



JFTconv 55

Contents

Contents	2
Introduction.....	3
Copyright	3
Register JFTconv 55	3
About JFile.....	4
Quick Start	5
Access to JFile	5
JFile to Access	5
Main Screen	6
Preferences.....	7
Palm Date / Time	7
Palm User.....	8
Create MDB for	8
JFile Options	8
Security	8
Access Options	9
Misc	9
Default Directories.....	9
Convert Options	10
Select Fields.....	11
List Field Support	12
Access Lookup Support	13
Templates.....	13
Command Line	14
Template Syntax	14
Filename / Option Syntax	14
Convert or Fields	16
Data Translation.....	16
JFile PDB to Access MDB.....	16
Access MDB to JFile PDB.....	17
JFTconv/Access Data Type Ranges.....	17
Access Integration.....	18
What's New	20

Introduction

JFTconv 55 is a simple utility to convert databases between Access and JFile 5.x formats. It supports reading and writing database formats from any version of Access on the PC, and JFile 5.x on the Palm. The older 3.x and 4.x versions of JFile are **not** supported.

JFTconv 55 can be run from a standard Windows interface or from a command line that permits its use in batch processing or other types of automation. It also supports the use of Templates, that save your field and other settings for easy reuse, both in the interface and from the command line.

The demo version of JFTconv 55 can only convert 10 records. Please register JFTconv 55 in order to take full advantage of its capabilities.

Copyright



JFTconv 55

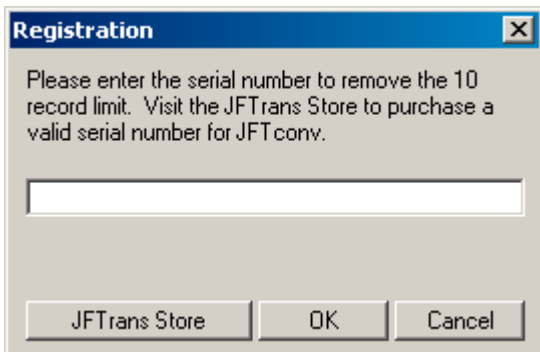
Copyright (c) 1999-2002
by Steven Hawkins

All Rights Reserved

JFile Copyright (c) 2002 by Land-J Technologies

Please write to jftrans@ica.net for tech support or inquiries.

Register JFTconv 55



To register JFTconv 55, click the **About** button and then **Register**. Just enter the license code you received and then click OK. The 10 record demo limit will be removed.

You can purchased a copy of JFTconv 55 by clicking the **JFTrans Store** button, or by going to this URL :

<http://store.esellerate.net/s.asp?s=STR186982738>

About JFile



The screenshot displays two side-by-side windows from the JFile 5.50 application. The left window, titled "JFile 5.50", shows a list of databases with columns for "Database Name" and "#". The right window, titled "Internet Por...-727", shows a list of ports with columns for "Port", "Keyword", and "Description". Both windows have a toolbar at the bottom with icons for "New DB", "Beam", "Modify", "Clone", "Done", "Add", search, list, filter, and trash.

Database Name	#
Apartment Finder	0
ASCII symbols	128
Camping packing	92
DVD Regions	6
Employee Eval. Log	1
Grand Prix 2001	17
Home Finder	0
Internet Ports	727
Major Battles	19
Meeting Manager	0

Port	Keyword	Description
11	systat	Active Users
12	-	Unassigned
13	daytime	Daytime
14	-	Unassigned
15	-	Unassigned [was n
16	-	Unassigned
17	qotd	Quote of the Day
18	msh	Message Send Pro
19	chargen	Character Genera
20	ftp-data	File Transfer [Def

JFile 5.50 is a fast, efficient, and user friendly database program for the PalmOS Platform. Included with JFile 5.50 is a PC based application that can convert Comma Separated Value (.csv) files to and from JFile .pdb format databases.

New Features in 5.50

- Maximum number of databases increased to 120
- Enhanced use of color in the app compared to JFile 4
- 25 different sort and filter settings can be saved for quick and easy usage
- Easier to use, updated user interface
- Multiple locked columns for left/right scrolling in the main database view
- Encryption of databases using 64-bit Blowfish algorithm
- Easy movement of databases to VFS memory cards
- New calculated value field types
- Default values for fields, and read-only field options available
- Enhanced resolution for Handera 330 devices
- Extensive keyboard input support
- Navigation of most common areas of JFile for JogDial equipped devices

The version of JFile 5.50 included with this package is a **demo** version. It is included here so that you can try out the full functionality of **JFTconv 55**. For more information on **JFile**, please go to the Land-J Technologies web site at :

<http://www.land-j.com>

Quick Start

The first thing to do when you run JFTconv 55 is to set the default settings in Preferences. At the very least, you should set the Access version you wish to support and the Palm User to install PDB files to. We also suggest that you set Default Directories in order to enable automatic path completion in JFTconv 55.

Access to JFile

-Select the MDB file to convert. If you have set default directories, you can simply type in a filename with the .MDB extension. When you move off of the MDB File field, JFTconv 55 will automatically :

Expand the Access filename to include the appropriate path.

