HELIOS

User manual

PDF HandShake G8

(Version 6.0.0)

User manual

September 10, 2020

Contents

1 About the chapters of this manual

In the following, we give a brief summary of each chapter in this manual. These summaries will help you find the information you are looking for.

Chapter 2 “An introduction to HELIOS PDF HandShake” discusses the use of the PDF file format as an exchange format in a prepress environment or in the printing business.

Installation of PDF HandShake

Chapter 3 “Installation” lists the requirements for the HELIOS Acrobat plug-ins, and describes the installation of the Mac tools.

Preparations

Chapter 4 “Before getting started” summarizes some general information about the structure and behavior of PDF HandShake.

Chapter 5 “Set up PDF HandShake with HELIOS Admin” shows how to choose suitable PDF HandShake settings via the HELIOS Admin program on a client. The chapter describes the relevant dialogs, including options for HELIOS ImageServer users.

Chapter 6 “PDF HandShake utility programs” describes the HELIOS programs of which PDF HandShake is composed, and describes in detail how to change options and attributes of these programs manually.

The print-only workflow

Chapter 7 “Printing with PDF HandShake” describes the “pdfprint” command line tool that allows printing PDF files directly from the server without opening an application. It also describes how to specify the same print options via the PDF HandShake print plug-in for Acrobat on Mac. Using either the plug-in or “pdfprint”, you can e.g. print separations and make use of the PDF HandShake color matching features.

Chapter 8 “Create PDF Server” is a PDF HandShake feature to automate the PDF creation of any PostScript job. Configuration and setup is easy and comfortable through HELIOS Admin. In addition, the chapter shows how to set up a lower quality “Create PDF” queue, e.g. for bitmap documents in an office environment, using the Ghostscript engine instead of the Distiller for PDF creation.

The OPI workflow (HELIOS ImageServer required)

Chapter 9 “Use PDF files in a PostScript OPI workflow” describes how to use PDF files in an OPI environment. It shows how to tag color profiles to PDF high-resolution files, and explains the default behavior of the OPI server when handling PDF files.

Chapter 10 “Export for Imposition with ImageServer” shows how to export PDF files (as PostScript files) for further use in an imposition application. Final printing from the imposition software to the output device requires HELIOS ImageServer.

Chapter 11 “PDF-native OPI workflow” is about an essential component of high-volume PDF document workflows based on Adobe InDesign software.

Additional features

Chapter 12 “PDF-to-PDF color conversion” introduces a hot folder based technology, which allows converting the color space of each object within a PDF document to a specified output color space.

Chapter 13 “Preferences” lists the PDF HandShake preferences and explains what they effect when set.

Chapter 14 “Example workflow” illustrates a possible production environment where a customer uses Microsoft Word for document creation and submits PDF files to his printers.

Chapter 15 “Create PDF files using Acrobat Distiller” gives instructions for how to set up the job options of the Acrobat Distiller software in order to create PDF files that are best suited for being used with PDF HandShake.

Chapter 16 “PDF Transparency” briefly lists PDF transparency requirements and features.

Chapter 17 “PDF layers” gives information on layers in PDF documents and shows how these can be viewed, selected and printed via PDF HandShake and its utilities.

Additional information

Appendix A “About fonts” gives some background information about fonts, different types of fonts and the problems that can arise during printing. The appendix also explains how PDF HandShake deals with fonts.

Appendix B “The fonts we deliver” contains a list of the PostScript Type 1 fonts we include in our software package.

Appendix C “Glossary” explains some product-related terminology.

2 An introduction to HELIOS PDF HandShake

PDF HandShake fully utilizes the benefits of PDF in existing production environments. Automated server-based PDF generation, PDF printing, PDF-native OPI, PDF-to-PDF color conversion, and batch tools are the striking features of this product.

Note: HELIOS PDF HandShake runs on top of the foundation provided by HELIOS Base. Please read the HELIOS Base manual for installation instructions and other important details.

2.1 About PDF

The PDF file format allows storing illustrations, text, and images in one file. It supports several color spaces (e.g. RGB, CMYK or Lab), compression modes such as JPEG, JPEG 2000, JBIG2, ZIP, CCITT, and the font types PostScript and TrueType.

