**

4D Insider Libraries	Object libraries
External file with extension.4il	External file with extension.4il
Any object that might appear in a database	Any form object with properties and any dependent objects (lists, style sheets, formats, pictures, related help tips, STR# resources, object methods)

#### **4D Customizer features integrated into 4D**

- Script Manager
- Print progression
- Display of splash screen
- Keyboard shortcuts
- Wedd resources
- Change of 4D Server port number

In the 4D Preferences dialog box

## Script manager

4D Preferences

TRIC resources

- 4<sup>th</sup> Dimension: TRIC resource or, if not available, TRI#
- System: Query and sort carried out by the system (ignores TRIC resource)
- German for v2.2 no longer used for former versions of 4D under German systems
- Turkish (present in order to compensate for a Turkish system bug)
- Date calculation (Farsi)

Concerning languages, there are two systems: Roman systems and Script manager systems.

Version 2004.2 of 4D supports a great number of languages with multi-lingual data entry and display (e.g.: Hungarian, Czech, Arabic, both in the same form or in different fields and/or variables) (Property list: Keyboard Layout).

Mac OS X: 15 languages within the operating system (English, Japanese, French, German, Spanish, Italian, Dutch, Swedish, Danish, Finnish, Traditional Chinese, Simplified Chinese, Korean and Portuguese).

Windows: Multi-lingual pack called MUI (Multilingual User Interface) sold by Microsoft (professional version).

## WEDD Resource

The WEDD resource is used to bind a structure with a particular data file. This means that only one data file, having the same WEDD signature as its corresponding structure, can be opened by this structure.

With 4D 2004, you can specify a WEDD signature for the structure file and data file using the 4D Preferences (Database > Data Management page).

To remove a WEDD signature, you simply need to delete the value entered in the "WEDD Signature" field of the Preferences.

## Keyboard shortcuts

### **User Mode**

2003 (4D Custo)	converted 2004	2004 (4D Prefs)
Accept entry	Alt/Option Enter	ditto Enter
Cancel entry	Alt/Option Esc	ditto Esc
Add subrecord	Enter          ditto	CTRL-CMD SHIFT /

## 4D Server: changing the port number

Changing a port number is now carried out in the Preferences, on the Client-Server/Configuration page.

## Stack Size

4D Customizer Plus also allowed the modification of the stack size for 4D functioning. These settings are no longer necessary and have thus been removed. Commands like New process continue to accept a stack size for compatibility reasons.

## System and command language

The application language can be changed as follows:

Mac OS: in the 4D bundle, Contents:Resources folder, you must have the folder of the language that you want to use. By default, in an English version, you have the « English.lproj » folder. To work with French, you will need to add the « French.lproj » folder in the same location (that you can find in the 4D English bundle). Then, at the application level, choose the « Get info » command and, in the « Language » section, choose the desired language.

Windows: You can simply take the « 4D.rsr » file from a French version of 4D to replace the English « 4D.rsr » and then you will have a French version of 4D.

## Integrated 4D Customizer features in plug-ins

### 4D Draw new features for 2003.3

To integrate the features of 4D Customizer Plus:

Localization of templates (server or client)

#### **DR SET GLOBAL PREFERENCES**

Management of PICTS (pictures or individual objects)

Backup dialog box

#### **DR SET PREFERENCES**

No longer supported: External window suffix



## New parameters

**DR SET ENTERABLE**(area; mode{;buttonMode})buttonMode Integer  
0: Button if area is less than 150 points in height or 300 points in length  
1: Always the area

**DR SET GLOBAL PREFERENCES**(read;write{;selection{;grid{;horLock{;verLock}}})

Selection -> Integer

- 0 No change
- 1 Selection of all objects touched
- 2 Selection of all objects enclosed

grid -> Integer

- 0 Integer
- 1 Grid not activated
- 2 Grid activated

horLock -> String

Letter for horizontal locking (by default: click + H)

verLock -> String

Letter for vertical locking (by default: click + V)

## Integration of 4D Backup and addition of new features

### New features

- Completely new algorithms
- Automatic triggering of backups and restores (structure, data, attached files, etc.)
- Setting of parameters and options with 4D, 4D Server, 4D Runtime Volume License
- Automatic integration of missing operations thanks to the log file
- Possibility of multi-level undo
- Compression and increased security of archives
- Programmed access by XML
- The backup can be carried out in several ways:
  - Design Mode > Preferences > Backup
  - User Mode > File Menu > Backup menu item
  - **BACKUP** command: Calls the **On Backup Startup** database method at the beginning and the **On Backup Shutdown** database method at the end.

