
3-Heights™ PDF to Image Converter Shell

Version 4.5

User Manual

Contact: pdfsupport@pdf-tools.com

Owner: **PDF Tools AG**
Kasernenstrasse 1
8184 Bachenbülach
Switzerland
<http://www.pdf-tools.com>

July 1, 2015

Table of Contents

1	Introduction	5
1.1	Description	5
1.2	Functions.....	5
	Features	5
	Formats	6
1.3	Operating Systems.....	6
2	Installation	7
2.1	Installing the 3-Heights™ PDF to Image Converter.....	7
	How to Set the Environment Variable "Path"	7
3	License Management	8
3.1	Graphical License Manager Tool	8
	List all installed license keys	8
	Add and delete license keys	8
	Display the properties of a license	9
	Select between different license keys for a single product	9
3.2	Command Line License Manager Tool	9
	List all installed license keys	9
	Add and delete license keys	9
	Select between different license keys for a single product	9
3.3	License Key Storage.....	9
	Windows.....	10
	Mac OS X.....	10
	Unix / Linux	10
4	User Guide	10
4.1	Getting Started.....	10
4.2	General Settings.....	11
4.3	Specify Image Type, File Name and Output Folder	11
4.4	Single Page or Multi Page Images	12
4.5	The Use of Wildcards (*.pdf)	12
4.6	How to Reduce the File Size	13
	Dimensions	13
	Resolution.....	14
	Bits per Pixel	14
	Format / Compression Type	14
	Image Content, Dithering.....	15
4.7	Options for Best Results on a Printer	16
4.8	How to Select the Color Profiles	16
4.9	Dithering	16
	Remarks	17
	Color Images.....	17
	Bi-tonal Images	18
	Guidelines.....	19
5	Reference Manual	21
5.1	Supported Codecs.....	21
5.2	Rendering Options	22

July 1, 2015

-1	Create one Image File per PDF Page	22
-b	Set the Bits per Pixel.....	22
-cms	Set the color management engine	22
-cn	Center Mode	22
-cs	Set the Color Space	23
-d	Set the Resolution	23
-dx	Set the X-Resolution	23
-dy	Set the Y-Resolution	23
-f	Fit Page Mode	23
-fax	Convert to Class F	23
-fo	Bit Fill Order	24
-fs	Filter Size	24
-g	Gray Color Space.....	24
-h	Dithering Mode.....	24
-i	Indexed Color Space	25
-lf	Set Line Width Multiplication Factor.....	25
-lw	Set Minimum Line Width.....	25
-m	Rendering Mode	25
-o1	Use Type1 Fonts.....	25
-oe	Do not Use Embedded Fonts	25
-ofp	Use Pre-Installed fonts	25
-op	Disable Pattern.....	26
-oq	Disable High Quality Rendering	26
-ot	Use TrueType Fonts	26
-p	Read an Encrypted PDF File	26
-pg	Set Page Range.....	26
-pgs	Set of Pages	26
-q	Set Image Quality of Lossy Compression	26
-r	Rotate Pages According to Page Rotate Attribute	27
-rl	Rotate Pages to Landscape	27
-rp	Rotate Pages to Portrait.....	27
-s	Set Width and Height of Image in Points.....	27
-sp	Set Width and Height of Image in Pixel.....	27
-sa	Set Width and Height in Pixel and Preserve Ratio	28
-t	Set Threshold When Dithering is Disabled	28
5.3	Compression Options	28
-g3	Compression CCITT Group 3	28
-g4	Compression CCITT Group 4	28
-l	Compression LZW.....	28
-j	Compression JPEG	28
-j6	Compression JPEG (6).....	29
-z	Compression ZIP	29
5.4	pdf2pdfimg Specific Options	29
-nl	Disable the copying of links.....	29
-no	Disable the copying of outlines	29
-nv	Disable the copying of viewer preferences	29
5.5	General Options.....	29
-v	Verbose Mode	29
-lk	Set License Key.....	30
5.6	Return Codes	30
6	Tips, Ticks and Troubleshooting.....	31

July 1, 2015

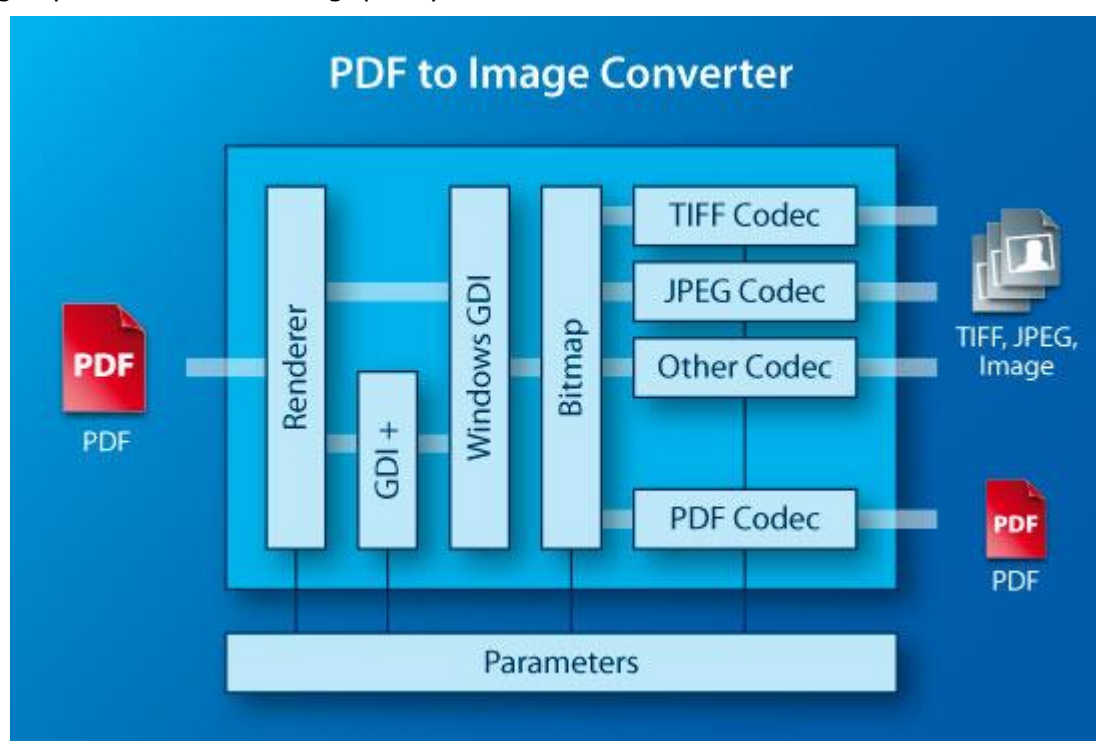
6.1	General	31
	Pages Are Rotated.....	31
	Generated Files Have a Large Size	31
	Images Are of too Low Quality	31
	Image Does Not Contain the Whole Content	31
	Colors Are Gone.....	31
6.2	Text	32
	Font Replacement Strategy	32
	Installed Font Collection.....	33
	Using the Font Mapping File <i>fonts.ini</i>	33
	Other Ways to Deal with Text Issues	34
6.3	Transparency	34

1 Introduction

1.1 Description

The 3-Heights™ PDF to Image Converter converts PDF documents into single page or multi-page raster images such as TIFF or JPEG. It can also convert PDF files into rasterized PDF.

