

# PDF-to-TIFF (CP2T.DLL)

**DLL Version**

C, C++  
.NET (C#, VB.NET, ASP.NET)

Version 5.0

## User Guide

1<sup>st</sup>. May 2009

Please print this User Guide for easy reference

## **Table of Contents**

Introduction .....	3
Installation .....	3
Demo Version .....	3
Requirements.....	3
How To Use .....	4
Functions .....	4
SetXY .....	4
SetBPP.....	5
SetCompress.....	5
SetSize.....	6
Page Range.....	6
Split .....	6
Convert .....	7
DeletePDF .....	7
Placement of Program Files (Important) .....	7
Fonts and Font Substitution.....	8
Password-protected PDF Files .....	8
Free-Type Engine .....	9
Technical Support .....	9
Other Software .....	9
Licensing .....	10
Copyright.....	10
License Agreement.....	10

## Introduction

The CP2T DLL converts PDF files to TIFF files. The DLL processes one file per call; a single-page PDF file is converted to a single-page TIFF file and a multi-page PDF file is converted to an equivalent multi-page TIFF file. The program supports both text-based and image-based PDF files.

This version supports C, C++, C#, VB.NET, and ASP.NET. A separate version is available for VB 6.0.

Please visit <http://www.hsm.ch/> and <http://www.expert-tools.com/> for the VB DLL, the user-interactive (GUI), and the command-line versions of the program.

## Installation

The program files are supplied in a zip file. Please use the following procedure to install the program:

1. Create a new folder for the program. We recommend that you call the new folder C:\PDF2TIFF but, you may give the folder any name of your choice.
2. Unzip the program zip file into that newly created folder. The unzipping process will create the correct directory structure with a sub-directory called PSFONTS. Do not change the directory structure.

## Demo Version

The Demo Version will cripple the TIFF-output file with a diagonal line from the upper-right to the lower-left corner of the image. The licensed version will not draw crippling lines. The download pack includes sample PDF files that you can use for testing.

An actual demo version of the DLL can be found at <http://www.expert-tools.com/>

## Requirements

- Windows 98/Me/NT4/2000/XP/2003/VISTA
- 256MB memory
- FreeType Engine (supplied and installed with CP2T), see Free-Type Engine section.

## How to Use

The CP2T DLL works like any Windows DLL. The eight functions and the required arguments are described below.

Sample codes using the DLL can be found at <http://www.expert-tools.com/>

## Functions

There are eight (8) functions:

1. **SetXY** (to set the Dots-per-Inch; resolution)
2. **SetBPP** (to set the Bits-per-Pixel; color)
3. **SetCompress** (to set the TIFF compression)
4. **SetSize** (to set width and height in pixels)
5. **PageRange** (to set the page range)
6. **Split** (to split the output into serialized single-page TIFF files)
7. **Convert** (to execute the conversion)
8. **DeletePDF** (to delete the source-PDF file)

The first six functions are called to specify the properties and attributes of the output TIFF file. These functions are optional; if they are not called, the system will use the default settings (300x300 resolution, monochrome TIFF Group 4 output files, and converts all pages at the original pixel size into multi-page file). The six functions, if used, must be called before the main Convert function.

The optional DeletePDF function will delete the source file after conversion. This function must be called after the Convert function. **Please use this function with caution as the PDF file may be deleted permanently.**

### SetXY

This function takes 2 integer values in the range of 1 – 1200, separated by a comma. The first value is for the horizontal resolution; the second value is for the vertical resolution.

The function will return 0 (zero) upon successful completion or 3 if one of the chosen values is out of range.

This function is optional.

If called, it MUST be called before the Convert function. If called, the converted TIFF will have the specified DPI values; if not called, the TIFF file will have a default value of 300dpi. The function must include both values.

Example: SetDPI (204,196)

### **SetBPP**

This function takes 1 integer value in the range of 1-3, where

- 1 = 1BPP (monochrome)
- 2 = 8BPP (greyscale)
- 3 = 24BPP (24-bit color)

The function will return 0 (zero) upon successful completion or 4 if the specified value is invalid.