### Commands available

On Backup Startup database method

On Backup Shutdown database method

BACKUP

RESTORE

**GET BACKUP INFORMATION**(selector;info1;info2)

GET RESTORE INFORMATION(selector;info1;info2)

## Files generated

DatabaseName[XXX1-0002].4BK: (2<sup>nd</sup> segment of first backup)

LogName[XX12].4BL: 13<sup>th</sup> version of log (starts with 0)

## Miscellaneous

- Compression Rate: If high, slower but takes up less space
- Interlacing Rate: If high, more security but slower (sectors are not adjacent)
- Redundancy Rate: If high, more security but slower (larger file size)
- Automatic backup option when creating a database
- Restore option when opening a database.

## Log file

- Creation of a new file after each backup
- Stores any additions, modifications, deletions or transactions
- Analyzable log: User Mode > File > Check Log File (on 4D Mono or 4D Server) - Action, Table, User, Date and Time, Field values
- Rollback: Undo all operations after the selected line

## Automation of repairs

Self-diagnosis on opening:

- Restore
- Integration of log file
- Startup of database (according to options in the Preferences).

## Automatic restart of 4D Server if it is registered as a service

## Manual restore

- Choose **Restore** tab when opening the database (4D Mono and 4D Server)
- RESTORE command
- Drag-drop the backup file onto the application icon (all 4D applications)

## Log and XML file

Backup log.txt: Pathname, size, archive contents, etc.

"Backup.xml" file in the DatabaseFolder:Preferences:Backup folder

## Integrated 4D ODBC

High-level ODBC commands. There is no compatibility between the 4D ODBC 2003 plug-in and the 4D ODBC Pro 2004 plug-in; however, the ODBC basic functionalities have now been integrated into 4D 2004.

- Integrated into all the 4D 2004 products
- No additional license required
- Import & export to and from an ODBC source
- Access to all ODBC sources via SQL
- ODBC 3.0 API
- Native Mac OS X
- Driver Manager included with MS Windows and Mac OS X

- Windows: Several drivers included
- Mac OS X: No drivers included – “Third party” available:  
<http://www.openlinksw.com/> API Documentation API ODBC
- [http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/htm/odbcodbc\\_api\\_reference.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/odbc/htm/odbcodbc_api_reference.asp)
- “ODBC Framework” component to replace the ODBC 2003 commands by the 2004 commands (can be installed with the ODBC PRO plug-in from the 4D Product Line 2004 CD or downloaded).

### 3 new plug-ins with 4D 2004

- 4D for MySQL
- 4D for PostgreSQL
- 4D for ADO

## Users and Groups

**BLOB TO USERS** (users)

**USERS TO BLOB** (users)

Only the administrator has access to this command.  
Identical to backup from the editor but only to a disk file in this case

**Compatibility Note:** The users and groups files (extension .4UG) created by the Save Groups command in 4<sup>th</sup> Dimension 2003 can be loaded in 4<sup>th</sup> Dimension 2004 using the sequence shown above.

## 6 – New behavior

### Formula editor

Project methods are no longer shown in the formula editor. You must expressly authorize them using the **SET ALLOWED METHODS** command, for the current session:

SET ALLOWED METHODS (methodsArray)

GET ALLOWED METHODS (methodsArray)

**EDIT FORMULA** (table; formula)

For example:

**ARRAY STRING** (15;methodsArr;2)

methodsArr {1}:="Formula@"

methodsArr {2}:="Grand total"

**SET ALLOWED METHODS** (methodsArray)

\$myFormula:=""

**EDIT FORMULA** ([Salaries];\$myFormula)

If (OK=1)

**APPLY TO SELECTION** ([Salaries];EXECUTE(\$myFormula))

End if

## Harmonization of list form operations in User Mode, Custom Menus mode and in subforms: Three selection modes for all lists

- **None:**

No setting of **UserSet** No generation of **On Selection Change** form event.