Its areas of use include the web, TIFF-based DMS solutions, archive and workflow systems and the protection of PDF documents. The Converter is characterized by its high speed and outstanding quality.



1.2 Functions

The 3-Heights™ PDF to Image Converter merges pages from different input files to form one or more files. Color space and image size are defined automatically during the process. The Converter supports scaled and non-scaled conversions and a variety of image formats such as PNG, TIFF, JBIG2 or JPEG2000.

Features

- Create single page and multi-page image files and rasterized PDF documents
- Convert individual pages

July 1, 2015

- Convert a PDF file into a PDF image file so that the new file contains the entire text and all the vector graphics as raster images (to prevent text extraction, for instance).
- Convert PDF files to CCITT fax files
- Set page mode
- Define page dimensions in points or pixels
- Set rotation (identical to the incoming PDF document or portrait or landscape)
- Resolution in DPI including X and Y direction
- Dithering (Floyd Steinberg, Halftone Block, Halftone Continuous)
- Set image filters
- Set color depth
- Set color space
- Set TIFF file compression
- Set the quality of lossy image compression
- Set bit filling order for fax files
- Define minimum line width

Formats

Input Formats:

- PDF 1.x (e.g. PDF 1.4, PDF 1.5, etc.)

Target Formats:

- TIFF (Tagged Image File Format)
- JPEG (Joint Photographic Expert Group)
- PNG (Portable Network Graphics)
- GIF (Graphics Interchange Format)
- BMP (Window Bitmap)
- EPS (Encapsulated PostScript)
- JBIG2 (Joint Bi-level Image Experts Group)
- JPEG2000
- Extended JPEG2000
- PBM (Portable Bitmap File Format)

1.3 Operating Systems

Windows XP, Vista, 7, 8, 8.1 - 32 and 64 bit

Windows Server 2003, 2008, 2008 R2, 2012, 2012 R2 - 32 and 64 bit

2 Installation

2.1 Installing the 3-Heights™ PDF to Image Converter

The retail version of the 3-Heights™ PDF to Image Converter Service comes as a ZIP archive containing various files including runtime binary executable code, documentation and license terms.

1. Download the ZIP archive of the product from your download account at www.pdf-tools.com.
2. Open the ZIP archive.
3. Check the appropriate option to preserve file paths (folder names) and unzip the archive to a local folder (e.g. *C:\program files\pdf-tools*).
4. The unzip process now creates the following subdirectories:
 - *Bin*: Contains the runtime executable binary code
 - *Bin\Fonts*: Contains the font ZapfDingbats and the font mapping file. Copy and thereby install the two fonts to the OS fonts directory (%SystemRoot%\Fonts, e.g. *C:\Windows\Fonts*), or deploy the directory together with the product's executable.
 - *Doc*: Contains documentation files
5. To start the 3-Heights™ PDF to Image Converter from a shell, the directory needs to be included in the "Path" environment variable.

How to Set the Environment Variable "Path"

To set the environment variable "Path" on Windows 2000, go to **Start -> Settings -> Control Panel -> System -> Advanced -> Environment Variables**

Windows XP, go to **Start -> Control Panel** (classic view) -> **System -> Advanced -> Environment Variables**.

Select "Path" and **Edit**, then add the directory where *pdf2img.exe* is located to the "Path". If the environment variable "Path" does not exist, create it.

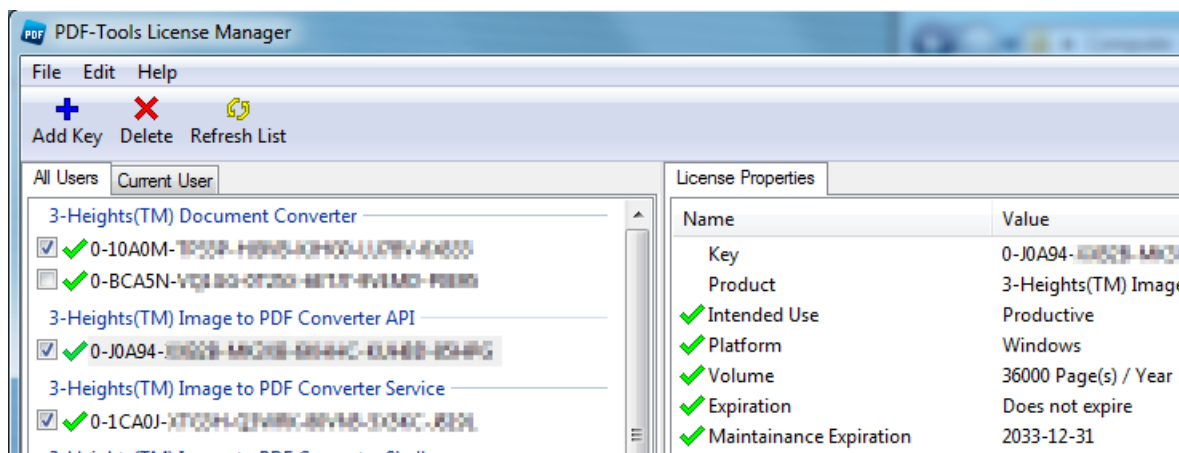
3 License Management

There are three possibilities to pass the license key to the application:

1. The license key is installed using the GUI tool (Graphical user interface). This is the easiest way if the licenses are managed manually. It is only available on Windows.
2. The license key is installed using the shell tool. This is the preferred solution for all non-Windows systems and for automated license management.
3. The license key is passed to the application at runtime via the command line switch `-lk` property. This is the preferred solution for OEM scenarios.

3.1 Graphical License Manager Tool

The GUI tool *LicenseManager.exe* is located in the *bin* directory of the product kit.



List all installed license keys

The license manager always shows a list of all installed license keys on the left pane of the window. This includes licenses of other PDF Tools products.

The user can choose between:

- Licenses available for all users. Administrator rights are needed for modifications.
- Licenses available for the current user only.

Add and delete license keys

License keys can be added or deleted with the "Add Key" and "Delete" buttons in the toolbar.

- The "Add key" button installs the license key into the currently selected list.
- The "Delete" button deletes the currently selected license keys.