With PDF HandShake, PDF becomes a perfect exchange format. Logos, illustrations and complete documents can be provided as PDF and used for any workflow, depending on your individual requirements. The PDF files can be transformed into EPSF (HELIOS ImageServer required) and then placed in other documents or they can be printed directly or exported for imposition.

2.2 New PDF HandShake features

For new features in the PDF HandShake software see the HELIOS website:
www.helios.de Go to *HELIOS Product Versions – New Features*

For HELIOS Base, the foundation used by all HELIOS products, see the HELIOS Base product web page:
www.helios.de Go to *Products* > *Base*

2.3 Product description

PDF HandShake is fully integrated with EtherShare, PCShare, WebShare, and optionally with ImageServer.

PDF HandShake offers the following features:

- Easy administration via HELIOS Admin
- Printing separations of composite PDF files
- Editing and printing pre-separated PDF documents
- Color matching
- Create PDF Server: Automatic PDF generation via HELIOS printer queues
- PDF transparencies are fully supported

PDF HandShake has built-in ICC based color management, co-developed by Heidelberg and HELIOS. For your convenience, we include in our software package a number of ICC profiles for different devices. Note that PDF HandShake recognizes embedded ICC profiles in PDF files and supports *tagging-by-reference*, where only a reference to the profile is embedded.

- Proof printing
  You can define profiles for both your final output device (printer or press), and a proof printer (e.g. an inkjet printer). Thus, if you print to the proof printer, you will obtain a very close simulation of the colors your final output device would produce.

If you are using ImageServer as well, additional features become available for PDF files, e.g.:

- Automatic generation of EPSF layouts from PDFs
- Transformation of PDF into EPSF high-resolution files
- Export of PDF files for imposition and final printing of the PostScript jobs that come from the imposition program
- ICC profile tagging
- Downsampling
- PDF-native OPI (resolve image references in PDF files)
- Full shading support

Please note that OPI comments that are included in a PDF file (including nested OPI references) can be resolved by ImageServer.

HELIOS PrintPreview

PrintPreview offers powerful preview options for all HELIOS print jobs. For details refer to the PrintPreview manual, which is available on the “HELIOS Applications” volume, and on the HELIOS website.

Flexibility

The PDF HandShake product consists of:

- Server modules:
  - PDF extension for EtherShare, PCShare, WebShare, ImageServer, and PrintPreview

❍ Server command line tools (see 6 “PDF HandShake utility programs”):

- “pdfprint”
- “pdfcat”
- “pdfform”
- “pdfinfo”
- “pdfnote”
- “pdfresolve” (requires HELIOS ImageServer)
- “pdftoeps” (requires HELIOS ImageServer)

❍ Mac tools:

- “PDF HandShake” print and export plug-in for Acrobat on Mac

Each of these modules supports a specific workflow. Together, they offer maximum flexibility when using the PDF file format.

2.4 Known limitations

The following non-printing related features are not supported in PDF HandShake:

- Embedded QuickTime/Flash multimedia data
- Printing of Acrobat forms
- ImageServer and PrintPreview previews of PDF patterns will be displayed as gray areas

3 Installation

Complete installation instructions for all HELIOS products can be found in the HELIOS Base manual.

PDF HandShake Acrobat plug-ins (Mac OS only)

The “HELIOS Applications” volume contains all installable applications that are part of the HELIOS product suite. The volume always contains the latest versions of the applications. If you install any HELIOS updates by means of the update installer, the contents of this volume will be updated automatically and the items in this volume can then be used to replace older copies of the applications that exist on the local hard disks of your Mac client computers.

Refer to the “README.txt” file (“HELIOS Applications > MacOS > PDF Tools > Acrobat Plug-Ins”) for installation details on different Acrobat versions.