This function is optional.

If called, it MUST be called before the Convert function.

If called, the converted TIFF will have the specified BPP value; if not called, the TIFF file will have a default value of 1BPP.

Example: SetBPP (3)

### **SetCompress**

This function takes 1 integer value in the range of 1-5, where

- 1 = NONE (uncompressed)
- 2 = RLE
- 3 = FAX3 (TIFF Group 3)
- 4 = G4 (TIFF Group 4)
- 5 = LZW

The function will return 0 (zero) upon successful completion or 5 if the specified value is invalid.

This function is optional.

If called, it MUST be called before the Convert function.

If called, the converted TIFF will have the specified Compress value; if not called, the TIFF file will have a default value of G4.

The SetBPP (Bits-per-Pixel) takes precedence over the compression option. If there is a conflict, the Bits-per-Pixel specification prevails. For example, if you specify color output and TIFF Group 4 Compression, the output file will be saved in LZW compression because TIFF Group 4 compression can only be monochrome.

For other compressions and for output in other graphics formats, please contact [support@hsm.ch](mailto:support@hsm.ch).

Example: SetCompress (5)

## ***SetSize***

This function takes 2 long values representing the width and height (in pixels) of the TIFF-output file.

The function will return 0 upon successful completion or a non-zero value to signify an error.

This function is optional.

If called, it **MUST** be called before the Convert function.

If called, the converted TIFF will have the specified width and height; if not called, the TIFF file will have the pixel size of the original file. The function must include both values.

Example: SetSize (1728,2044)

## ***Page Range***

This function takes 2 integer values representing the beginning (first page) and the end (last page) of the page range

The function returns 0 (zero) upon successful completion or a non-zero value to signify an error.

This function is optional.

If called, it **MUST** be called before the Convert function.

If called, the converted TIFF will include only the pages in the specified range. If the first page is incorrect, the function will default to page 1; if the last page is incorrect, the function will default to the correct last page.

Examples: PageRange (3,6)

PageRange (1,1)

## ***Split***

Takes 1 integer value (0 or 1), where

1 = split the output into serialized single-page TIFF files

0 = same effect as if the function was not even called at all

The function returns 0 (zero) upon successful completion or a non-zero value to signify an error.

This function is optional.

If called, it **MUST** be called before the Convert function.

If called, the output of a multi-page PDF file will be split into serialized single-page TIFF files. The output file names will be suffixed with serial numbers such as somefile\_1, somefile\_2, etc.

Example: Split (1)

### **Convert**

This function takes 2 strings (in & out) as parameter, where  
in = file and path of the PDF to be converted  
out = file and path of the TIFF-output file  
The two strings have to be separated by a comma

The function returns 0 (zero) upon successful completion, a 1 if the PDF file cannot be loaded, or a 2 if the TIFF file cannot be created.

When out = "NULL", then the PDF file will be converted to a tiff file in the same path with the same name but a .tif extension.

If the file name includes spaces, the file name must be encapsulated in quotes.

Please read the Password section below for handling **password-protected PDF files**.

Examples: Convert C:\somedir\somefile.pdf,C:\otherdir\somefile.tif  
Convert "C:\somedir\my file.pdf","NULL"  
Convert "C:\somedir\my file.pdf","C:\otherdir\my file.tif"  
(File names with spaces must be encapsulated in quotes)

### **DeletePDF**

This function takes a single string parameter; the path and name of the PDF file which will be deleted. **Use this function with caution as the PDF file will be deleted permanently.** Therefore, it is highly recommended to make a backup of the file before conversion. *HSM Informatik AG declines any responsibility if a file is deleted due to inattentive use of this function.*

Example: DeletePDF(c:\somedir\somefile.pdf)  
(File names with spaces must be encapsulated in quotes)

## **Placement of Program Files (Important)**

The unzipping of the program installation places the files into a fixed directory structure. The CP2T.DLL file, the HSM\_PDF\_RASTER.DLL and the PSFONTS subdirectory are placed in the folder that you specify when you unzip the program installation file. For the purpose of this user guide, we use C:\PDF2TIFF as the name of the main folder, but you can choose any folder name. While you may

change the name of the main folder, it is IMPORTANT that the main folder always retains the PSFONTS subdirectory with its files. The DLL will not run if either of the files or the PSFONTS subdirectory is missing.

