



Document Converter (DCT), Version 2.10 User's Guide

Table of Contents

Introduction	3
Installation	4
Architecture	5
Testing Scenarios	7
File Listing	9
File Associations	10
DCT Engine API	11

If you have any comments or wish to see additional features, please contact:

Metasoft Systems, Inc.
Tel: (604) 683-6711
Fax: (604) 683-6704
Email: info@meta-soft.com

Copyright © 1998-2004 Metasoft Systems, Inc

Introduction

This Imager Document Converter (DCT) demonstrates the use of the Imager Print Capture Driver (PCD) to convert from any 'printable' file format to a Fax Image format.

A brief history of document conversion solutions can be summarized as follows:

- Build a custom converter for all file formats
- Get everyone to use the same format
- Use the native application to do the file conversion

Building custom file converters (like Inside-Out), or getting everyone to use one common file format (like PDF) does not address the fact that many documents are WORD or EXCEL in origin. For Windows based documents (true-type, graphics, extremely-complex, rapidly changing file formats) the most efficient (time/cost) rendering solution is to somehow use a Windows machine.

Recognizing that all back-end WEB Servers are not always Windows Servers, we propose the following solution:

1. WEB Server accepts a 'file' for rendering from Web
2. WEB Server connects to pipe on a DCT Server on local network
3. DCT Server accepts a file and to render it
4. DCT Server passes completion code back up pipe to WEB Server.

The WEB Server can connect with more than one DCT Servers. Basically a DCT Server is a NT based Windows machine. File conversion are processor intensive and not something you want to bog down a WEB Server with.

The DCT controls file conversion to TIFF by first launching the file native application (e.g. Word, Excel, IE), then instructing the native application to print the document via Windows shell commands.

Our experience is that no one utility can convert all file formats. However, most applications have a mechanism used by the Windows shell to support file printing.

The PDFilCnv.exe application is a console application that is launched with a command line. The command line includes the name of the source file (among other things). The PDFilCnv.exe application terminates with a 'number of pages' printed return code, or a negative value if there was an error. This way, even if a document conversion does fail (due to some required user intervention); other conversions can still take place. In addition, a Windows Daemon application (WPDaemon.exe) is provided to support file conversions from a NT service application (such as a CGI script running on a WEB Server) or from a multi-threading application.

To convert multiple documents in a folder, run the PDDirCnv.exe application. The PDDirCnv.exe application repeatedly calls the PDFilCnv.exe application, and is a superset program.