4 Before getting started

Support for protected PDF files

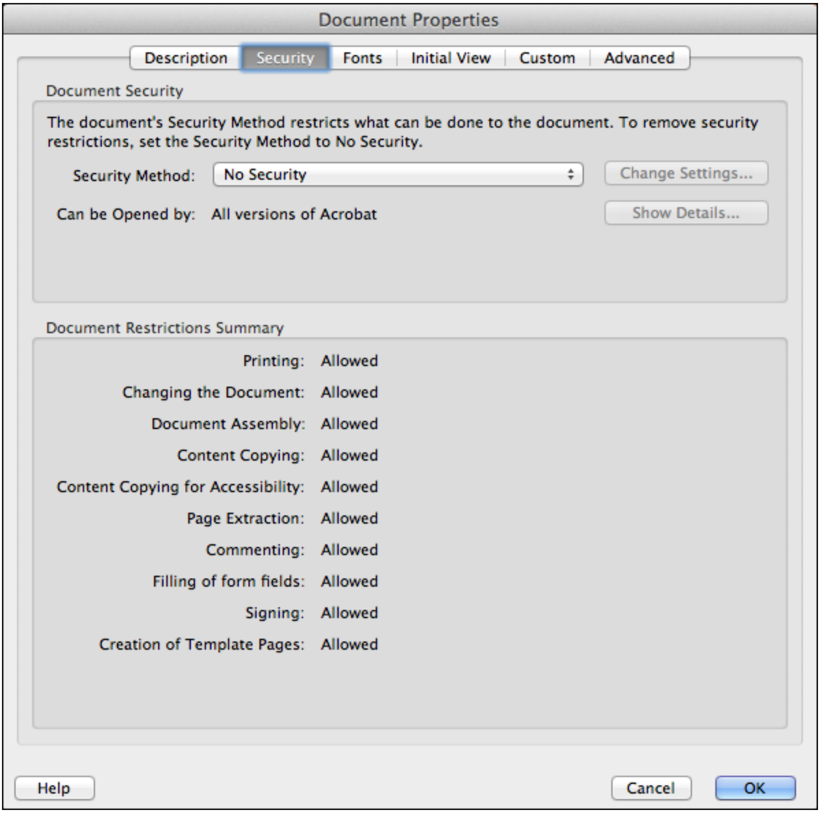
PDF HandShake checks the security settings in a PDF document. All documents with an unchecked “Require a password to open the document” option are supported, even if other security options are specified.

➢ To find out whether a document is protected by security settings, use the Acrobat “Document Properties” window, as shown in Fig. 4.1.

“Printing” for the document must be allowed in high resolution. Otherwise, the menu items `PDF HandShake Print` and `PDF HandShake Export` will be grayed out, and OPI layout generation will be disabled for this particular file.

A password to open the document – if there is one – must be removed in this way as well. Otherwise the document cannot be handled by PDF HandShake.

Important: Changing settings requires knowledge of the Security Password. Thus, for proper operation, refer to the Acrobat documentation.



Document Security

The document's Security Method restricts what can be done to the document. To remove security restrictions, set the Security Method to No Security.

Security Method: No Security

Can be Opened by: All versions of Acrobat

Document Restrictions Summary

Printing:	Allowed
Changing the Document:	Allowed
Document Assembly:	Allowed
Content Copying:	Allowed
Content Copying for Accessibility:	Allowed
Page Extraction:	Allowed
Commenting:	Allowed
Filling of form fields:	Allowed
Signing:	Allowed
Creation of Template Pages:	Allowed

Fig. 4.1: Checking security options with Acrobat

Composite and separated PDFs

PDF HandShake can handle composite *and* pre-separated PDF documents.

Whether you print composite or separations from your original application (before generating PDF) depends on your workflow and requirements. For example, if you wish to include trapping information in your PDF file, you may be forced to print separations. This is due to the fact that some applications such as QuarkXPress do not include trapping information in the PostScript output if you print composite.

Pre-separated PDF documents are automatically recognized by the “pdfprint” command line program and by the PDF HandShake Acrobat plug-ins. When printing or exporting pre-separated documents with PDF HandShake, the following rules apply:

❍ `Print separations` is enabled

❍ The options `In-RIP Separations`, `Convert Spot To Process`, and `Overprint Black` are disabled

❍ Color matching (`PDF RGB/CMYK/Gray Profiles)` is disabled.