When you write your application, it is important that the compiled executable of your application be placed in the same folder as CP2T.DLL and HSM\_PDF\_RASTER.DLL. You must therefore move CP2T.DLL, HSM\_PDF\_RASTER.DL, and the PSFONTS subdirectory to the folder where you placed your application's executable.

While testing your application, it is important that CP2T.DLL, HSM\_PDF\_RASTER.DLL, and the PSFONTS subdirectory are placed in the correct directory. In VB.NET, for example, they would have to be placed in the Bin folder. It may be simpler, though, not to run the application in development mode but, instead, to compile the application and place the executable directly in the PDF2TIFF folder.

## Fonts and Font Substitution

CP2T converts the most common fonts to the correct style and appearance. Some rare fonts may not be translated to the exact font's appearance and style. If a font cannot be translated, CP2T will automatically apply the available font most similar to the original font.

## Password-protected PDF Files

You will accept responsibility for opening password-protected PDF files.

### PDF Passwords Explained

There are two types of passwords for PDF files:

- a) Owner Passwords. The password can be unlocked only by the owner (creator) of the PDF File. Depending on the scope of the password, it disables the edit, print, and/or search functions. With an Owner Password, the PDF file can be opened and viewed but typically cannot be edited, printed or searched. There is no method for the user to unlock the password.
- b) User Password. The user must enter the password in order to open the PDF file. Once the file is open, the user can view, print, edit and search the file (edit and search only if text based). If the PDF file is also protected by an Owner password, then the user cannot edit, print and/or search the PDF file, depending on the scope of the Owner password.

The PDF2TIFF programs provide a function to enter the User password. Please contact Support for instructions. Owner passwords cannot be unlocked.

## **How to process a password-protected PDF file** (only for PDF files with User Passwords)

VB.NET

### ***SetPar (char \*)***

This function takes 1 string (char \*).

The function will return 0 (zero) upon successful completion or non zero upon failure.

## **Free-Type Engine**

The CP2T includes the Free-Type Engine (copyright 1996-2002, The Free Type Project). It is here used under the terms of the Free Type Project License. The Base 14 Type 1 fonts are part of the font set contributed to the Ghostscript project by URW++ Design and Development Incorporated of Hamburg, Germany (<http://www.urwpp.de/>). They have been released under the GNU General Public License (GPL) – see "COPYING" file in the lib/t1fonts directory. Adobe Systems holds the copyrights (1985-2001) for the PDF data structures, operators, and specifications.

## **Technical Support**

For technical support please contact [support@hsm.ch](mailto:support@hsm.ch)

## **Other Software**

Please visit <http://www.expert-tools.com> for other useful software, such as:

- Informatik Image Driver is a high-performance driver that converts any printable Windows document to TIFF, PDF (and other formats).
- TIFF2PDF utilities convert TIFF files to PDF.
- Textlog extracts text from text-based Windows documents.
- Further conversion utilities for PNG, PDF, TIFF, and other graphics formats.

# Licensing

For licensing fees please visit [www.expert-tools.com](http://www.expert-tools.com). Please note that the demo version outputs TIFF files crippled with a diagonal line drawn across the image. The licensed version does not draw such a line.