Enter the filename as the PDB source file, add the appropriate path, and proper-case the filename so that it appears correctly in the JFile database list

-Select the Table you wish to convert

-Enter the path to the PDB file you wish to create, if it was not automatically entered for you. If you just type in a filename, then the file will be created in the current JFTconv 55 directory. If you have set a default directory for created PDB file, the file will be placed there. You can also specify a full path and filename if you wish to save the file to a specific location. **Do not** save a PDB file into your Palm\Backup\UserName directory; **always** save to another location. The title of the JFile database, as it appears in JFile, is derived from the PDB filename. Please **be aware** that if the named PDB file already exists, it will be overwritten, without any warning confirmation.

-If you want to specify which Access fields are written to PDB, or modify the JFile fields types etc. then click the Fields button and make the required changes. If the Access table contains more than 50 fields, you must select which fields you want to write to PDB format. If you do not select them beforehand, then you will be prompted on conversion.

-If you want to set specific JFile options such as a security model or password, then click the Options button.

-Click the **Save** button to write a template of your settings for reuse at another time.

-Click **Convert** to create the PDB file. If the **Install PDB File** option was turned on, then you can simply HotSync to use the database on the Palm device.

JFile to Access

-Enter the path to the PDB file you wish to convert. Use the Browse button to the right of the PDB File entry to locate the PDB file. By default, the Windows file dialog box will open on the Backup directory for the select Palm User. If you have set default directories, you can simply enter a filename (with extension), and JFTconv 55 will automatically :

Expand the JFile filename to include the appropriate path.

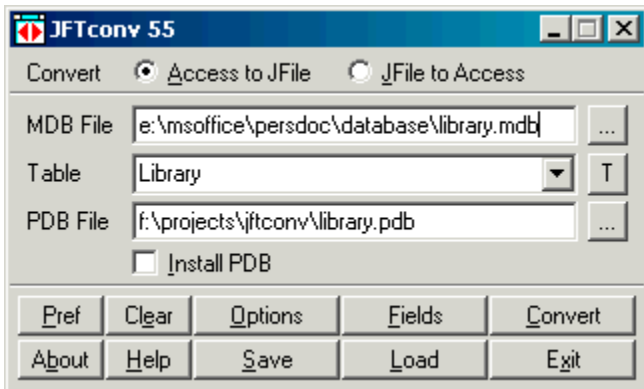
Enter the filename as the MDB source file and add the appropriate path.

-Select the MDB file to save the PDB data into, if it was not automatically entered for you. Please **be aware** that if the specified MDB file already exists, it will be overwritten, without any warning confirmation. If it does not exist, it will be created in the specified path.

-Select the specific Table you wish to write to. If the table does not exist, it will be created. If the table exists, JFTconv 55 will delete all of the current records and replace them with those from the PDB file, unless the **Add to MDB** option is set.on.

-Click **Convert** to write the PDB data to the Access file.

Main Screen



Pref	Set default JFTconv 55 settings and preferences
Clear	Clear the current entries
Options	Display the Options dialog box
Fields	Select the Access fields to convert
Convert	Convert the selected file
About	Display the About Box
Help	Display this Help file. You can also press F1
Save	Save a Template File
Load	Load a Template File
Exit	Exit JFTconv 55

Convert

Set the direction of the conversion, either Access to JFile or JFile to Access.

MDB File

Set the Access MDB file to be created or used as the source date to write to a PDB file. You can type in the path and filename directly, or use the **Browse** button to the right to select the file from the Windows standard file dialog box.

Table

Set the name of the Access table to read from or add records to. When you select an Access MDB file in the previous setting, the Table setting will display a list of available tables in the file. You can either select an existing table or type in the name of a new one you wish to be created. At any time you can refresh the table list by clicking the **T** button to the right of the pull-down list.

PDB File

Set the JFile PDB file to be created or used as the source date to write to an Access MDB file. You can type in the path and filename directly, or use the **Browse** button to the right to select the file from the Windows standard file dialog box.

When converting Access to JFile, the title of the JFile database, as it appears in JFile, is derived from the PDB filename. The title is taken directly, so if you want to add capitalization etc., type in the filename as you want it to appear on the Palm. For example, 'Sales Results.pdb' would appear in JFile as 'Sales Results'.

The following option is **only** enabled when the **Access to JFile** conversion option is selected.

Install PDB

This option instructs JFTconv 55 to install the created PDB file to the Palm user specified in Preferences. This option is disabled if the Palm software is not installed on your system.

The following two option are **only** enabled when the **JFile to Access** conversion option is selected.