July 1, 2015

Display the properties of a license

If a license is selected in the license list, its properties are displayed in the right pane of the window.

Select between different license keys for a single product

More than one license key can be installed for a specific product. The checkbox on the left side in the license list marks the currently active license key.

3.2 Command Line License Manager Tool

The command line license manager tool *licmgr* is available in the *bin* directory for all platforms except Windows.

A complete description of all commands and options can be obtained by running the program without parameters:

```
licmgr
```

List all installed license keys

```
licmgr list
```

The currently active license for a specific product is marked with a star '*' on the left side.

Add and delete license keys

Install new license key

```
licmgr store X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX
```

Delete old license key

```
licmgr delete X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX
```

Both commands have the optional argument *-s* that defines the scope of the action:

- *g*: For all users
- *u*: Current user

Select between different license keys for a single product

```
licmgr select X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX
```

3.3 License Key Storage

Depending on the platform the license management system uses different stores for the license keys.

July 1, 2015

Windows

The license keys are stored in the registry:

- HKLM\Software\PDF Tools AG (for all users)
- HKCU\Software\PDF Tools AG (for the current user)

Mac OS X

The license keys are stored in the file system:

- /Library/Application Support/PDF Tools AG (for all users)
- ~/Library/Application Support/PDF Tools AG (for the current user)

Unix / Linux

The license keys are stored in the file system:

- /etc/opt/pdf-tools (for all users)
- ~/.pdf-tools (for the current user)

Note: The user, group and permissions of those directories are set explicitly by the license manager tool.

It may be necessary to change permissions to make the licenses readable for all users. Example:

```
chmod -R go+rx /etc/opt/pdf-tools
```

4 User Guide

The 3-Heights™ PDF to Image Converter Shell provides two executables: pdf2img and pdf2pdfimg.

pdf2img is used to convert PDF files to raster images, such as TIFF, JPEG, etc. The functionality of this executable is documented in this manual.

pdf2pdfimg is used to render PDF files to images and re-embed these images back into a new PDF document. As a result the new PDF document contains exactly one image per page and no further content objects. Alternatively the text can be retained (option -oa). That means only images and vector graphics are converted to an image and the text is added on top of the image. pdf2pdfimg is not further described in this documentation. Most of its switches are equivalent to those of pdf2img.

All switches are described in the usage of the tools. (Type pdf2img or pdf2pdfimg with no arguments to list the usage).

4.1 Getting Started

The simplest command requires one parameter: The name of the PDF input file. When no output file name is specified, the output file will be named as the input file, and the image type TIFF is selected.

July 1, 2015

Example: The following command

```
pdf2img input.pdf
```

creates an uncompressed TIFF file. It inherits the name of the input file and is named:

```
input.tif
```

If the PDF file has more than one page, the generated TIFF will be a multi-page TIFF. Keep in mind that TIFF supports multi-paging whereas most other image formats are single-page formats.

4.2 General Settings

```
pdf2img -lk X-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX-XXXXX input.pdf output.tif
```

Pass a license key to the application at runtime instead of installing it on the system.

4.3 Specify Image Type, File Name and Output Folder

The name and type of the output file can be specified using a second parameter. Here is a list of supported extensions and the corresponding file type:

Table: File Format	
Extension	File Format
.tif, .tiff	Tagged Image File Format
.jpg, .jpe, .jpeg	Joint Photographic Expert Group
.png	Portable Network Graphics
.gif	Graphics Interchange Format
.bmp	Window Bitmap
.j2b	Joint Bi-level Image Experts Group
.jp2	JPEG2000
.jpx	Extended JPEG2000
.pbm, .pgm, .pnm, .ppm	Portable Bitmap File Format
.eps	Encapsulated PostScript (Output only)

Example: The following command creates a JPEG image in the current working folder:

```
pdf2img input.pdf output.jpg
```

Example: The output directory can simply be added in front of the output file name:

```
pdf2img input.pdf myfolder\output.jpg
```

or absolute:

```
pdf2img input.pdf C:\myfolder\output.jpg
```

Example: Quotes must be used for paths or file names that contain blanks:

```
pdf2img "My File.pdf" "My Documents\output.jpg"
```

4.4 Single Page or Multi Page Images

By default, the PDF to Image Converter generates multi-page images if the TIFF format is selected and the input PDF has more than one pages. To create one TIFF file per PDF page, simply use the switch -1. This option should also be used for all other formats, which do not support multi-paging, if the input PDF document has more than one page.

When doing so, the output files can be named with wildcards.

Example: Create images consisting of 1 page per image, add the page number to the file name as four digit number.

```
pdf2img -1 input.pdf output%04d.png
```

Image files created this way are named:

```
output0001.png, output0002.png, output0003.png, etc.
```

4.5 The Use of Wildcards (*.pdf)

Wildcards can also be used to process multiple input files.

Example: If a directory contains the following PDF files:

```
A01.pdf, A02.pdf, A03.pdf, B01.pdf
```

Then the following command processes all PDF files starting with "A".

```
pdf2img A*.pdf
```

Or this command processes the files that end with "01.pdf"

```
pdf2img *01.pdf
```

Note that the file extension of the input file must always be ".pdf".

When using wildcards, it is helpful to enable the verbose mode using the switch -v.

Example: Using wildcards in verbose mode:

```
pdf2img -v *01.pdf
```

And the generated output message looks like this:

```
Converting file A01.pdf
```

```
Rendering page 1 ...
```

```
Rendering page 2 ...
```

```
...
```

```
Converting file B01.pdf
```

```
Rendering page 1 ...
```

July 1, 2015

...

Done.

Wildcards return a list of existing documents, therefore they cannot be used for naming not yet existing documents. If you would like to convert all PDF documents in a directory to individual multi-page TIFF documents, it is required to use a variable to name the output files.

Example: Use the FOR command of the CMD shell, to convert all *.pdf images to TIFFs with the same name and the extension tif, in the same folder:

```
for %i in (*.pdf) do pdf2img -v %i %~ni.tif
```

Example: Of course, one can adjust the paths, or use a different output name:

```
for %i in (C:\PDF\*.pdf) do pdf2img %i C:\Img\%~ni.tif
```

For additional help to the FOR command, type the command:

```
for /?
```

Note that in batch files, variables have two leading %% instead of just one %.

4.6 How to Reduce the File Size

There are different ways to reduce the file size of an image. One needs to be aware that from a certain point on, a smaller file size results in a poorer visual quality.

The main factors on which the file size of an image depend are:

- Dimensions in pixel (width and height)
- Bits per pixel
- Compression Type
- The content of the image (influenced by dithering)

Dimensions

Reducing the dimensions and therefore the amount of the total pixels, reduces also the file size. Obviously a 1024x768 pixel image has a larger file size than an equivalent 600x480 image.