❍ The halftone settings for each color are applied to the pages with the corresponding plate colors

Pre-separated PDF documents are also recognized by ImageServer. They are handled as follows:

❍ The “layout” and “pdftoeps” command line programs generate DCS files with default composite previews, which are raster based. The “layout” program creates single-file DCS-style layout images while the “pdftoeps” program creates DCS-1 multi-file images or DCS-2 single-file images. The plate file suffixes for CMYK will be *.C*, *.M*, *.Y*, and *.K*. Spot color plate files will be assigned other suffixes, namely letters so far unused, in alphabetical order. The suffix does not have any relation to the name of the spot color.

❍ Printing separations and printing composite of documents containing placed layouts of pre-separated PDF files to an OPI printer queue (with image replacement) is fully supported for all output devices, except for PrintPreview.

Note: Composite printing of documents containing placed layouts of pre-separated PDF files to a PrintPreview queue will lead to blank picture boxes in the preview.

Access to fonts

You may run PDF HandShake in demo mode. Then, our server fonts will only be available for OPI layout generation – they cannot be used for printing. The only exception is the "Courier" font, this font is always fully accessible. For background information about the different types of fonts and the way they are handled by PDF HandShake, please see A "About fonts".

About compressions

Compressed PDF documents and compressed images in PDF documents can be handled without problems. PDF HandShake recognizes all common modes of compression like JPEG, JPEG 2000, JBIG2, CCITT, and ZIP.

Note that if you are using ImageServer and convert a PDF file into EPSF (either high-resolution EPSF or layout), this new EPSF file may – depending on server settings – not be compressed.

About separations and spot colors

Spot colors in PDF files are recognized. PDF HandShake even offers a `Spot To Process` option that lets you convert all spot colors in a PDF file into CMYK, when printing separations. This option is available if you print directly from your server platform, using the "pdfprint" command or if you print from a Mac client using the HELIOS Acrobat plug-in.

About source profiles for PDF documents

PDF files can contain many color objects based on different device color spaces without ICC profile information. Therefore, for correct color matching, these PDF files should have one source profile for RGB objects and one source profile for CMYK objects in the document. Optionally, you can specify a source profile for Grayscale objects in the document, too. If you are using

ImageServer you can, with the "Tagger" program, assign profiles to your PDF documents. Otherwise, you can define server-wide defaults using HELIOS Admin or tag PDF documents temporarily before printing. The HELIOS Acrobat plug-in provides pop-up menus for that purpose, and the "pdfprint" program also offers the required parameters (see 7.1 "pdfprint").

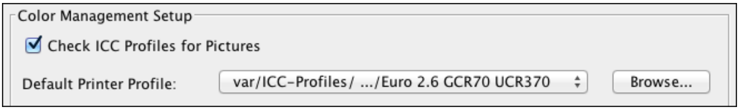
Profile information

If you receive PDF files from a customer you may check whether these files already come with profile information that has been "tagged" to the files with ImageServer. If so, the profile names will be displayed in the `Comments` field of the Mac OS 9 "Info" box, or via "HELIOS Meta" on OS X and Windows clients. WebShare clients can get profile information from the comments line which is included in the sharepoint directory list. ImageServer users may check profile information with the "HELIOS ICC Tagger" program.

About color data transformation without OPI

If you are using PDF HandShake with EtherShare, PCShare or WebShare only, your server will behave as follows.

With ICC printer profile:



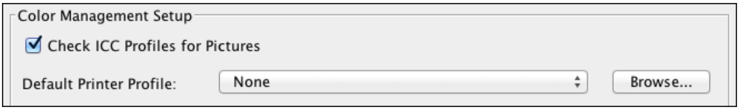
Color Management Setup

☑ Check ICC Profiles for Pictures

Default Printer Profile: var/ICC-Profiles/ .../Euro 2.6 GCR70 UCR370

On principle, you activate the color matching module explicitly by selecting an ICC printer profile on the desired printer queue (this is described in 5.2.2 "Printer queue settings for PDF"). If you do so, all colors in your PDF files will be transformed into the printer's color space during printing.