Add to MDB

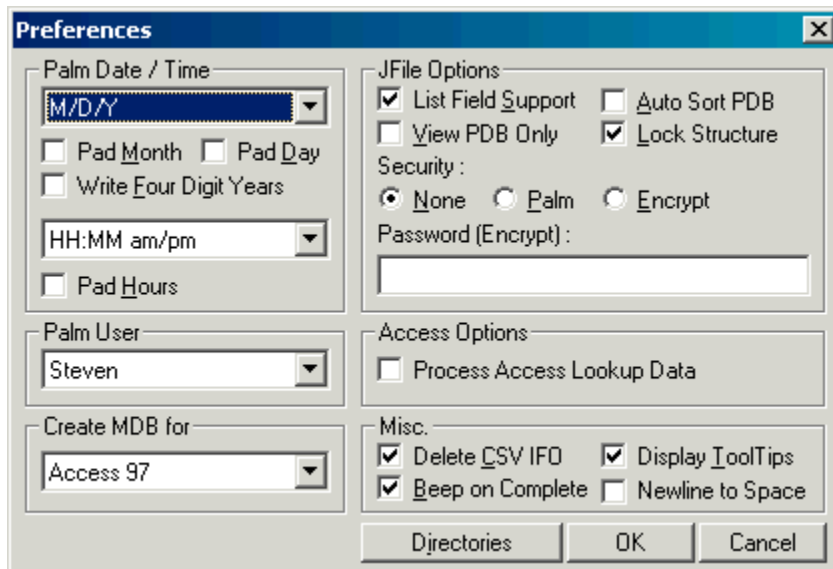
This option instructs JFTconv 55 to add the PDB records to the specified table.

Remove Duplicates

This option instructs JFTconv 55 to remove any duplicate records from the specified table after the table has been created or you have added records using the **Add to MDB** option.

Preferences

The **Preferences** settings determine the **default** settings for various functions in JFTconv 55. The JFile settings can be overwritten on a file by file basis using the Options button, for use with Template files.



Directories Set Default Directories
OK Accept Changes and Close dialog box
Cancel Discard Changes and Close dialog box

Palm Date / Time

Date Format

Set the format for Access dates and times when creating the JFile PDB file. This setting should match the Date Format setting used in Preferences on the Palm.

Pad Month

Instructs JFTconv 55 to write date information to JFile with padded, 2 digit, months. By default this option is **off**.

Pad Day

Instructs JFTconv 55 to write date information to JFile with padded, 2 digit, days. By default this option is **off**.

Write Four Digit Years

Instructs JFTconv 55 to write date information to JFile with 4 digit years. By default this option is **off**.

Time Format

Set the format for Access times when creating the JFile PDB file. This setting should match the Time Format setting used in Preferences on the Palm.

Pad Hours

Instructs JFTconv 55 to write time information to JFile with padded, 2 digit, hours. By default this option is **off**.

The above Pad options are **off** by default in order to reduce the size of the PDB file created by JFTconv 55. JFile stores all data as text, so these options simply reduce the number of characters in the file when unchecked.

Palm User

The pulldown list enables you to select the Palm User who will receive any installed PDB files. Only one user name will appear in the list unless multiple Palm users have been setup to share the Palm software. This option will be disabled if the Palm software has not been installed on your system.

Create MDB for

JFTconv 55 can read MDB files in any Access format. This option allows you to select the Access format used when creating a new Access MDB file from a JFile PDB database. The available choices are :

Access 2000 (or later), 97, 95, 2.0, 1.1, 1.0

The Access 2000 option is not available in JFTconv 97, the Access 97 version of the converter.

JFile Options

Auto Sort PDB

Set Auto Sort to automatically sort the PDB on the Palm device. By default this option is **off**.

View PDB Only

Make PDB database view only so it cannot be edited. By default this option is **off**.

Lock Structure

Lock data structure from modifications in JFile. This option is invaluable if you have to frequently transfer data back and forth between an Access MDB and a JFile database because it ensures that the JFile structure will always match the MDB as it cannot be modified in JFile. For this reason, the option is **on** by default.

Security

These settings determine the **default** security used for any PDB file you create from an Access MDB source. All of these settings can be overwritten by command line switches, with the exception of the **password** used by the Blowfish encryption.

None

Remove any security on the Palm device. Anyone can open the database in JFile.

Palm

Use the built in Palm security to protect the database. The device password must be entered to view the database.

Encrypt

Protect the database with JFile's Blowfish encryption. This is the most secure option, as a password is required to open the database and the contents are encrypted when the file is not open.

Password

The password entry is only enabled when the **Encrypt** option is turned on. The password you enter here will be used for **all** PDB files you create with Blowfish encryption enabled. The converter will also use this password to open an encrypted PDB file that you are converting to Access format. If the PDB password is different, you will be prompted for the file password.

Access Options

Process Access Lookup Data

This option enables or disables the support for Access Lookup data. By default this option is turned **off** as the processing is usually a one time action. Please see the Access Lookup Support topic for detailed information.

Misc

Delete CSV and IFO

By default this option is **on** and JFTconv 55 will delete the CSV and IFO files created by the JFile converter whenever a conversion is successful. If you want to preserve the CSV and IFO files that are created in the same directory as JFTconv 55, then turn this option **off**.

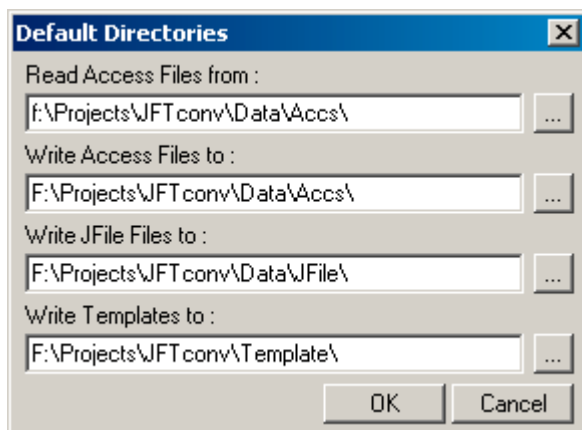
Beep on Complete

By default this option is **on** and JFTconv 55 will beep when a conversion has completed. This would make it easier for a user to know that the conversion has finished when running in command line mode from an icon.