Example: Set the size of the image to 600 by 480 pixel, scale the page to fit the given dimensions:

```
-sp 600 480 -f
```

Example: Set the size of the image to 600 by 480 points, scale the page to fit the given dimensions:

```
-s 600 480 -f
```

If the dimensions are set in points, the dimensions in pixel are calculated depending on the resolution.

July 1, 2015

Resolution

The resolution in dots per inch (dpi) lets you specify how detailed the image is. The default value is 150 dpi, which generates an image that looks sharp when not zoomed into. A larger value generates a more detailed image, but also will increase the file size, because it requires more pixels. On the other hand, a lower resolution generates a file with a smaller file size, but the image is also of lower visual quality.

Example: Set the resolution to a value of 75 dpi in order to reduce the file size to about one quarter.

```
-d 75
```

Bits per Pixel

Using 1-bit (black/white) or 8-bit grey scale instead of 24-bit true color will reduce the file size. Keep in mind that not all formats support all color depths.

Example: The following command will create a bi-level image (black/white) TIFF image.

```
-b 1 -h 6 -oq
```

8-bit grey scale images are a third as large in size as 24-bit color images. With 1-bit images that use dithering, the size heavily depends on the content. It can be as small as 1% of the 8-bit image.

Format / Compression Type

The 3-Heights™ PDF to Image Converter supports various image formats. For most formats the compression is given. For example a PNG image is always Flate-compressed, a JPEG image is always JPEG-compressed. However for TIFF, the compression type is selectable.

Images formats that are supported by most Internet browsers are JPEG, GIF and PNG.

There are two fundamentally different types of compression: Lossless and lossy.

Lossless compression

The transformation from the original to the compressed state of the image does not change the content. Thus the transformation is reversible and the original image can be regained from the compression state.

Lossless compression is normally used for artificial images or scanned text.

It is applied to the following types of images: GIF, PNG, BMP, JPEG2000 if quality is set to 100, JBIG2 if quality is set to 100 and TIFF compressed with G3, G4, LZW or Flate.

Lossy compression

The compression algorithm alters the content of the image in a way that it compresses better. Thus a lossy compressed image cannot be reverted back to its original state. It also means multiple applications of lossy compression to the same image alter the image every time and thereby reduce the quality every time. How much the image may be altered to improve the compression rate is

July 1, 2015

controlled by a quality index ranging from 1 to 100 and normally defaulted at 75.

Lossy algorithms usually provide a better compression rate, at the cost of visual quality. Lossy compression is normally used for photographs.

It is applied to the following types of images: JPEG, JPEG2000 if quality is less than 100 and JBIG2 if quality is less than 100.

There are various compression types supported for the TIFF image format. These are:

CCITT Group 3, Group 3-2D

CCITT Group 3 is the predecessor to CCITT Group 4, it is a simpler algorithm that normally results in a lower compression ratio.

CCITT Group 4

CCITT Group 4 is the standard compression for bi-level TIFF images (i.e. facsimile).

LZW

LZW (Lempel-Ziv-Welch) compression is a lossless compression algorithm for images. Please consult the copyright laws of your country prior to using this compression algorithm.

JPEG

TIFF allows images to be compressed with JPEG, which is a lossy compression algorithm. JPEG provides a high compression ratio for 8 and 24 bit images. It is best suited for TIFFs containing photographs and little or no text.

ZIP (Flate)

ZIP is a lossless compression algorithm. It is useful for the compression of large images with no loss in quality.

Flate compression (also used by the ZIP format) and JPEG compression can be used for color or grey scale images. CCITT Group 3, 3-2D and 4 as well as Flate can be used for black and white images.

Example: Apply Flate compression to a TIFF image.

-z

Image Content, Dithering

The content of the image itself has a direct impact on how well it compresses. It seems quite obvious that a plain white image compresses much better than a page filling photograph.

Dithering is an algorithm that arranges the pixels of an image in a way that it creates a visual effect of colors that do not exist in the available colors of the image, such as different grays in a 1-bit black and white image. This complex arrangement of pixels

July 1, 2015

however does not compress well and increase the file size. Disabling dithering therefore reduces the file size. In the PDF to Image Converter, dithering is also implemented for color images.

Example: *Disable dithering, e.g. for a black and white scanned text.*

```
-h 0
```

See also chapter Dithering.

4.7 Options for Best Results on a Printer

Regular laser printers use a resolution of 600 or 1200 dpi. For best results on b/w printers, choose a resolution of the same value (600 dpi). In this case, it is also important to generate black and white images and thus set the bits per pixel to 1.

Example: *Create a bi-tonal, 1200 dpi, CCITT G4 compressed TIFF.*

```
pdf2img -b 1 -d 1200 -g4 input.pdf output.tif
```

A compressed A4 black and white image with a resolution of 1200dpi will be about 1MB in size. Note that using such a high resolution in combination with 8bit grey scale or 24bit color images will generate huge files (several hundreds of MBs uncompressed, and around 10 MB using JPEG compression).

DPI values larger than 2400 will take a lot of CPU power and memory, we recommend not using values above 2400 for A4 paper size PDF documents.

4.8 How to Select the Color Profiles

The 3-Heights™ PDF Rendering Engine works in the RGB color space. For color conversions, the two color profiles in the folder *Icc/* are used. Color profiles can be downloaded from the links provided in the directory *Icc/*. The color profiles that are applied must be named *CMYK.icc* and *sRGB.icm*.

Alternatively color profiles can be downloaded from the following websites:

- www.pdf-tools.com/public/downloads/resources/colorprofiles.zip
- www.color.org/srgbprofiles.html

In order to change the default color profile, replace these two color profiles with new color profiles.

If no color profile is available, the conversion is done algorithmically, using Neugebauer.

4.9 Dithering

Dithering is a common means used in images to simulate colors that are not available as actual colors. Its use is best observed in image with a low color depth, where colors or shades of grey need to be simulated with other colors (e.g. only black/white pixels).

July 1, 2015

Remarks

1. All images below have quite a low resolution. As a result the effects of the different dithering types become more obvious. The higher the resolution and the larger the number of colors, the higher the quality of the image.
2. The rendering filter and current zoom level of the PDF viewing application may have an additional impact on how the images below are displayed.

Color Images

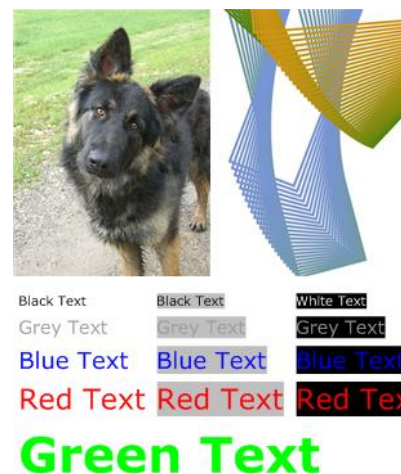
Color Space: RGB (24 bit)

Dithering: **None**

File Size as PNG: 129 kB.