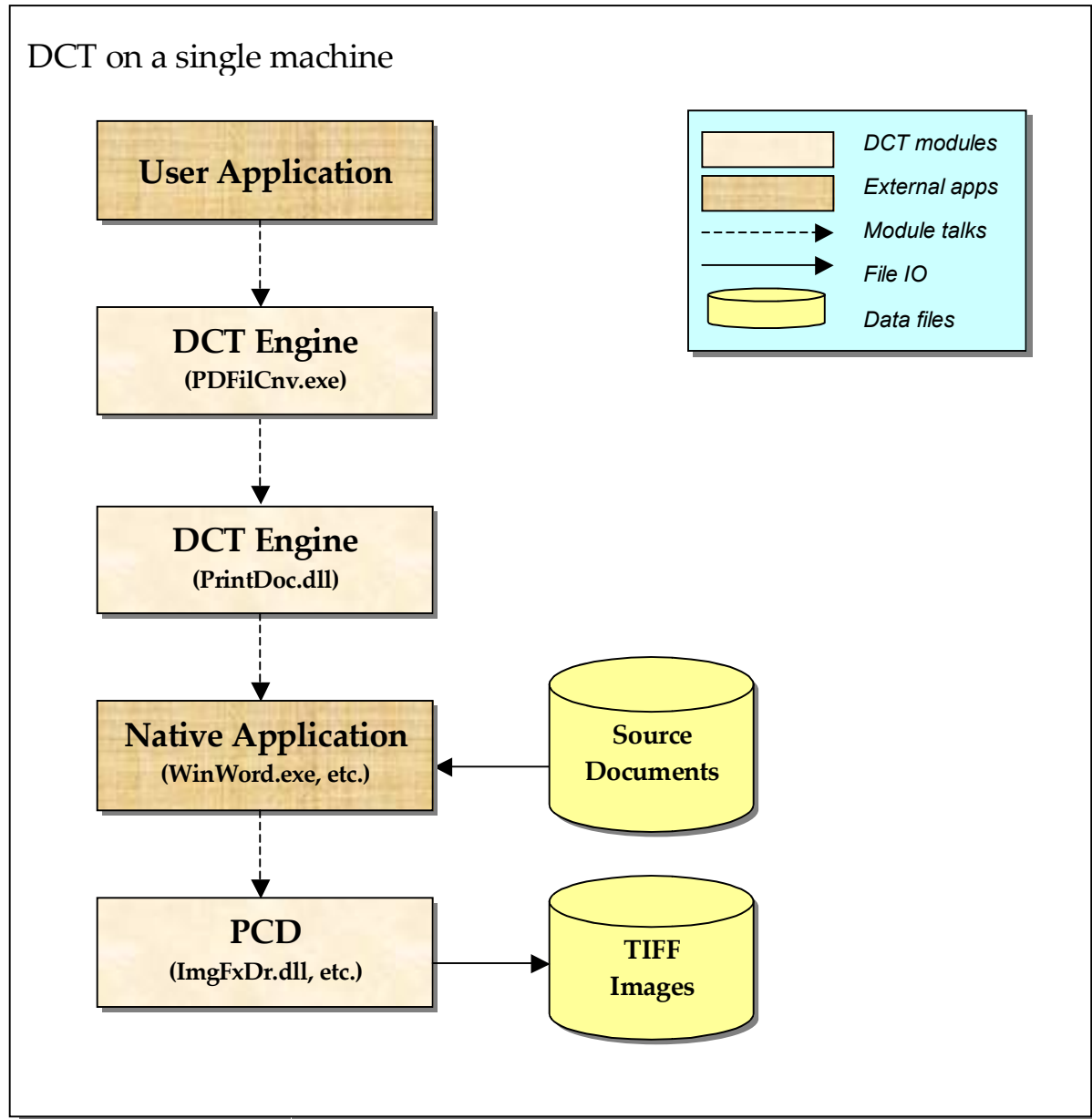
We have tested this conversion process with many document types, including Word, Excel, PowerPoint, RTF, Adobe PDF, HTML, Windows BMP, JPEG, AutoDesk DWG, PostScript, etc.