Without printer profile:



Color Management Setup

☑ Check ICC Profiles for Pictures

Default Printer Profile: None

If you do not select a printer profile, color data transformations might be performed “behind the scenes” – depending on whether you print composite or separations:

- *Printing composite:*
  All color spaces in your PDF files remain as they are. The color matching module will not interfere.
- *Printing separations:*
  Indirect color matching will be performed. Non-CMYK color spaces in your PDF files (like CIELab, RGB or Indexed colors) will automatically be transformed into CMYK. The server will use the “CMYK standard” for the color data transformations. By default, this standard is Euroscale – you can only change it to SWOP if you are using ImageServer.

Bilevel/Grayscale objects remain unchanged unless they are tagged with an ICC profile and the used printer queue has an ICC printer profile.

Note: Do not remove the “ICC-Profiles” volume from the server, because this would disable indirect color matching.

PPD file check

PPD files must be selected carefully to guarantee correct output on the printing device. You can select a PPD file with HELIOS Admin when setting up a printer queue, and you can select a PPD file on your Mac client when choosing a printer. These two PPD files should be identical. The Acrobat plug-in checks this for you and will issue a warning if they are not identical.

Note: The two PPD files are only checked by their name but not by their content. In order to guarantee best printing results ensure that PPD files with the same name do indeed have the same content. PPD files can be stored in the “Settings” volume (`Settings > PPDs`) in order to have them readily at hand.

Colors, color matching, proofing – basic concepts

Please read the chapter “Colors, color matching, proof printing – basic concepts” in the ImageServer manual.

Known restrictions

Page size problems when distilling EPSF files
We recommend using Create PDF Server for PDF generation. The Distiller may have problems recognizing the page size of EPSF input files. See Fig. 15.5 in 15 “Create PDF files using Acrobat Distiller” and the explanation about the `Use Prologue.ps and Epilogue.ps` option.

5 Set up PDF HandShake with HELIOS Admin

5.1 Access to the Admin

PDF HandShake can be easily set up from any of your clients.

In this section we discuss the use of HELIOS Admin to set up high-quality color matching for the PDF files you want to print, and how to specify the parameters for layout generation (if you are also using ImageServer).

Before opening the Admin program you should mount the “ICC-Profiles” volume. This volume is created automatically during installation and contains profiles which can be used. If you have your own profiles and store them in another location, you should move them to the “ICC-Profiles” volume. Inside the top-level folders “Printer”, “Scanner”, and “Monitor” you can create subdirectories at any time Fig. 5.1.

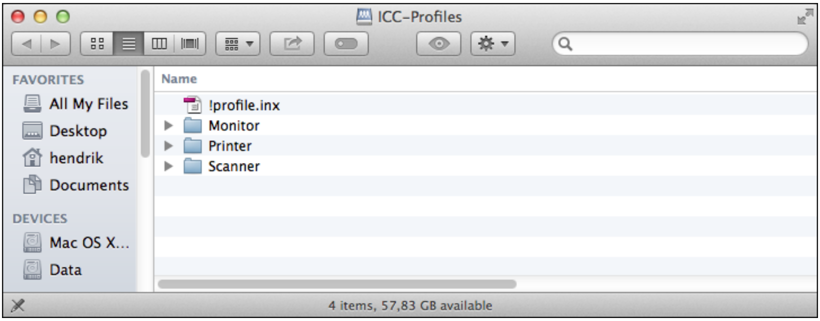


Fig. 5.1: Content of the “ICC-Profiles” volume

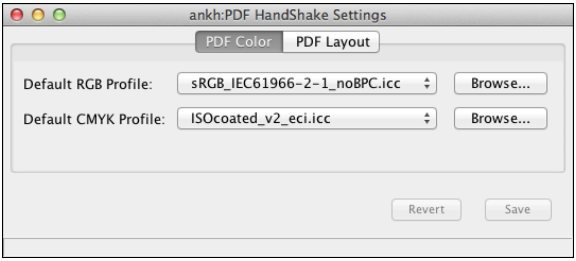
Note: If you have added your own image profiles to the “ICC-Profiles” volume issue the following command to update the ICC index file, in order to notify the OPI server about the changes:

```
# srvutil reconf opisrv
```

5.2 PDF HandShake options

5.2.1 Server-wide PDF settings

➢ First, open the `Settings` menu and select `PDF HandShake Settings` to open the dialog for the PDF HandShake server settings (Fig. 5.2).