Newlines to Space

By default, JFTconv 55 processes newline characters (carriage return / linefeeds) in your text type fields to ensure that the newlines are transferred properly between the two platforms. You can have JFTconv 55 convert the newlines to spaces if you prefer to strip them out on conversion.

Default Directories

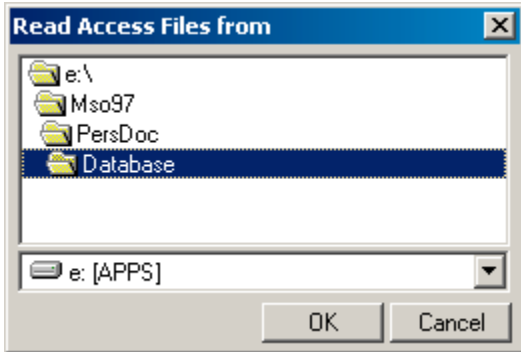


The **Default Directories** box allows you to set distinct paths for where Access files are read from and written to, and where PDB files are written to. JFTconv 55 always assumes that JFile PDB files are read from the **Backup**

directory of the currently selected Palm User.

By setting default directories, JFTconv 55 can automatically enter paths for you, and save the hassle of finding the file with a Windows file dialog box. For example, if you are entering an Access filename, you can simply type in the name and extension, and JFTconv 55 will automatically add the correct path when you move off of the MDB File field. This automatic path completion also applies to command line filename arguments.

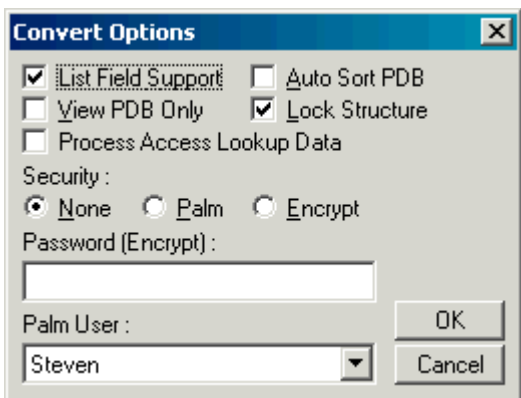
To set the paths, either type them in or select them with the **Browse** button to the right of each path field.



In the Browse box, navigate to the directory, double click on the directory name, and click OK to update the Default Directory entry.

Convert Options

The **Convert Options** settings **only** apply to the current PDB file you are creating. By default, these settings are set to the same settings you assigned in Preferences. The **Options** box provides a way of changing the JFile settings on a file by file basis, to be saved in a Template file.



OK Accept Changes and Close dialog box
Cancel Discard Changes and Close dialog box

Auto Sort PDB

Set Auto Sort to automatically sort the PDB on the Palm device.

View PDB Only

Make PDB database view only so it cannot be edited.

Lock Structure

Lock data structure from modifications in JFile. This option is invaluable because it ensures that the JFile structure will always match the MDB as it cannot be modified in JFile.

Process Access Lookup Data

This option enables or disables the support for Access Lookup data. Please see the Access Lookup Support topic for detailed information.

Security :

These settings determine the security used for the PDB file you create from an Access MDB source.

None

Remove any security on the Palm device. Anyone can open the database in JFile.

Palm

Use the built in Palm security to protect the database. The device password must be entered to view the database.

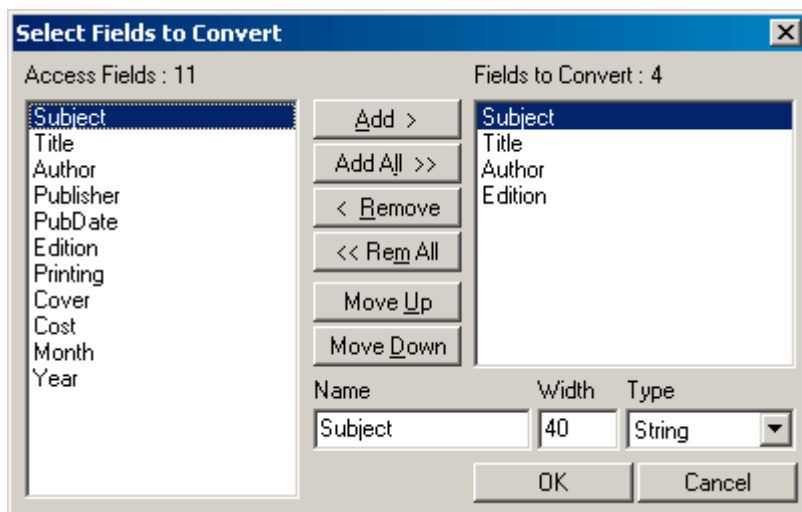
Encrypt

Protect the database with JFile's Blowfish encryption. This is the most secure option, as a password is required to open the database and the contents are encrypted when the file is not open.