Installation

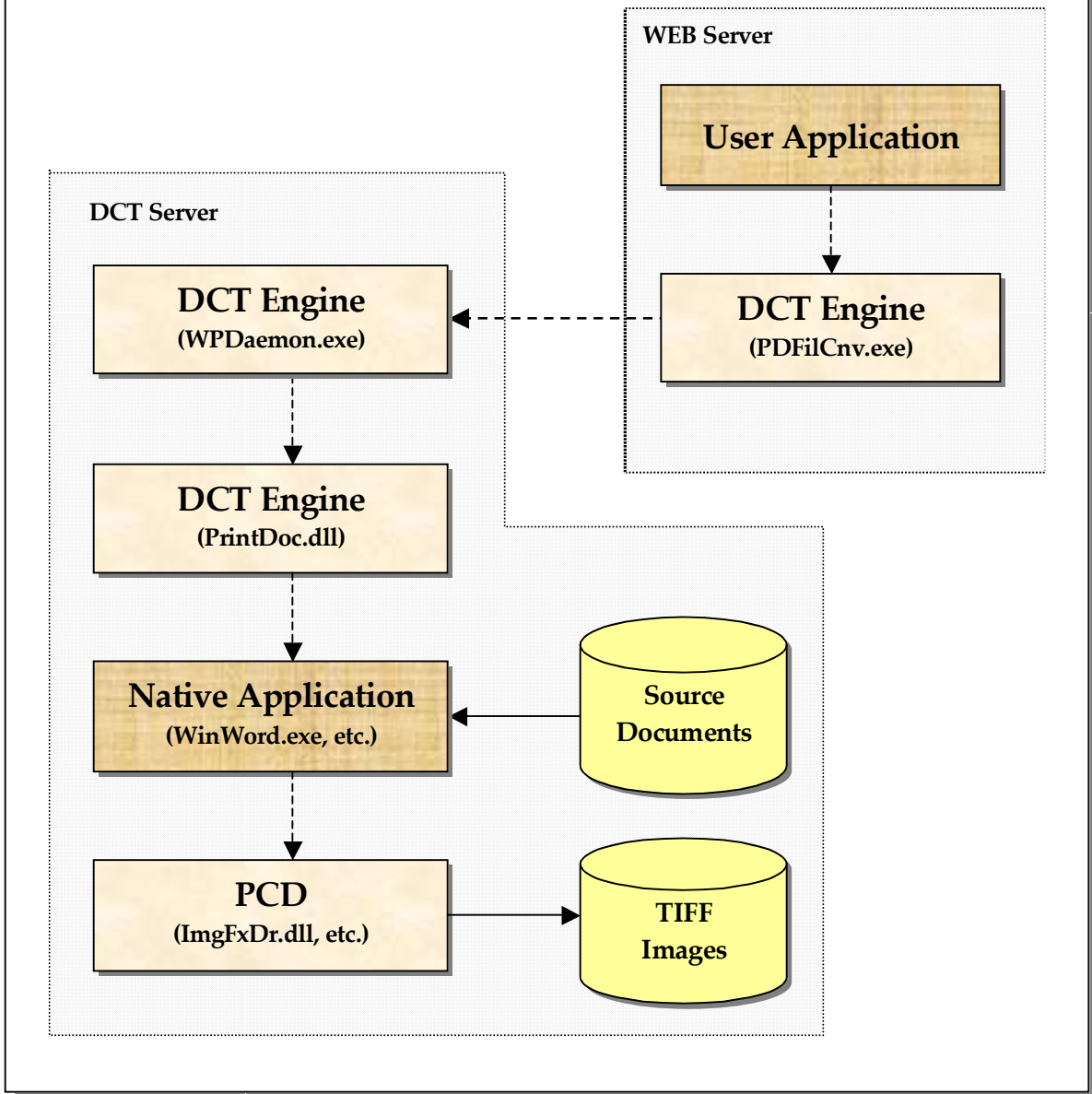
1. System requirements:
 - Platform: Windows NT/2000/XP/2003
 - Software Pre-Installed:
 - i. Microsoft Office 2000 or up (for converting office document)
 - ii. Acrobat Reader 4.X or up (for converting PDF document)
 - iii. Kodak Imaging or IrfanView (for converting Image document)
 - iv. Other native applications for various file formats
2. If you have already installed any of these products, please uninstall it:
 - Metasoft Imager Document Converter
 - Metasoft Imager Folder to Folder Conversion Server
 - Metasoft Legal Imager
3. Run Setup.exe in the DCT package and follow the directions.
4. Set the default printer to "Imager SDK DocCnvert Driver".
5. Setup and verify the file associations for each file format. Please check the "File Associations" section.

Architecture

The following diagrams show the relationship between Document Conversion components.



DCT on a remote machine



Testing Scenarios

◇ Single machine testing scenarios ◇

- The single machine can be one of these platforms:
 - Windows NT 4 Workstation
 - Windows NT 4 Server
 - Windows 2000 Professional
 - Windows 2000 Server
 - Windows XP professional
 - Windows Server 2003
- Your application and the DCT will all be installed on this machine.
- Scenario 1 is ideal for single thread applications.
- Scenario 2 is ideal for multiple threads or service applications.

Scenario 1 - Testing without Daemon on a single machine:

1. Start a command line prompt.
2. Change to the directory in which you installed the DCT.
3. Type command:
PDFilCnv guide.doc /fc:\Imager\guide1.tif
4. Check the output TIF file in C:\Imager with a TIFF Viewer.

Note: Your application has two ways to call DCT:

- Shell execute PDFilCnv.exe
- Call Printdoc.dll via the DCT API
VC/VB sample source will be provided by Metasoft on request.