Default RGB Profile: sRGB_IEC61966-2-1_noBPC.icc

Default CMYK Profile: ISOcoated_v2_eci.icc

Fig. 5.2: Opening the “PDF HandShake Settings” dialog

PDF Color

The pop-up menus allow you to select an RGB and a CMYK default source profile for your PDF documents. These profiles will be used as input profiles for color data transforms in the case that you print PDF files that have not yet been tagged with profiles.

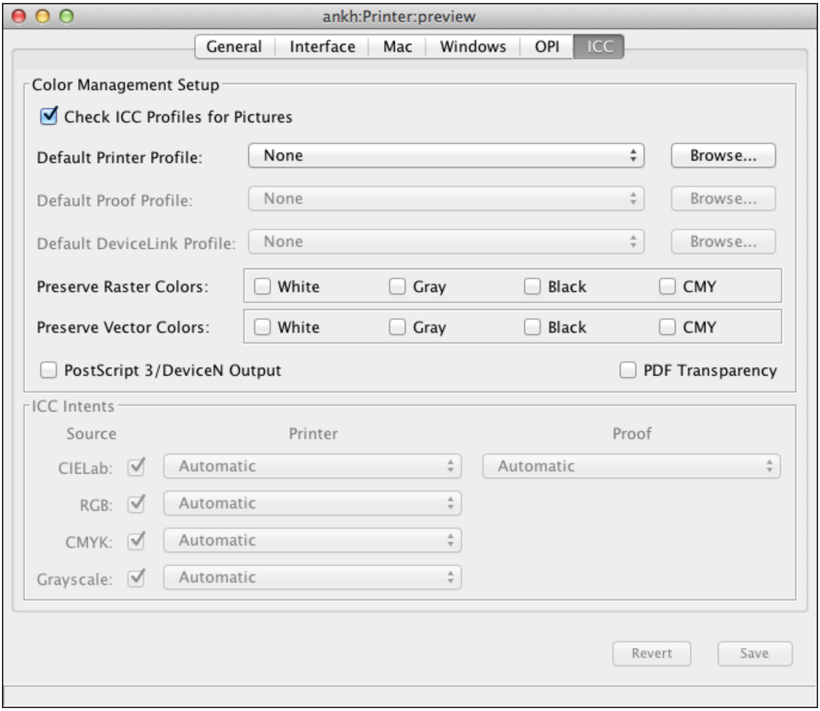
The default profiles shown in Fig. 5.2 are set automatically after product installation. They are both available in the “ICC-Profiles” volume on your server. You can define your own server defaults via the `Browse...` button. American customers, for instance, should switch to a SWOP CMYK profile.

PDF Layout

These options are available when ImageServer is installed (see 5.3 “More options for ImageServer users”). Otherwise they will be grayed out.

5.2.2 Printer queue settings for PDF

➢ To set up printer queue specific parameters, open the desired printer from the `Printers` list and then select the `ICC` tab to open the dialog shown in Fig. 5.3.



Color Management Setup

☑ Check ICC Profiles for Pictures

Default Printer Profile: None

Default Proof Profile: None

Default DeviceLink Profile: None

Preserve Raster Colors: □ White □ Gray □ Black □ CMY

Preserve Vector Colors: □ White □ Gray □ Black □ CMY

□ PostScript 3/DeviceN Output □ PDF Transparency

ICC Intents

Source		Printer	Proof
CIELab:	☑	Automatic	Automatic
RGB:	☑	Automatic	
CMYK:	☑	Automatic	
Grayscale:	☑	Automatic	

Fig. 5.3: Defining ICC settings for a specific printer queue

Check ICC Profiles for Pictures

This option applies to OPI raster image and PDF document replacement.