A 24 bit RGB color image can have up to 16.7 millions of different colors. Dithering does not need to be applied since all required colors exist and none need to be simulated.

- + Highest quality
- Highest file size

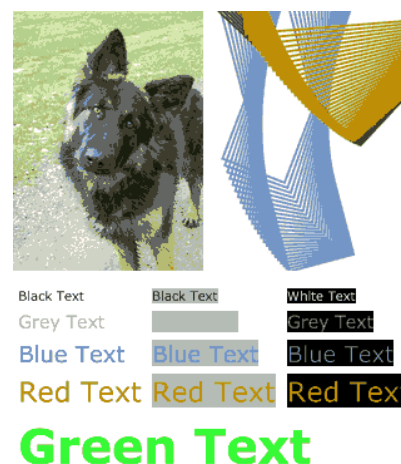


Color Space: 16 colors (4 bit)

Dithering: **None**

File Size as PNG: 16 kB

- + Small file size
- + Works well for images with a small number of colors (artificial images, text)
- Does not work well for images with lots of colors (photographic images) - parts of the image can become plain-colored and details get lost.



July 1, 2015

Color Space: 16 colors (4 bit)

Dithering: **Floyd-Steinberg**

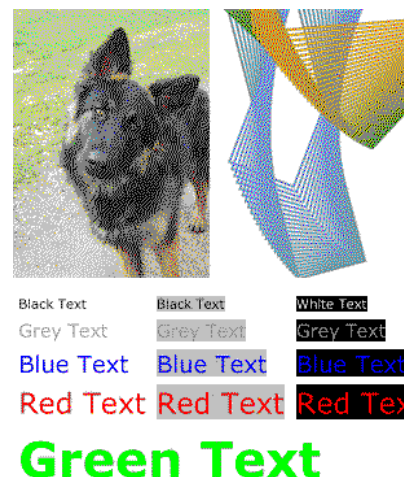
File Size as PNG: 18 kB

+ Renders details better

+ Usually better overall quality, specially in photographic images than without dithering

- Sometimes generates unwanted artifacts (striking pixels)

- Larger file size then without dithering

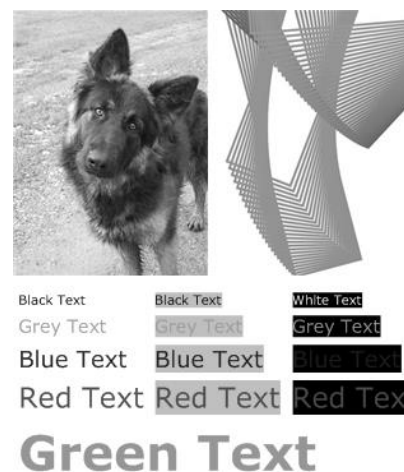
**Bi-tonal Images**

Color Space: Grayscale (8 bit)

Dithering: **None**

File Size as PNG: 46 kB

(The 8 bit image just acts as reference.)



Color Space: Grayscale (1 bit)

Dithering: **None**

File Size as PNG: 2.6 kB

+ Smallest File Size

+ Works well for documents with high contrast (black text on white background)

+ Does not generate artifacts

- Details get lost, because shades of gray are not approximated, but converted to either black or white (in fact images or part of them can become completely black or white)



July 1, 2015

Color Space: Grayscale (1 bit)

Dithering: **Floyd-Steinberg**

File Size as PNG: 9 kB

- + Generally higher quality, specially of phtographic images
- + Can approximate any shade of gray
- Larger file size than without dithering
- Generates artifacts (e.g. a very bright gray paper is approximated by far-spread single black pixels)
- Not well suited for text, unless the color of the text must be reflected

Color Space: Grayscale (1 bit)

Dithering: **Halftone**

File Size as PNG: 4 kB

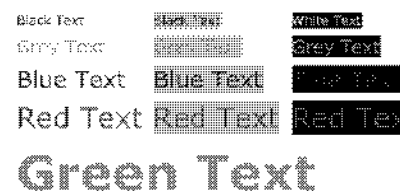
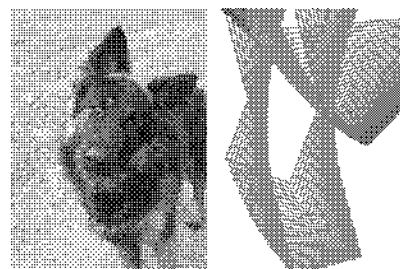
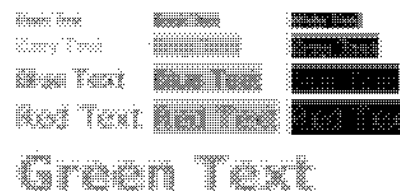
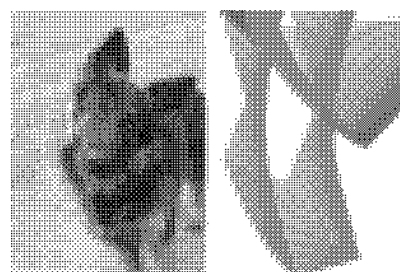
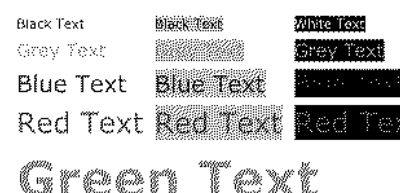
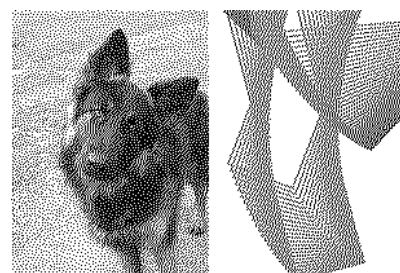
- + Small file size
- + Approximates shades of gray
- Not well suited for text or artificial images

Color Space: Grayscale (1 bit)

Dithering: **Pattern**

File Size as PNG: 5 kB

- + Works acceptable for all types of content (text, photographic images, artificial images)
- Is not excellent in any type of content



Guidelines

As seen in the examples above, different types of dithering behave different for different types of content. Below are some suggestions, which dithering type is normally best for a give type of content:

Text, OCR

No dithering

July 1, 2015

Artificial images with few colors and no bright colors No dithering

Artificial images with many colors Test which dithering type yields the best result

Photographic images Floyd-Steinberg

Mixed content Test which dithering type yields the best result

Mixed content, high-resolution For resolutions above 300 dpi, Floyd-Steinberg almost always yields the best result (exception: for pure black text on white background, use no dithering)

Keep in mind that dithering should only be applied for images with a low color depth, such as black and white (1 bit). Dithering for images with a color depth of 8 bit or higher (256 colors or grey scale) has little to no visual impact.