Scenario 2 - Testing with Daemon a single machine:

1. Start the DCT Daemon.
2. Start a command line prompt.
3. Change to the directory in which you installed the DCT.
4. Type command:
PDFilCnv guide.doc /fc:\Imager\guide2.tif /dr
5. Check the output TIF file in C:\Imager with a TIFF Viewer.

Note: If you want to call DCT in a multi-threading way, you need the Daemon started. The Daemon will only allow one conversion running at a time.

◇ Remote DCT Server testing scenarios ◇

- The machine on which you installed DCT will be known as the DCT Server. The machine on which you'll run PDFilCnv.exe will be known as the WEB Server.
- Please install DCT on the DCT Server and copy PDFilCnv.exe to the WEB Server.
- If the source document is not on the DCT Server, you need to put it in a shared folder on your network. Assuming here we have a shared folder \SharedDir on computer \\SharedPC.
- If you want to save the output files to a computer other than the DCT Server, you need to configure the DCT driver to be able to access a shared drive on your network. To do that on the DCT Server:
 - Open Start | Settings | Printers | Imager SDK DocCnvt Driver | Properties | Device Settings | Access specific share on server.
 - Change "Use specific server account" to "Enabled"
 - Apply a user name and password. This user name (in <domain\account> format) should have the WRITE privilege to the \\SharedPC\SharedDir.

Scenario 3 – Remote DCT Server testing by specifying the WEB Server

1. Start the DCT Daemon on the Conversion Server.
2. In the DCT Daemon main window press the 'Stop' button to stop the service and select "Configure" from the "Service" menu. Enter the name of the WEB Server. Press 'OK' then press the start button to start the service.
3. On the WEB Server, change to the directory in which you copied PDFilCnv.exe and type command:
PDFilCnv \\SharedPC\SharedDir\guide.doc /f\\SharedPC\SharedDir\guide3.tif /dr
4. Check the output TIF file in \\SharedPC\SharedDir with a TIFF Viewer.

Note: In this case, when you run the Daemon it writes its pipe name to the registry key HKLM\\Software\\ImageMaker\\PrintDoc\\PipeNames on the WEB Server machine. When you run PDFilCnv.exe with the /dr option it looks to this registry key for available pipes. When the Daemon exits it removes its pipe name from the registry.

Scenario 4 - Remote DCT Server testing by specifying the DCT Server's IP

1. Start the DCT Daemon on the DCT Server.
2. On the WEB Server, change to the directory in which you copied PDFilCnv.exe and type command:
PDFilCnv \\SharedPC\SharedDir\guide.doc /f\\SharedPC\SharedDir\guide3.tif
/d:123.456.789.012
where 123.456.789.012 is the IP address of the DCT Server.
3. Check the output TIF file in \\SharedPC\SharedDir with a TIFF Viewer.

Note: This would be useful if you have multiple WEB Servers or multiple DCT Servers.

File Listing

File name	Component	Description
WPDAEMON.EXE	DCT engines	Daemon process (for remote server conversion)
PDFILCNV.EXE	DCT engines	Single file conversion utility
PDDIRCNV.EXE	DCT engines	Folder conversion utility
PRINTDOC.dll	DCT engines	Conversion Engine DLL
MetaTif2Pdf.dll	PDF Conversion	Convert a TIFF to non searchable PDF file
ImgFaxDr.dll	PCD	Windows NT4 driver
ImgFaxMn.dll	PCD	Windows NT4 driver
ImgFaxUI.dll	PCD	Windows NT4 driver
ImgFx5Dr.dll	PCD	Windows 2000 driver
ImgFx5Mn.dll	PCD	Windows 2000 driver
ImgFx5UI.dll	PCD	Windows 2000 driver
ImgFx5Mu.dll	PCD	Windows 2000 driver
ImgFx6Dr.dll	PCD	Windows XP/2003 driver
ImgFx6Mn.dll	PCD	Windows XP/2003 driver
ImgFx6UI.dll	PCD	Windows XP/2003 driver
ImgFx6Mu.dll	PCD	Windows XP/2003 driver
Img32mfx.dll	PCD	Driver communication DLL
PdDlg.EXE	PCD	Driver control dialog
ImgFax.hlp	PCD	Driver help file
Readme.txt	Documents	DCT user guide in text format
Guide.doc	Documents	DCT user guide in Word format
Products.doc	Documents	Imager products guide
Setup.exe	Setup programs	Main setup program
Setup.ini	Setup programs	Setup INI script
ImgInstl.dll	Setup programs	Driver installation DLL