- **Single**

No setting of **UserSet**

The record becomes the current record.

UP and DOWN arrow keys can be used.

CTRL/CMD+CLICK inverts the selection

**On Selection Change** event generated each time current record is changed.

- **Multiple**

**UserSet** available (Warning: For subforms, the UserSet can be retrieved using the **GET HIGHLIGHTED RECORDS** command – see below).

Clicking does not change the current record.

UP and DOWN arrow keys can be used

CTRL/CMD+click or SHIFT+click for a multiple selection.

**On Selection Change** event generated each time selected record is changed. If the

**enterList** parameter (of the DISPLAY SELECTION command) is set to **True** or

**Enterable in List** is checked, adding and modifying will be in List mode; otherwise, it will be in Detail (Page) mode.

## New features concerning Output forms

- No more Done button
- All types of active objects: in headers, details, breaks and footers
- The "Add subrecord" and "Modify subrecord" standard actions now also operate for output forms.

## New events concerning lists

- **On Load Record**: When changing to Entry in list mode
- **On Selection Change**: Modification of the current record or current selection in a list form, a subform or a list box.

## New commands concerning lists

**GET HIGHLIGHTED RECORDS** ({table;}setName)

setName: Set into which the User selection is copied.

Can be used in several subforms since the name of the table is specified.

The set can be local/client, process or interprocess.

**SCROLL LINES** ({\*;}object|table{;position{\*}})

- \* Subform object (string)
- \* Displays the line in the first position after scrolling.

Can be used to scroll the lines of a subform or an output form using the **MODIFY SELECTION** or **DISPLAY SELECTION** commands.

The *position* parameter is optional; if it is not specified, the command will be positioned on the first record selected.

## Modified commands concerning lists

**GOTO SELECTED RECORD** ({table;}record)

Can now be used to deselect all the records of a list by passing 0 in the *record* parameter.

**HIGHLIGHT RECORDS** ({table}{}; setName{\*})

The *table* parameter specified can be the table of a subform.

Disables the automatic scrolling of the list in order to allow custom management using the **SCROLL LINES** command.

## Modified commands concerning Entry in list

- Entry in list is now available in all types of records lists.

**DISPLAY SELECTION** ({table}{};selectMode{};enterList{\*{}\*}})

**MODIFY SELECTION** ({table}{};selectMode{};enterList{\*{}\*}})

selectMode: 0 No Selection, 1 Single Selection, 2 Multiple Selection; enterList: True or False (False by default)

## New features concerning subforms

- Horizontal scrollbar
- Management of double-click on a line
  - Do nothing
  - Modify record
  - Display record
- Double-click on Empty Line (**converted databases** only)
  - Do nothing
  - Add record (if Enterable in List property checked: in list (output) form; otherwise, in detail (input) form)
- The subform object is focusable
- No more flashing triangle

Comparison of subform properties under 4D 2004 with prior versions:

4D properties of prior versions		4D properties for version 2004	
Enterable	Enterable and double-clickable	Focusable Enterable in List Selection mode Double-click on Line Double-click on Empty Line	yes yes none modify record add record
	Enterable and not double-clickable	Focusable Enterable in List Selection mode Double-click on Line Double-click on Empty Line	yes yes none do nothing do nothing
Selectable	Selectable and double-clickable	Focusable Enterable in List Line selection mode Double-click on Line Double-click on Empty Line	no no single display record do nothing
	Selectable and not double-clickable	Focusable Enterable in List Line selection mode Double-click on Line Double-click on Empty Line	no no single do nothing do nothing
Non-modifiable		Focusable Enterable in List Selection mode Double-click on Line Double-click on Empty Line	no no none do nothing do nothing

The "Double-click on Empty Line" property only appears in **converted databases** in order to ensure compatibility with previous operation. This property is not recommended for use, in conformity with the recommendations concerning the OS ergonomics.

- If the Double-click on Empty Line property no longer allows adding a line, you can still manage this as follows:
  - Either by buttons in the main form
  - Or by buttons in the header and footer of a subform
  - Or by buttons in the footer of the main form
- The current record is no longer loaded:  
Any display or refreshing of the display (or a 4D object getting the focus on the form) caused by the system will unload the current record but keep the pointer to the current record.