5 Reference Manual

5.1 Supported Codecs

The following table lists which capabilities of the different codecs are supported by the PDF to Image Converter.

Table: Codecs					
Codec	Bits Per Pixel	Gray	Indexed	Quality	Compression
TIFF	1, 2, 4, 8, 24*	Yes	Yes	Yes	Raw, Flate, LZW, JPEG, Group3, Group3_2D, Group4
JPEG	8, 24	Yes	No	Yes	JPEG (Lossy only)
BMP	1, 2, 4, 8, 24*	Yes	Yes	No	Raw
GIF	2-8	Yes	Yes	No	LZW
PNG	1-8, 24	Yes	Yes	No	Flate
JBIG2	1	Yes	No	Yes	JBIG2 (Lossless: Q = 100)**
JPEG2000	8, 24	Yes	Yes	Yes	JPEG2000 (Lossless: Q = 100)**
PBM	1-8, 24	Yes	No	No	Raw
EPS	1, 2, 4, 8, 24*	Yes	No	No	Raw

Codec: The **Compression/Decompression** Type

Bits Per Pixel: The supported values for bits per pixel. 1 = bi-tonal, 8 = 256 colors/grey scales, 24 = true color

Gray: This format supports grey scale

Indexed: This format supports indexed colors

Quality: This format supports the setting of a quality parameter

Compression: Supported compression types

*) For palette creation: The number of palette entries is equal to $2^{\text{BitsPerPixel}}$ where BitsPerPixel is smaller or equal to 8. This means it is possible to create a 3 bits per pixel TIFF, BMP or EPS, but the palette size is equal as for 4 bits. However the 3 bits per pixel image will compress better than the 4 bits per pixel image.

**) To create lossless JBIG2 and JPEG2000 images, set the quality parameter to 100. For values <100, a lossy compression algorithm is applied.

5.2 Rendering Options

-1 Create one Image File per PDF Page

The PDF to Image Converter creates multi-page Tiffs by default. For other formats that do not support multi-paging, it creates single page output files.

Example: To generate one TIFF image for every page of the PDF input file, the switch `-1` can be used:

```
pdf2img -1 input.pdf
```

This will generate a series of one-page Tiff images like this:

```
input_1.tif, input_2.tif, input_3.tif, etc.
```

Example: To specify the output name and directory, use a command like this:

```
pdf2img -1 input.pdf ../output_%d.tiff
```

The output files will then be named

```
output_1.tiff, output_2.tiff, output_3.tiff, etc.
```

-b Set the Bits per Pixel

How many bits are used to represent one pixel.

The argument for this parameter depends on the image format (see table Codecs).

For JPEG this option has two allowed values: 8 for grey scale and 24 for true color.

Example: The following command creates a grey scale JPEG image:

```
pdf2img -b 8 input.pdf output.jpg
```

When using 1 bit per pixel, it is suggested to disable anti-aliasing (option `-oq`) and set a suitable dithering algorithm (option `-h`).

-cms Set the color management engine

The transformation of colors from one color space to another is performed using a color management engine. The default is the built-in engine 'lcms'. Other choices are 'msicm' which denotes the Windows CMS and 'neugebauer' which is an algorithmic engine which doesn't work with profiles. If this switch is set to 'none' then the algorithms specified in the PDF reference are used.

Example: The following command selects the neugebauer color management engine:

```
pdf2img -cms neugebauer input.pdf output.jpg
```

-cn Center Mode

Center the PDF. This option is useful in combination with setting page dimensions.

-cs Set the Color Space

This options sets the color space. Supported values are:

Table: Color Space	
Value	Description
0	Grey-scale
2	RGB
4	CMYK
7	Indexed
9	CMYK with K only

-d Set the Resolution

The default resolution is set to 150 dpi (dots per inch). This switch is redundant to the specialized switches **-dx** and **-dy**, meaning the last switch set on the command takes precedence.

Example: To set the resolution to a value to 100 dpi use the switch like this:

```
pdf2img -d 100 input.pdf
```

-dx Set the X-Resolution

Set the resolution only for the X-axis. The default resolution is set to 150 dpi.

Example: Set the resolution in X to 72 dpi.

```
pdf2img -dx 72 input.pdf
```

-dy Set the Y-Resolution

Set the resolution only for the Y-axis. The default resolution is set to 150 dpi.

-f Fit Page Mode

Make the PDF fit the page (in either width or height). This option is useful in combination with setting page dimensions.

Example: Convert a PDF to 800 by 600 pixel TIFF image and scales the page of the PDF to fit the page of the image.

```
pdf2img -s 800 600 -f input.pdf output.tiff
```

-fax Convert to Class F

This switch creates a Class F TIFF which is used by fax machines. There are two types:

Table: Class F TIFF		
Value	Description	Is equal to the following settings

July 1, 2015

s	standard	<code>-rp -sp 1728 0 -dx 204 -dy 98 -g3</code>
h	high resolution	<code>-rp -sp 1728 0 -dx 204 -dy 196 -g3</code>

Example: Convert a PDF to a standard Class F Tiff

```
pdf2img -fax s input.pdf output.tiff
```

-fo Bit Fill Order

Set the fill order of bits used in fax compressions.

Table: Bit Fill Order		
Value	Description	
1 (default)	Most Significant Bit (MSB) first.	
2	Least Significant Bit (LSB) first.	

-fs Filter Size

This setting is used to enable and parameterize super-sampling, a technique to initially render the image at a higher resolution and then sample it down to the target resolution. As a result of that process the final image appears smoother, i.e. anti-aliased.

Applying super-sampling improves the image quality when rendering at low target resolutions (72 dpi or less); the higher the target resolution the less the visual impact.

This setting requires memory and CPU time quadratically to the ratio, therefore only small values, such as `-fs 2` or `-fs 3` should be used.

If a too high value (in combination with the original image size) is set, it is ignored.

-g Gray Color Space

No longer supported: Use the switch `-cs` instead.

-h Dithering Mode

Set the dithering mode. Allowed values are:

Table: Dithering Mode		
Value	Description	
0	no dithering	
1 (Default)	Floyd-Steinberg	
2	Halftone block	
3	Halftone continuous	
6	Atkinson dithering is very fast and produces images that can be	

July 1, 2015

compressed really well with a reasonably good image quality.

Dithering provides a better image quality, especially for 1 bit images, at the cost of a larger file size.

Example: Disable dithering for a bi-tonal image.

```
pdf2img -h 0 -b 1 input.pdf
```

-i Indexed Color Space

No longer supported: Use the switch `-cs` instead.