File Associations

File associations can be changed at:

Explorer | Tools | Folder Options | File Types | <format> | Advanced | Printto

Below are some of the possible native applications that can be associated with a specific format:

File Extension	Native Application	Comment
DOC, RTF	WinWord	Launch WinWord once and print a document via "Imager SDK DocCnvt Driver" once (Thus set it as the default driver).
HTM, HTML	Internet Explorer WinWord Netscape	You can also use WinWord or Netscape to print HTML files. If you prefer IE, you need to launch IE once and print a document via "Imager SDK DocCnvt Driver" once (Thus set it as the default driver).
XLS,	Excel Excel Viewer 97/2000	Excel Viewer is recommended
CSV	Excel	
PPT, PPS	PowerPoint	"Printto" → "D:\Program Files\Microsoft Office\Office10\POWERPNT.EXE" /p "%1"
PDF	Acrobat Reader	Need to run Acrobat Reader once
WRI,TXT,LOG	WordPad	
BMP,PCX,JPG, GIF,TIF,DCX	Kodak Imaging IrfanView	If you prefer IrfanView, please add "Printto" to all extensions with "%1" /print and change IrfanView default printer name to Imager SDK DocCnvt Driver at Properties Extra 2.
DFT	Solid Edge Smart Viewer	
DWG,DXF,DWF	Volo View Express 2	
PS	GSView	Need to run GSView once

DCT Engine API

You can call the DCT Engine in two ways:

1. Shell execute PDFilCnv.exe from your VC/VB program;
2. Call the PrintDocEx() function in PrintDoc from your VC/VB program

Metasoft provides various sample source for you to integrate DCT into your own application.

Note you can enable the debugging log by running PDFilCnv.exe with /l switch. DCT Daemon's log can be enabled on its main dialog. Log files are written to the following files:

```

\log\printdoc.log - status from PrintDoc.dll
\log\dialog.log - status from the Control Dialog PDDlg.exe
\log\pdfilcnv.log - status from PDFilCnv.exe
\log\wpdaemon.log - status from WPDaemon.exe

```

Below is the command line options for PDFilCnv.exe:

Command line options for PDFilCnv.exe		
Syntax	PDFilCnv "source file" [options]	
source file	Source document file name. If name has spaces enclose in quotes.	
Options	/cn	Command, [default is 2] n=0 -use printer default settings n=1 -use dialog settings n=2 -use command line parameter settings
	/d /dr /d:<server> /dr:<server>	Communicate request to Conversion Daemon If server name specified use the Conversion Daemon on server. If 'r' specified look to local registry for available Conversion Daemon. If r:server specified look to registry on <server> for available Conversion Daemon. <server> can be either IP address or computer name
	/e	Dump a result file (pdfilcnv.out, text file with either error code or number of pages printed) after conversion.
	/l	Log debugging info to file.
	/nt	t [optional] is pipe name used to communicate with driver
	/on	n [optional] specifies timeout in seconds. Will wait seconds for the print job to complete.
	/p<printer>	Specify the name of the printer to print to. Use only without the /d option. If name has spaces enclose in quotes.
	/q	Produce no printing output (quiet mode)
	/tn	Output file type: PS_TIF_GAMA 0 // reverse PS_TIF_NORMAL 1 // normal PS_DCX 2 // Intel DCX PS_PCX 3 // Paintbrush PCX PS_MR_REVERSE 4 // Group 3 2D (reverse) PS_MR_NORMAL 5 // Group 3 2D (normal) PS_MMR_REVERSE 6 // Group 4 (reverse) PS_MMR_NORMAL 7 // Group 4 (normal) PS_PACKBITS 8 // Normal Tif Pack Bits PS_PDF 1000 // Non Searchable PDF