Solution: Use the **LOAD RECORD** command.

- The "**On Display Detail**" event is also generated for empty lines. This means that a variable will be displayed until the bottom of the list, which was not the case previously.

Solution using an additional test:

```

-
If (Is record loaded)
...
End if

Or

$n:=Displayed line number
$p:=Records in selection([Clients])
If ($n>$p)
  SET VISIBLE (*,"Popup@";False)
Else
  SET VISIBLE (*,"Popup@";True)
End if
  
```

- Display of objects located in the header of the break area.
- Display of empty lines.
- Visibility of objects depending on their selection.
- All types of active objects in the **header** and in the **break** area.
- Triggering of form events:  
In versions prior to 4D 2004:
  - Trigger
  - On Display Detail form method.

In 4D 2004 (databases created AND converted) – operation identical to a **MODIFY SELECTION** or **DISPLAY SELECTION**

- **On Header** object method
- **On Header** form method
- On Display Detail object method
- On Display Detail form method

## 7 – A few essential keyboard shortcuts

**CTRL/CMD+,**: Displays Explorer  
**CTRL/CMD+;**: Displays toolbox  
**CTRL/CMD+P**: Displays methods  
**CTRL/CMD+L**: Displays forms  
**CTRL/CMD+\***: Displays compiler

### **Modified shortcuts**

**CTRL+CLICK/CMD+CLICK** on a form can now be used to select an object in a group (or on a different level).

The previous operation can be found with: Contextual menu "Select Similar Objects."

**CTRL or COMMAND+SPACEBAR** (Displays the list of tables) has become:

Records->Tables List or

CTRL/COMMAND+0 (zero)

**CTRL+W or COMMAND+W** has become:

File>Flush Data Buffers or

ALT CTRL/OPTION COMMAND +SHIFT+S

**CTRL +CLICK** (cursor in a list) has become:

ALT/OPTION+ CLICK

### Added shortcuts

**ENTER** in order to enter data in a text area of a form and to validate the entry.

**ALT+CLICK** to switch to an object on another page.

### Switching from Custom Menus mode to User mode

Mac OS: no change between 2003.7 and 2004. The shortcut is still Option+F from the splashscreen.

Windows: the former shortcut, Alt+F4, is now used to exit the database (standard operation). To switch to User mode or Design mode, the following shortcuts must now be used:

- - 4D 2003.7 and higher: Alt+Shift+Right click – Switches to Trace mode and adds a line with the Design or User standard action
- - 4D 2004: Alt+Shift+Right click – Switches to User mode

[ftp://ftp2-public.4d.fr/Documents/Products\\_Documentation/LastVersions/Line\\_2004/VIntl/PDF\\_Fomat/4D\\_2004\\_Keyboard\\_Shortcuts.pdf](ftp://ftp2-public.4d.fr/Documents/Products_Documentation/LastVersions/Line_2004/VIntl/PDF_Fomat/4D_2004_Keyboard_Shortcuts.pdf)

## 8 – New features in 4D 2004

- Pointer to local variables;
- New welcome dialog box with management of Favorites;
- Allows opening of structure file in read-only (Design Mode, Access page in the Application theme of the Preferences dialog box: Addition of the "Do not display warning when structure file is in read-only mode" option)
- Sending of the files and folders contained in the Extras folder of 4D Server to the 4D Client machines;
- In subform areas, checkbox in the Property List can be used to allow (or not allow) deletion in the area using the Delete and Backspace keys;
- Type-ahead feature in the Method editor with "(" or ";"
- New licenses dialog box accessible via the Update License menu command or the **CHANGE LICENSES** command;
- Documents opened for SAX reading must be opened imperatively in read-only;
- New dictionaries are available using the **SET DICTIONARY** command;
- Management of data fork resources.
- Version certified Mac OS 10.4 "Tiger"