-lf Set Line Width Multiplication Factor

In cases where lines are rendered with an undesired width (e.g. too thin) the switch `-lf` can be applied to scale them.

This option only affects lines. It has no influence when lines are drawn in any another way as by using the PDF operators `m` and `l` (move to, line to). It does not affect text unless text is drawn with lines instead of using a font.

-lw Set Minimum Line Width

In cases where lines are rendered too thin, a minimum line width in PDF points can be defined. Any line will then be printed with at least the defined minimum line width. Note that as a result, thin and very thin lines can no longer be distinguished. On order to scale all lines with a certain factor, use the switch `-lf` instead.

This option only affects lines. It has no influence when lines are drawn in any another way as by using the PDF operators `m` and `l` (move to, line to). It does not affect text unless text is drawn with lines instead of using a font.

-m Rendering Mode

Deprecated. Version 2.0 and higher only support one rendering mode.

-o1 Use Type1 Fonts

This switch converts OpenType fonts to Type1 fonts.

-oe Do not Use Embedded Fonts

This switch disables the use of embedded fonts and replaces them with fonts installed on the operating system.

-ofp Use Pre-Installed fonts

When using this switch embedded fonts are replaced by fonts installed in the operating system. Only non-symbolic fonts that have an equal name are replaced. The default behavior is that all embedded fonts are used.

July 1, 2015

-op Disable Pattern

This switch disables patterns.

-oq Disable High Quality Rendering

This switch disables anti-aliasing.

Compatibility Note: In previous versions, the switch also disabled filtering, which is no longer the case with the current version.

-ot Use TrueType Fonts

This switch converts OpenType and Type1 fonts to TrueType fonts.

-p Read an Encrypted PDF File

When the input PDF file is encrypted and has a user password set, (the password to open the PDF) the password can be provided with the switch **-p**.

Example: If the user password were "userpwd", then the command to read and process the encrypted PDF would look like this:

```
pdf2img -p userpwd input.pdf
```

When a PDF is encrypted and the user password is not provided or is incorrect, pdf2img cannot decrypt and read the file. Instead it will generate the following error message:

```
Password wasn't correct.
```

-pg Set Page Range

With this switch the page range to be converted can be set. Note that the output image format must support multi-page. TIFF and JBIG2 support multi-page.

Example: Convert only pages 1 through 3.

```
pdf2img -pg 1 3 input.pdf output.tif
```

-pgs Set of Pages

A set of pages can be defined using single pages, ranges of pages and comma-separated combinations thereof.

Example: Convert pages 1, 2-4, 6 and 10.

```
pdf2img -pgs 1,2-4,6,10 input.pdf output.tif
```

-q Set Image Quality of Lossy Compression

This switch sets the image quality of images that are compressed using a lossy compression algorithm such as Jpeg, Jpeg2000 or Jbig2. For images which support

July 1, 2015

lossy and lossless compression, such as Jpeg2000 or Jbig2, a quality value of 100 means lossless compression. (Note that Jpeg is always lossy.)

The default value is 75.

A lower value of the quality parameter results in a higher compression at the cost of visual quality loss.

Example: Lossy compression with a quality index of 50:

```
pdf2img -q 50 input.pdf output.jpg
```

Example: Lossless compression:

```
pdf2img -q 100 input.pdf output.jp2
```

-r Rotate Pages According to Page Rotate Attribute

A PDF document can have a page rotation attribute, that describes if the PDF is to be rotated when displayed (for example when a 90 degree rotated portrait is displayed as landscape). The PDF to Image Converter by default ignores this attribute. Using the switch **-r**, the generated images have the same rotation as the input PDF.

-rl Rotate Pages to Landscape

This switch rotates all pages to landscape.

-rp Rotate Pages to Portrait

This switch rotates all pages to portrait.

-s Set Width and Height of Image in Points

When not specified, the image will have the same dimensions as the input PDF. To set the dimensions manually, use the switch **-s** followed by two parameters for the width and the height. If either of the dimensions is set to 0, the value will be calculated proportionally based on the other value.

Example: The following command will generate an image that is 400x300 points.

```
pdf2img -s 400 300 input.pdf
```

-sp Set Width and Height of Image in Pixel

This option can be used to set the dimensions of the image in pixels manually. If either of the dimensions is set to 0, the value will be calculated proportionally based on the other value.

Example: The following command creates a file with a width of 1024 pixels, the height is calculated proportionally.

```
pdf2img -sp 1024 0 input.pdf
```

-sa Set Width and Height in Pixel and Preserve Ratio

The switch `-sa` has the same effect as `-sp`, but the aspect ratio is preserved. This means you can specify the maximum size allowed, the image will then be scaled to fit one of the dimensions.

Example: A 400 by 400 points PDF is converted with the option `-sa 600 800`. The ratio width to height of the input file is preserved, which is 1:1. The maximum that fits in 600 by 800 pixels is therefore a 600 by 600 pixel image.

-t Set Threshold When Dithering is Disabled

When producing bi-tonal images, e.g. by the switches `-b 1`, `-g3` or `-fax`, then the dithering mode is set with the `-h` switch. If "no dithering" is selected (`-h 0`) then the conversion of color or grayscale images is done by applying a threshold. All pixels with brightness above the threshold are converted to white and all others to black.

The switch `-t` can be used to set a custom threshold. The allowed range is 0 (all white) to 255 (all black). The default value is 181.

5.3 Compression Options

Compression options can only be set for TIFF images. For all other image types the compression is defined through the image format (.gif, .jpg, etc.).

Note that not all image viewer support all compression types for TIFF.

-g3 Compression CCITT Group 3

CCITT Group 3 is the predecessor to CCITT Group 4, it's a simpler algorithm that normally results in a weaker compression level.

-g4 Compression CCITT Group 4

CCITT Group 4 is the standard compression for bi-level images (i.e. facsimile).

-l Compression LZW

LZW (Lempel-Ziv-Welch) compression is a lossless compression algorithm for images. Please consult the copyright laws of your country prior to using this compression algorithm.

-j Compression JPEG

TIFF allows images to be compressed with JPEG, which is a lossy compression algorithm. JPEG provides a high level compression of 8 and 24 bit images. It is best

July 1, 2015

suited for pictures, but not for text images. The image quality parameter is set to 75 per default.

-j6 Compression JPEG (6)

This is an older version of JPEG. Certain (older) image software may support this compression, but not the newer version of JPEG (e.g. Photoshop 8).

-z Compression ZIP

ZIP is a lossless compression algorithm. It is useful for the compression of large images with no loss in quality.

5.4 pdf2pdfimg Specific Options

-nl Disable the copying of links

Link annotations are not copied to the output document if this switch is used.

-no Disable the copying of outlines

Outlines (Bookmarks) are not copied to the output document if this switch is used.