	/wn	Communications timeout for the Daemon. If /d:server specified, wait n seconds for Conversion Daemon to become available.
	/mn	Page format [default is 1] n=0 specifies single page per output file n=1 multiple pages per file n=2 single page per file with new file names (xxxx001.tif, xxxx002.tif...)
	/fs	S is output filename. If name has spaces enclose in quotes.
	/?	Help screen
Returns:		<ul style="list-style-type: none"> > 0 Number of pages printed 0 Timeout, please check file association settings -1 Problem parsing command line -2 Timeout connecting to Daemon -3 Keyboard hit, conversion aborted -4 Cannot access the output folder or file -5 Daemon not running

Below is the API provided by PrintDoc.DLL:

PrintDocEx () - Convert a file using the Imager PCD	
Description	The routine looks up the file type in the registry, then 'spawns' the native application with the 'PrintTo' DDE string. Logging information is written to \log\printdoc.log on the default Windows drive.
Syntax	<pre>int __stdcall PrintDocEx (LPSTR pszPrinterName, // name of printer or "" for default LPSTR pszAnsiFileName, // input file name LPSTR pszOutFileName, // output file name int fileType, // output file format int multiplePages, // if 1, multiple pages per file // if 0, single page per file int command, // 0 -use printer default settings // 1 -use dialog settings // 2 -use command line settings, // default is 2. char *pipeName, // pipe name of daemon, default is: // "\\.\pipe\PrintDocDaemonPipe" int timeout, // # of seconds to wait for printing int mfx_enabled, // 1: display conversion dialog int log_enabled // 1: write out to \log\printdoc.log);</pre>
File Types	<pre>PS_TIF_GAMA 0 // reverse PS_TIF_NORMAL 1 // normal PS_DCX 2 // Intel DCX PS_PCX 3 // Paintbrush PCX PS_MR_REVERSE 4 // Group 3 2D Modified Read (reverse) PS_MR_NORMAL 5 // Group 3 2D Modified Read (normal) PS_MMR_REVERSE 6 // Group 4 Modified read (reverse)</pre>

	PS_MMR_NORMAL 7 // Group 4 Modified read (normal) PS_PACKBITS 8 // Normal Tif Pack Bits
Returns	>0 number of pages printed 0 Timeout, check the file association settings. -1 Problem parsing command line. -2 Timeout connecting to Daemon. -3 Keyboard hit, conversion aborted. -4 Cannot access the output folder or file. -5 Daemon not responding -6 Registry Daemon not running
Pipe Names	Pipe names are opened by the WpDaemon and can be addressed as follows: \\servername\pipe\PrintDocDaemonPipe where 'servername' is the name of the server the daemon is running on. If the daemon is running on the local server, set pipename to: \\.pipe\PrintDocDaemonPipe NULL do not use daemon "" do not use daemon "\\.pipe\PrintDocDaemonPipe" use default pipe name for daemon on current machine "\\ServerName\pipe\WordConversion" use custom pipe name "HKEY:HKEY_LOCAL_MACHINE\...\\" use custom registry key to find daemon
Sample	<pre>int ret = (*ptrPrintDoc)("Imager SDK DocCnvt Driver", "Source.doc", "C:\\Output\\Image.tif", 1, 1, "", 2, 60, 1, 1);</pre>

If conversion fails, two possible failure modes can occur:

1. The printing application stops before printing and is looking for user intervention (macro, password, etc). This is not a problem, as the print operation is not affected. All that will happen is the printing application doesn't return, and PDFILCNV times out. If you call DCT from your server application, you may want to terminate the native application after the timeout.
2. The native application hangs, or does not complete. Two likely causes are the printing process is taking way longer than the allocated timeout (you need to adjust the timeout), or the print process has stopped (likely caused by a Control Dialog problem). All following print jobs will be queued, and nothing will print. Remedy is to delete the jobs, stop, and then re-start the print spooler.

The key to the conversion process is to process only one job at a time. If more than one job is queued, then PDFilCnv.exe will continue to time out, and nothing will be printed.