## **Copyright**

Copyright 1996-2009 HSM Informatik AG. All Rights Reserved

## **License Agreement**

HSM Informatik AG and Informatik Inc ("LICENSOR")

**IMPORTANT - READ CAREFULLY:** This License Agreement is a legal agreement between you and the supplier of the PDF2TIFF software identified above which may include associated software components, media, printed materials, and "online" or electronic documentation ("SOFTWARE "). By installing, copying, or using the SOFTWARE, you agree to be bound by the terms of this License Agreement. If you do not accept the terms of this License Agreement, do not install or use the SOFTWARE.

The SOFTWARE is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE is licensed, not sold.

1. GRANT OF LICENCE: The SOFTWARE is licensed as follows:

a) Developers License (DLL version):

Installation and Use: LICENSOR grants you the right to install and use a single copy of the SOFTWARE on your computer for the development of one (1) application ("APPLICATION"). The SOFTWARE must be a minor part of your APPLICATION. The application must be an end-user APPLICATION, it must NOT be a wrapper program, and must not be a toolkit or a component that can be used as development tools or utilities in other applications. The software must not be used in the form of a static library. The APPLICATION must NOT be used on a web server, unless specifically approved in advance by the LICENSOR in writing. If the SOFTWARE is used in more than one application, a license is required for each such application.

b) Single-user or multi-user version (command-line and user-interactive versions only): LICENSOR grants you the right to install and use a single copy of the SOFTWARE on one computer in the case of a single-user license, or on the specific number of computers licensed in a multi-user license. Installation on a server requires the server version. The SOFTWARE must not be used for development of an application. The SOFTWARE must only be used by the licensee; it must not be distributed to other companies or persons, without a prior written agreement by the LICENSOR.

**Backup Copies:** You may also make copies of the SOFTWARE as may be necessary for backup and archival purposes.

## 2. DESCRIPTION OF OTHER RIGHTS AND LIMITATIONS:

Maintenance of Copyright Notices: You must not remove or alter any copyright notices on all copies of the SOFTWARE.

Distribution: In the case of a Developers License, you may freely distribute the APPLICATION that uses the SOFTWARE, subject to the limitations outlined in this LICENSE AGREEMENT. You may not distribute copies of the documentation relating to the use of the SOFTWARE. Your APPLICATION must include the SOFTWARES' copyright statement "Portions of this application were created using software by HSM Informatik AG. All Rights Reserved.

- Prohibition on Reverse Engineering, De-compilation, and Disassembly: You may not reverse engineer, de-compile, or disassemble the SOFTWARE.
- Rental: You may not rent, lease, or lend the SOFTWARE.
- Transfer: You may not transfer all of your rights under this License Agreement without the prior agreement of the LICENSOR in writing.
- Compliance with Applicable Laws: You must comply with all applicable laws regarding use of the SOFTWARE.

3. TERMINATION: Without prejudice to any other rights, LICENSOR may terminate this License Agreement if you fail to comply with the terms and conditions of this License Agreement. In such event, you must destroy all copies of the SOFTWARE.

4. COPYRIGHT. All titles, including but not limited to copyrights, in and to the SOFTWARE and any copies thereof are owned by HSM Informatik AG. All rights, not expressly granted, are reserved by copyright holder.

(c) HSM Informatik AG, 1996-2009.

5. NO WARRANTIES: LICENSOR or its agents, suppliers, and distributors expressly disclaim any warranty for the SOFTWARE. THE SOFTWARE AND ANY RELATED DOCUMENTATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OR MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. THE ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE REMAINS WITH YOU.

6. LIMITATION OF LIABILITY: To the maximum extent permitted by applicable law, in no event shall LICENSOR or its suppliers, agents, resellers, be liable for any special, incidental, indirect, or consequential damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use the SOFTWARE or the provision of or failure to provide Support Services, even if LICENSOR or its suppliers, agents, resellers have been advised of the possibility of such damages.

7. Licensee agrees that the trial version was fully tested prior to purchase of the SOFTWARE and it is clearly understood that the license fee is NOT REFUNDABLE.

8. The place of jurisdiction is Lucerne, Switzerland.