Password

The password you enter here will be used for the specific PDB file you create with Blowfish encryption enabled.

Select Fields



Add	Add Highlighted Field to the Convert List
Add All	Add All Field to the Convert List
Remove	Remove Highlighted Field from the Convert List
Rem All	Remove All Fields from the Convert List
Move Up	Move Highlighted Field Up
Move Down	Move Highlighted Field Down

OK	Save the Changes and Close the Dialog Box
Cancel	Discard the Changes and Close the Dialog Box

The **Select Fields** box allows you to select the specific fields you wish to convert, or modify the field names, display widths, and data types in the converted file. **Select Fields** is only available when converting from Access to JFile PDB.

The **Access List** on the left contains all of the fields in the selected Access file / table. The **Convert List** on the right contains all of the fields that will be converted, and the sequence in which they will appear in JFile on the Palm. Highlight a field in the Access List, and use the **Add** functions to add them to the Convert List. Highlight a field in the Convert List and use the **Remove** functions to exclude them from the conversion. Use the **Move** functions to change the sequence of a highlighted field in the Convert List.

To modify a field, highlight the field in the **Convert List** on the right. The **Name**, **Width**, and **Type** boxes update to describe the highlighted field. You can edit the field name or width as you wish. The **Type** pulldown list will display possible data types that you can convert to. These types primarily allow you to set a specific JFile data type that is not directly supported in Access. For example, a numeric value could be either an integer or a floating point number, a text field could be a JFile popup list or a multi-list type. The **Type** pulldown list will **only** display these alternate types, based on the original Access field type.

When the required fields have been selected and modified, click the **OK** button to save your changes and continue.

If you want to be able to convert back and forth between a JFile database and Access, then do not modify the field names. If you do, JFTconv 55 will not be able to match the fields when you go to convert the PDB back into the original Access table.

List Field Support

The JFile Popup and Multi-List field types are very useful to speed up data entry and avoid inconsistent entries or typos. JFTconv 55 can support JFile Popup and Multi-List field types by saving the list data into separate tables in an Access MDB file. This support can be enabled or disabled with the **List Field Support** option in Preferences and Convert Options. By default, **List Field Support** is enabled in JFTconv 55.

The status of the **List Field Support** option is saved to a Template, so you can enable / disable the processing on a file by file basis.

List Field Support can also be set from the command line with the **/l** switch.

JFile to Access

When you convert from a JFile PDB to an Access MDB file, any popup or multi-list field data is preserved in distinct tables in the Access file. These tables are named **xJList#**, where **#** represents the position of the field, starting at Field 1. If the Access MDB file already contains **xJList#** tables, then they are deleted before the JFile data is read into the Access file. This ensures that the field data will always match the current structure and contents of the JFile database.

Access to JFile

When you select an Access MDB file to convert, JFTconv 55 automatically checks the file for any **xJList#** tables. If found, the associated Access field is tagged as a JFile Popup type field for conversion. When the JFile database is created, the **xJList#** table contents is written to the appropriate field list in JFile. If you need to set the list field as a Multi-List type, then you **must** use the Select Fields function to set the specific field to that type. JFTconv 55 will **always** set the field to a Popup type unless you change it in this manner.

Important !

Because the relationship of a list field to its data is determined by the field position in the database, it is essential that you **do not insert or delete fields in the Access table**. Doing so will break the link between the field and the data, and the conversion will not happen correctly. If you have to modify the structure, it is better to do so in JFile on the Palm device and then convert the PDB file to an Access MDB file. The Access table will then reflect the structure changes and assign the list fields / tables correctly.

If you delete a list table in the Access file, JFTconv 55 will simply set the associated field to short string (less than 250 characters) type. If you want keep the field as a Popup type in JFile, delete the **contents** of the **xJList#** table, and **not** the table itself.

JFile imposes limits on the amount of data that a list can contain. Access does not have any limit, so if you want to add records to a **xJList#** table you can, but please stay within these limits. JFile restricts lists to 100 items, a limit that is enforced by JFTconv 55. If you add more than 100 records to a list table, they will be discarded on conversion to PDB. JFile limits the amount of text that a list can contain to around 4,000 characters, so please do not add long strings to a list as they could likely get corrupted on conversion, or later items could be discarded due to the excessive character limit.

Access Lookup Support

A field in an Access file can contain lookup data that is displayed in a combo or pulldown type list. This list can be populated from another field in another table (Table/Query type), or from an entered list of items (Rowsource). When you build a JFile database from an Access file, you can have the converter set the associated JFile field to a Popup type and include this data in the popup for use in JFile.

The Access Lookup support can be enabled by default in Preferences, or on a file by file basis in the Convert Options dialog box.

When enabled, JFTconv 55 identifies any fields in the Access table with Lookup assignments, sets the associated JFile type to popup, and fills the JFile list with your popup data. This lookup processing **only** applies to Access Text fields, as JFile cannot handle other data types in popups. If your popup list contains more than 100 items, the converter will only write out 100 items, the JFile limit.