- Modification of location of Preferences files
- Automatic creation of new data segments
- 4D Server Mac OS and 4D Runtime Mac OS: Keep cache in RAM (to avoid unloading in virtual memory, thus on the hard disk, which can slow down the application).
- Possibility of testing the execution of a form in Design mode
- Property list for a form contains Window Size: With Constraints (**databases converted to 4D 2004**)  
When this option is not checked, the window can be resized without moving the objects (which corresponds to the former "Resizable" operation).  
When this option is checked, the standard mode applies.
- Drag-and-drop of a row from a list box to another 4D object and vice versa.
- Shift key for keyboard shortcuts associated with a menu item.
- Support of non-Roman alphabets: Arabic, Simplified and Traditional Chinese, Croatian, Cyrillic, Greek, Hebrew, Icelandic, Japanese, Korean, Romanian, Thai, Vietnamese. These alphabets can be used for data entry, display and data processing (sorts). Two systems available: Roman systems and Script manager systems.  
4D 2004.2: Supports a large number of languages with multilingual data entry and display (e.g.: Hungarian, Czech and Arabic on the same form, in different fields or variables) (Property list: Keyboard Layout).  
Mac OS X: 15 languages present in the operating system (English, Japanese, French, German, Spanish, Italian, Dutch, Swedish, Danish, Norwegian, Finnish, Traditional Chinese, Simplified Chinese, Korean and Portuguese).  
Windows: Multilingual pack called MUI (Multilingual User Interface) sold by Microsoft (professional version).
- Data entry and display according to the keyboard layout in fields, enterable variables, hierarchical lists, list boxes and combo boxes.

## New Language features

### Method editor:

A new shortcut can be used to open a project method directly in a new window:

Alt+Double-click (Windows) and Option+Double-click (Mac OS)

### Stack size of processes:

Management modified in 2004.3. It is now recommended to allocate 64 KB to the stack of each process (the 32 K was previously often exceeded, automatically, by 4D). From now on, 4D will strictly use the size indicated

## New commands

**SHOW ON DISK** (pathName { ;\*})

Displays the folder specified in the *pathName* parameter. If the item is a folder, it displays its content.

**DOM Get parent XML element**

**DOM Get previous sibling XML element**

## DOM Get last child XML element

### Modified commands

**Drop position:** Now accepts a parameter that retrieves the column number of the list box where the drop occurred.

**PLATFORM PROPERTIES:** New *language* parameter that is used to find out the current system language.

**SET DATABASE PARAMETER, Get database parameter:** New Real Display Precision constant, Selector = 32

**SET FIELD TITLES, SET TABLE TITLES:** New \* parameter.  
When this parameter is passed, the virtual structure is used in the formula editor.  
When it is omitted, the formula editor does not use the virtual structure.

**PRINTERS LIST:** Modification of 2<sup>nd</sup> parameter to allow the use of custom printer names under Mac OS.

**SET DATABASE PARAMETER** ({table;}Debug Log Recording; value)  
Debug Log Recording or 34  
Value: 0: Do not record, 1: Record; 2: Record in detailed mode  
Generates a file called "4DdebugLog.txt" (warning: This file fills up very quickly).

**SET PRINT OPTION** (Hide printing progress option; value)  
**GET PRINT OPTION** (Hide printing progress option; value)  
Value: 0: Do not hide; 1: Hide  
Supplements the "Printing progress" check box of the Preferences

### Renamed commands

DOM Get first XML element becomes **DOM Get first child XML element**  
DOM Get next XML element becomes **DOM Get next sibling XML element**  
SHOW LISTBOX SCROLLBAR becomes SET SCROLLBAR VISIBLE  
Since this command can be used not only with list boxes but also with subforms and scrollable areas.

### New parameters

The **Get 4D folder** command now accepts the following constants in the *folder* parameter: Active 4D Folder, Extras Folder, Licenses Folder and 4D Client Database Folder.

- **SET ENVIRONMENT VARIABLE** accepts two additional constants:
  - \_4D\_OPTION\_CURRENT\_DIRECTORY (Mac: Relative path)
  - \_4D\_OPTION\_HIDE\_CONSOLE (Windows: Hides the console)
- Get database parameter and SET DATABASE PARAMETER commands accept new parameters:
  - Get database parameter** ([table];selector)
  - SET DATABASE PARAMETER** ([table];selector;value)
  - Used to get/set the next number sent by **Sequence number**

**Get database parameter** ([table]; 4D Server Log Recording)

**Get database parameter** ([table]; Web Log Recording)

**Get database parameter** ([table]; Client Web Log Recording)

**SET DATABASE PARAMETER** ([table]; 4D Server Log Recording)

**SET DATABASE PARAMETER** ([table]; Web Log Recording)

**SET DATABASE PARAMETER** ([table]; Client Web Log Recording)

Saves and reads 4D Server, Web or 4D Client Web requests.