-nv Disable the copying of viewer preferences

Viewer preferences are not copied to the output document if this switch is used.

5.5 General Options

-v Verbose Mode

This option turns on the verbose mode. In the verbose mode, the steps performed by pdf2img are described.

Example: The command below

```
pdf2img -v input.pdf
```

will document the performed steps, for example:

```
Rendering page 1 ...
```

```
Rendering page 2 ...
```

```
...
```

July 1, 2015

-lk Set License Key

Pass a license key to the application at runtime instead of installing it on the system.

5.6 Return Codes

All return codes other than 0 indicate an error in the processing.

Table: Return Codes	
<i>Value</i>	<i>Description</i>
0	Success
1	PDF Input File could not be opened or invalid parameters
2	PDF Output File could not be created
3	Invalid option or option values were entered
4	PDF Input File is encrypted and password is incorrect or not provided

6 Tips, Ticks and Troubleshooting

6.1 General

Pages Are Rotated

Use the switch `-r` to generate images with the same rotation as in the input PDF.

Generated Files Have a Large Size

Read in the chapter "User's Guide" "How to reduce the file size".

Images Are of too Low Quality

Increase the DPI value to increase the resolution. This is done with the switch `-d`. For lossy compression algorithms, such as JPEG, increase the quality parameter, e.g. `-q 85`.

Image Does Not Contain the Whole Content

This can happen when the switch `-s` is used to set dimensions that have a different ratio than the original dimensions. To automatically make the page fit the new dimensions, use the switch `-f`. Alternatively ensure the page dimensions of the image are large enough to hold the complete page.

Colors Are Gone

The switch `-b` allows you to set the bits per pixel. For JPEG 8 bit is always grey scale, since indexed colors are not supported for this format. For TIFF and GIF, the indexed colors need to be enabled if 8 bit is selected. This is done with the switch `-i`.

6.2 Text

Font Replacement Strategy

This section describes the exact behavior of font handling of the rendering engine. It is rather technical and it is not required to be understood in order to properly use the software.

The following steps are performed sequentially in the search of a font. If a font is found, the search is stopped; otherwise the next step is performed.

1. If the font is not embedded or -ofp is set:
 - a. If the font name appears in the [Replace] section in the configuration file "fonts.ini" the name is replaced and looked up in the installed font collection
 - b. if it is a standard font¹ it is replaced by the equivalent TrueType font name and it is looked up in the installed font collection
 - c. If the font name appears in the [Fonts] section in the configuration file "fonts.ini" the name is replaced and looked up in the installed font collection
 - d. If the font has "Italic" or "Bold" in its name the font without these styles is looked up in the installed font collection
2. If a font name is looked up in the installed font collection then the name compare is performed as follows:
 - a. PostScript name
 - b. TrueType name without blanks (a missing style is interpreted as "Regular" or "Normal")
 - c. TrueType name without modifications
3. If the font is embedded, it is converted to a Windows compatible font and temporarily installed. If -oe is used then the glyphs of the fonts are converted to either bitmaps or outlines². If -oo is used then the glyphs are converted to outlines only.
4. If the font is not embedded and the Unicodes are available then the nearest font from the installed font collection is tailored to the metrics of the font.
5. If the font is embedded then it is converted to outlines.
6. In all other cases the nearest font from the installed font collection is used

¹ e. g. Times–Roman, Helvetica, Courier

² The outline of a glyph is a vector graphic without any reference to the original font program.

Installed Font Collection

The installed font collection contains fonts from the directories %SystemRoot%\Fonts and "Fonts", which must be a direct sub-directory of where the product's executable resides.

The fonts and their properties are cached in a font cache, located in the files %TMP%\font-database*. It is recommended to clear the cache, if you add or remove fonts from the font directories.

Using the Font Mapping File *fonts.ini*

"fonts.ini" is a configuration file to map fonts used in the PDF to fonts pre-installed on the system. The mapping file must reside a directory named "Fonts", which must be a direct sub-directory of where the main DLL or executable resides.

The mapping file is optional. It consists of two sections: [fonts] and [replace].

Both sections are used to map fonts in the PDF to fonts in the installed font collection on the operating system. This comes into play when the font in the PDF document does not have an embedded font program, or the embedded font is not usable.

The mapping only works if the font types of the specified fonts are matching; e. g. if the font in the PDF is a symbolic font, such as "Symbol" or "ZapfDingBats", the mapped font must be symbolic too.

The section [fonts] is only considered if the font-matcher does not find an appropriate font amongst the existing installed fonts. It is suggested to only use this section.

The section [replace] is stronger and applied before the font-matcher. This means a font will be replaced as defined, even if the correct installed font is available on the system.

Syntax

The syntax of the mapping file is this:

```
[fonts]
PDF_font_1=installed_font_1,{font_style}
PDF_font_2=installed_font_2,{font_style}

[replace]
PDF_font_n=installed_font_n,{font_style}
```

PDF_font_* is the name of the font in the PDF. This name can be found in one of the following ways:

- Use any tool that can list fonts. Such as [3-Heights PDF Extract](#) or [3-Heights PDF Optimization](#). Ignore possible prefixes of subset fonts. A subset prefix consists of 6 characters followed by the plus sign. For example "KHFOKE+MonotypeCorsiva", in this case only use "MonotypeCorsiva" as font name in the mapping file.

July 1, 2015

- Open the document with Adobe Acrobat, use the "MarkUp Text Tool" , mark the text of which you would like to know the font name, right-click it, select "Properties..."

installed_font_* is the font family name of the installed font. To retrieve this name, find the font in the Windows' font directory and open it by double-clicking. The first line in the property window displays the font family name (this may vary depending on the operating system). The font family name does not include font styles; so an example of a font family name is "Arial", but not "Arial Italic".

font_style is an optional style, that is added coma-separated after the font family name. The style is always one word. Examples of font styles are "Italic", "Bold", "BoldItalic".

Example

[fonts]

Ryumin-Light=MS Mincho

GothicBBB-Medium=MS Gothic

[replace]

ArialIta=Arial,BoldItalic

Other Ways to Deal with Text Issues

The following list provides possible work-arounds if text is printed incorrectly. Options should be tried in ascending order.

1. Using the option (*-oe*) inhibits all embedded fonts from being used in the spool file and the printer hardware. Instead the glyphs are converted to either bitmaps or outlines. Using the option (*-oo*) at the same time the conversion is restricted to outlines.
2. Using the option (*-ofp*) inhibits embedded fonts which have the same name as the corresponding installed font from being used.
3. Pre-render the page in a bitmap and use the pre-rendered image (*-ob*).

6.3 Transparency

The 3-Heights™ rendering engine supports transparency functions such as a number of blend modes as well as isolated and non-isolated transparency groups, but not transparency in general.