This processing can handle items from a Rowsource or a simple Table type lookup. It **cannot** process Access Query lookups. If the lookup type is a Query, the associated JFile field will still be set to a Popup type, but the list will not be populated from the Access data.

The Access Lookup support **only** applies when you create a PDB file from an Access MDB. The converter cannot create Access lookups from JFile popup data as the automation is simply not there in the data engine used by JFTconv 55. The function is primarily designed to enable you to create a new JFile database from an Access MDB with lookups. Once the PDB is created, the popup data will be converted back to the MDB as an xJListx table and will be maintained properly thereafter. If you use the Access file exclusively with JFile, then it is better to create the PDB file, convert the PDB back to Access, and change the lookup fields in the Access MDB to map to the appropriate xJListx table. The Access file will then present the identical popup data as the JFile version.

Templates

You can save the current option and fields selections to a **Template** for reuse. If you frequently convert the same files, this can be a huge time saver.

At any time, you can click the **Save** button, use the **Alt-S** keyboard shortcut, to save the current status of the

converter. A template is a simple ASCII text file that is written to your hard drive, and defines the current options and field selections and modifications. A template uses two different file extensions, depending on the direction of the conversion. If you are converting from Access to JFile, and **.ATJ** extension is used. If you are converting from JFile to Access, a **.JTA** extension is used. The advantage of the two extensions is that you can save the settings for a conversion of an Access file to JFile, and the conversion back to Access, with the same base name. For example, if you have a Birthdays database in JFile, you can create a BIRTHDAYS.JTA template to define the conversion from JFile to Access, and a BIRTHDAYS.ATJ template to convert in the opposite direction. JFTconv 55 presents an appropriate list of available template files in the Windows file dialog box, depending on the current **Convert** option.

To use a template, simply click the **Load** button or use the **Alt-L** keyboard shortcut, and select the .ATJ or .JTA file to load. The current settings will be replaced by the contents of the template file. Please be careful as you are **not** prompted before the current settings are overwritten.

You should **never** modify the .ATJ or .JTA files with a text editor, such as the Windows Notepad. Doing so may break the file for use in JFTconv 55.

Command Line

The JFTconv 55 converter runs transparently and converts between Access MDB and JFile PDB formats. The converter determines the direction of the conversion based of the sequence of files provided on the command line.

To provide maximum flexibility, we strongly suggest than you create a Template file for each conversion that you do on a regular basis. You can then specify the template file on the command line instead of using command line switches. The other advantage to templates is that you can easily test the conversion in JFTconv 55, and when you know it works as you want, just call it from the command line.

Template Syntax

JFTconv "Template File"

The 'Template File' option must be provided as a complete path to the .ATJ or .JTA file, enclosed in double quotes. JFTconv 55 will load in the template file and convert based on the setup information it contains. This command line option is the preferred way to convert frequently used files, as it allows you to select specific fields and the order they appear in the database, as well as supporting specific JFile setting.

Filename / Option Syntax

The Access MDB to JFile PDB sequence is :

```
JFTconv "MDB-Filename" "MDB-Tablename" "PDB-Filename" "PalmUser"
```

The JFile PDB to Access MDB sequence will be :

```
JFTconv "PDB-Filename" "MDB-Filename" "MDB-Tablename"
```

These elements **MUST** be enclosed in double quotes to avoid any long filename with spaces nonsense. The file names should be fully qualified with a complete path. Each element must be separated by a quote space quote in order for the utility to parse the command line.

If you have set default directories, you can eliminate the path and just enter the filename (with extension), between double quotes. JFTconv 55 will see that the path is missing and use the appropriate default directory.

The 'Palm User'" element should be the **exact** name of the user, as displayed in the JFTconv 55 Preferences dialog box list. If the element is omitted from the command line, then the default user set in Preferences will be used. If you want to install to all Palm Users installed on the system, then use "All" as the element. This element will be completely ignored if the **/i** switch is not included on the command line.

Option Switches

There are also nine option switches. The options can be grouped before the first file name or after, such as :

JFTconv **/c/e/a/d/i/l/ns/ps/bf/97** "MDB-Filename" "MDB-Tablename" "PDB-Filename" "PalmUser"

or

JFTconv "MDB-Filename" "MDB-Tablename" "PDB-Filename" "PalmUser" **/e/a/d/i/l/ns/ps/bf/97**

The option switches are :

/e - Display an error box with any error messages

/c - Display a 'Completed' dialog box where conversion has completed

/a - Add PDB records to an existing MDB table.

/d - Remove duplicate records when adding to an existing MDB table from PDB.

/i - Install the created PDB file after the conversion.

/l - Enable List Field Support

/ns - Do not protect the database at all.

/ps - Protect the database with Palm security

/bf - Protect the database with Blowfish encryption

/Vers - Create MDB file in specified Access format.

This option allows you to determine the format in which the Access MDB file is created if JFTconv 55 is building a new MDB file. JFTconv 55 uses the setting from Options by default, and the command line option overrides the default for the current run only. The switch setting does **not** become the new default.