- New "Is Text" constant authorized for sending by BLOB:

**SOAP DECLARATION** (blob, Is Text;input\_output)

Can be used to return SOAP arguments that are larger than 32 KB by using BLOBs.

## New form event

**On After Edit:** Triggered in any enterable area by a paste, cut, delete, undo, drop or keyboard entry action or by a command simulating a user action.

(Event to be used instead of **On After Keystroke**, which is more restrictive).

## New Backup features

### Two new commands:

New log file

- > Text: Complete pathname of log file closed

INTEGRATE LOG FILE (pathName)

Used for setting up a logical mirror (see addendum 2004.3)

### New backup preference:

On the Backup/Backup page of the Preference, a new checkbox has been added "Cancel the operation after XX attempts."

Automatic backup when log file reaches critical size of 2 GB

## New List Box features

**SCROLL LINES** (\*;object;position) displays the line of the list box indicated in the *position* parameter at the top of the list box.

Two new constants are available:

**Get listbox information** (\*;object; Position listbox ver scrollbar)

**Get listbox information** (\*;object; Position listbox hor scrollbar)

Returns the position of the vertical and horizontal scrollbars of the list box in pixels.

Blank dates appear empty and no longer as 00/00/00.

Display of three-state checkboxes in list boxes.

Synchronization of arrays and list boxes.

## New 4D Tools feature

Structure tab: Additional checkbox used to delete the compiled code when compacting the structure.

A new option is available: **Regenerate all the database forms**. To be used for converted databases and those with damaged forms (after copying the database and compacting it).

## New 4D Write features

- The WR ON EVENT and WR SET AREA PROPERTY commands have new parameters:  
**WR ON EVENT** (area;wr on timer;method)

Calls the method every 50 seconds by default. Time limit can be changed in: **WR SET AREA PROPERTY** (area;wr timer frequency;value)

- Improved conversion of 4D Write documents to HTML.

- New print commands:

**WR Print settings to BLOB**

**WR BLOB TO PRINT SETTINGS**

Used for storing and retrieving print parameters, layout parameters, or both.

**Warning:** The structure of 4D Write 2004.2 documents has changed. They can no longer be opened with version 2004.1.

### Document compatibility

Since the internal storage format of 4D Write documents has been modified, you can no longer open 4D Write version 2004.3 documents (or 4D Write 2003.7) with earlier versions of 4D Write.

### Border management:

Improved management of paragraph borders.

Change of existing names for constants 43 and 44 of **WR SET TEXT PROPERTY**  
wr top border (43) becomes **wr inside top border** (space added above and below paragraphs)

wr bottom border (44) becomes **wr inside bottom border** (space added above and below paragraphs)

New constants:

**wr top border** (46) (space added above)

**wr bottom border** (47) (space added below)

### Application of zoom value on opening:

New constant:

**WR SET AREA PROPERTY** (area; **wr use saved zoom value**; value)

Value: 1: to use saved zoom value; 0: to return to 100%

### **Deletion of tabs:**

New "Delete All" button can be used to delete all the tabs of the selected text area. This button is found in the tab management dialog box.

### **New 4D View features**

Partial printing of document possible;

New "Advanced Properties" dialog box available in the Property list with display, entry, selection and drag-and-drop, triggers, miscellaneous, ASCII codes and automatic generation of code options.

Two new commands:

PV Print settings to blob (area) -> blob

PV BLOB TO PRINT SETTINGS (area;blob)

### **New 4D Pack features (2004 and 2004.1)**

**AP Add Table and Fields** (tableName; fieldNamesArray; fieldTypesArray; fieldLengthsArray)

**AP Get Templates** (templateNameArray)

**AP Create relation** (sourceTableNum;sourceFieldNum;destTableNum; destFieldNum)

**AP Create method** (methodName; propertiesArray; methodCode {; folderName })

### **Component**

**Current method name** now operates in a private method of the component.