The switches are :

/2000 - Access 2000 or greater

/97 - Access 97

/95 - Access 95

/2.0 - Access 2.0

/1.1 - Access 1.1

/1.0 - Access 1.0

The **/e**, **/a**, and the Access version switches only apply to the PDB to MDB conversion, and the **/i**, **/ns**, **/ps** and **/bf** switches to MDB to PDB conversion. The **/i** switch will only work if the Palm software is installed on the system. If it is not, then the switch will simply be ignored.

The / 2000 option is not available in JFTConv 97, the Access 97 version of the converter.

The /I List Field Support switch is only required if you have the option turned off by default in Preferences.

Be careful to only use one of the security options, /ns, /ps and /bf. If you use more than one, the last option in sequence will be used.

When the converter runs, you will only see the Windows hourglass cursor. The utility has no visible display and it does not appear on the taskbar. It simply does its thing and unloads from memory. If an error is encountered, it beeps and writes the error message to a text file named **JFTCONV.ERR** in the same directory as the utility. For testing purposes, I suggest that you use the /e switch to divert the messages to a dialog box. This will save the hassle of endlessly opening the ERR file. The **JFTCONV.ERR** file is deleted every time the utility runs so that its existence may serve as a means of knowing that an error occurred.

JFTconv 55 calls the JFile converter to make the PDB file. The JFile converter generates an ERR file of its own if it finds a problem. The file, **JCONV5.ERR** will contain any JFile specific error messages, and is also deleted every time the JFTconv 55 utility is run, and created on error.

Convert or Fields

The JFTconv 55 converter provides two options to convert Access files to JFile format, a quick method using the **Convert** button, and a manual method using **Fields**. The choice you make depends on what you want to do and on data type limitations.

If you simply click **Convert**, the converter assumes that there are 50 or fewer fields in the Access table, and that the data types map directly to their JFile counterparts. For example, there are no List or Time types in Access. In Access, a List is just a Text type, and Time is just a Date type where the display format is set to show the time portion of the date. These field types **cannot** be converted automatically to JFile's List and Time types. You must use the Fields function to change the corresponding JFile types before you click **Convert**.

If your Access database only contains numeric, text, and date type fields and you do not need JFile List or Time types, then **Convert** is more direct. It will convert every field in the Access table and map it to the corresponding JFile data type.

If you need to select fields to convert and or change the corresponding JFile data types, then you **must** use Fields **before** you convert the file.

Data Translation

Due to the fairly large differences between the Access and JFile database engines, JFTconv 55 makes some assumptions about the type of data and how to translate it.

JFile PDB to Access MDB

JFTconv 55 converts the data as follows when converting from PDB to MDB:

JFile	JFTconv 55
String	Text (255 Characters Max)
String	Memo (10,000 Characters Max)
Boolean	Boolean
Integer	Long Integer
Float	Double Precision

Date	Date Type
Time	Date Type
Popups	Text (255 Characters Max)
Auto Increment	Long Integer
Create Date	Date Type
CreateTime	Date Type
Modify Date	Date Type
ModifyTime	Date Type
MultiList	String (255 Characters Max)

All strings are created as variable length strings to reduce the MDB file size.

Because JFile has one string data type and Access two, JFTconv 55 can map JFile string fields to either of the two Access types, depending on the JFile 'always less than 250 character' string field option. If the option is set in JFile, then JFTconv 55 converts all JFile string fields to the Access Text type and truncates any long data to fit within a 250 character maximum of the Access type. If the option is not set, then JFTconv 55 will map all JFile string fields to the Access Memo type and thereby allow the full character limit supported by JFile, 10,000 characters. When creating PDB files from Access sources, the converter will automatically set the JFile string field option appropriately.

Because Access is much more rigid about naming fields etc.. JFTconv 55 will automatically strip out invalid characters from any JFile field names. The only valid characters are letters, numbers, space and the underscore. Any others will create errors in either Access or the print engine. The other rule is that the first character in a field name must be a letter. JFTconv 55 will correct this situation automatically and change an invalid character into an 'a'.

JFTconv 55 will also ensure that no duplicate field names exist in the current file, and will add letters to a duplicate name to make it unique. For example, three fields named Film will be converted to Film, Filma, and Filmb.

Access MDB to JFile PDB

JFTconv 55 converts the data as follows when converting from MDB to PDB. Because Access does not support the specific types found in JFile, JFile types such as Modify date and Popup etc, will be converted to the closest generic Access type.

JFTconv 55	JFile
Text	String
Memo	String
Boolean	Boolean
Byte	Integer
Integer	Integer
Long Integer	Integer
Single Precision	Float
Double Precision	Float
Currency	Float
Date	Date
Time	Date

When converting Memo field types to JFile, JFTconv 55 will truncate the data to 10,000 characters in order to stay within the limits imposed by JFile.

JFTconv/Access Data Type Ranges

String
Up to 250 characters

Memo

Up to 10,000 characters

Byte

0 to 255

Integer

-32,768 to 32,767

Long (long integer)

-2,147,483,648 to 2,147,483,647

Single (single-precision floating-point)

-3.402823E38 to -1.401298E-45 for negative values

1.401298E-45 to 3.402823E38 for positive values

Double (double-precision floating-point)

-1.79769313486232E308 to -4.94065645841247E-324 for negative values

4.94065645841247E-324 to 1.79769313486232E308 for positive values

Currency (scaled integer)

-922,337,203,685,477.5808 to 922,337,203,685,477.5807

Access Integration

You can easily make calls to the converter from within an Access VBA project. The following sample code illustrates how to do this. You will notice that this sample does not use the VBA **Shell** command, but instead uses a Windows API call. This is because the VBA **Shell** command just runs the command and then immediately returns control to the calling application. The **Shell32** function, provided below, waits for the process (converter) to finish before returning control.

To call the converter, just execute the **RunJFTconv** sub with the required parameters. If you do not include parameters, the converter will run in GUI mode. The subroutine will automatically pad any included parameters with double quotes.

Declarations

Option Explicit

```
Private Declare Sub Sleep Lib "kernel32" _  
    (ByVal dwMilliseconds As Long)
```

```
Private Declare Function OpenProcess& Lib "kernel32" _  
    (ByVal dwDesiredAccess As Long, _  
    ByVal bInheritHandle As Long, _  
    ByVal dwProcessId As Long)
```

```
Private Declare Function GetExitCodeProcess& Lib "kernel32" _  
    (ByVal hProcess As Long, _  
    lpExitCode As Long)
```

```
Private Const STILL_ACTIVE = &H103
```

```
Private Const PROCESS_QUERY_INFORMATION = &H400
```

RunJFTconv Subroutine

Sub RunJFTconv(Optional Parm1 As String, Optional Parm2 As String, Optional Parm3 As String, Optional Options As String)

```
Dim Cmd As String
Dim Qt As String
```

```
On Error GoTo RunErr
```

```
'Double Quote Character
Qt = Chr$(34)
```

```
'Get Full Path and EXE name to Converter
Cmd = GetSetting("JFTconv", "Startup", "Converter", vbNullString)
```

```
If Cmd = vbNullString Then
' No Path Returned
MsgBox "Converter Path Not Found", vbExclamation, "Shell Error"
Else
' Path Returned
' Assemble Command Line
' Wrap each Parameter in Double Quotes
If Parm1 <> vbNullString Then Cmd = Cmd & " " & Qt & Parm1 & Qt
If Parm2 <> vbNullString Then Cmd = Cmd & " " & Qt & Parm2 & Qt
If Parm3 <> vbNullString Then Cmd = Cmd & " " & Qt & Parm3 & Qt
If Options <> vbNullString Then Cmd = Cmd & " " & Options
' Run the Command Line
' You can set the way that the converter is displayed by using one
' of the VB constants below. It is usually best to use the Normal
' focus setting as the default.
' vbHide (0)
' vbNormalFocus (1)
' vbNormalNoFocus (4)
Call Shell32(Cmd, vbNormalFocus)
End If
```

```
Exit Sub
```

```
RunErr:
```

```
MsgBox "(" & Str$(Err.Number) & ")" & Err.Description, vbExclamation, "Shell Error"
```

```
End Sub
```

Shell32 Subroutine

Sub Shell32(ByVal Cmd As String, Mode As Integer)

```
Dim hProcess As Long
Dim RetVal As Long
Dim Hndl As Integer
```

```
Hndl = 0
```

```
hProcess = OpenProcess(PROCESS_QUERY_INFORMATION, Hndl, Shell(Cmd, Mode))
Do
  GetExitCodeProcess hProcess, RetVal
  DoEvents
  Sleep 100
Loop While RetVal = STILL_ACTIVE

End Sub
```

What's New

This version of JFTconv 55 supports JFile 5.5, the newest release. While the format of 5.5 databases is very similar to the older 5.x release, there are a few new changes that require consideration.

This version adds support for newline (carriage return / linefeed) characters in the text field types. The newlines will now be maintained on conversion. You can optionally turn off this feature and strip the newlines out to spaces if you prefer.

This release contains stronger PDB file checking and will verify the version of the PDB before trying to convert from PDB to Access MDB. If the file is not in JFile 5.0 or 5.5 format, an error will be generated.

This release can import dates set to International format in the PDB file, but will convert them to the set Palm Date Format on conversion to Access format.

JFile 5.5 Lookup fields are ignored on conversion to Access. The data itself will convert properly, but the field will NOT be set back to Lookup on conversion from PDB to MDB.

JFTconv 55 can now build a JFile popup list from an existing Access Lookup. If you have a Lookup field in an Access file, JFTconv 55 will find it, build a matching Popup field in the JFile database, and populate the list to match the Access data. Please review the Access Lookup Support topic for details.

This version **modifies** the Template structure, and your older templates will **not** load properly into JFTconv 55. If a template is very complicated, you can modify the .ATJ or .JTA file by opening it in Notepad. Edit the first line to read vr.1.1 instead of vr.1.0. Move down to the [Options]] header. Below the header you will see 5 lines of the numbers 0 or 1. Insert a new line after the 4th. number and type in a 0. Save the file, and it should now load properly in JFTconv 55. **Do not** use Word or similar word processor as you will damage the file. Always use Notepad to edit any templates.

Bug Fixes :

The field sequence is now correctly written out to the JFile PDB file. Previously the selected fields were managed properly but the sequence was ignored.

The Popup list handling now screens out some other list type data in JFile that was being pulled into a popup list when it should not have.