Default Printer Profile

`Default Printer Profile` lets you choose the correct ICC profile for your output device and, at the same time, serves to switch color matching on for this printer queue. The pop-up menu contains four different options:

❍ `None`
Switches explicit color matching off on the queue. This does not mean that there is no color transformation at all. If you print separations and your PDF files contain e.g. RGB or Lab objects, these objects will be transformed into CMYK "behind the scenes". For these transformations the server uses a default CMYK profile (either `Euro` or `SWOP`, according to the `Default CMYK Profile` setting in the HELIOS Admin `ImageServer Settings`).

❍ `sRGB_IEC61966-2-1`
May be selected if you want to print to an RGB device, e.g. a monitor.

❍ `Lab D65`
This setting is best suited if you want to select a device-independent output color space and leave the separation to the final PostScript RIP. In that case, you have to use a Level 2 (or 3) output device. The RIP will then be initialized with a specific CRD (*Color Rendering Dictionary*) and, using this dictionary, will perform the separation into CMYK. This workflow makes sense if you are sending your documents (e.g. a magazine) to different production sites. Your print job is not yet separated for a given output device and the Lab D65 color space is device-independent. The different RIPs then transform the color data into the gamut of the respective printer/or press and the output results should all be identical.

❍ `Composite - Preserve Colors`
Switches color matching for composite print jobs off on the queue. This means that there will be no color transformation at all.

If you intend to define a default printer profile that is not listed in the pop-up menu, you may select a profile from a repository on your server by clicking the `Browse...` button.

Default Proof Profile

This item becomes only available if you have selected a default printer profile. Here, you can select a profile for your proof printer.

Note: For more details about the color matching of PDF files during printing, you may read the paragraph **Profile information** in 4 “Before getting started”. For background information about color matching and proofs, please read the corresponding chapter in the ImageServer manual.

Default DeviceLink Profile

This pop-up menu allows you to specify a default DeviceLink profile for color matching. DeviceLink profiles have a higher priority than standard ICC profiles, i.e. standard ICC profiles are overridden by DeviceLink profiles, when applicable.

Preserve Raster Colors

Exclude `White`, `Black`, `Gray` or `CMY` colored *raster* objects from color transformation, so that they preserve the selected color(s), even if color matching is done. For example, gray raster image pixels in arbitrary color space are detected and converted to Gray only for CMYK output, if the `Gray` option is checked.

Note: PDF objects that use an Indexed color space are handled by the `Preserve Raster Colors` options only, regardless of whether they are raster-based or vector-based.

Preserve Vector Colors

Exclude `White`, `Black`, `Gray` or `CMY` colored *vector* objects from color transformation, so that they preserve the selected color(s), even if color matching is done. For example, gray text and other gray vector objects in arbitrary color space are detected and converted to Gray only for CMYK output, if the `Gray` option is checked.

Note: PDF objects that use a separation color space or shadings are handled by the `Preserve Vector Colors` options only, regardless of whether they are raster-based or vector-based.

PostScript 3/DeviceN Output

The `PostScript 3/DeviceN Output` checkbox applies only to OPI image replacement. If you want to print a PDF document with output optimized to use the PostScript 3/DeviceN feature, you must print it using the "pdfprint" command line tool with the appropriate option (see 7.1 "pdfprint"):

```
$ pdfprint -o preservedevicen
```

Another possibility is to print via the PDF HandShake Print plug-in for Acrobat, with activated `Preserve DeviceN` checkbox.

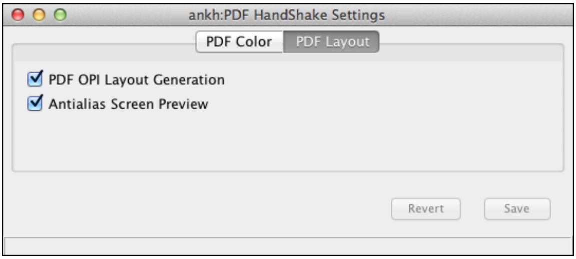
PDF Transparency

If this option is enabled, printing PDF documents with transparencies generates PostScript with "pdfmark" constructs, which is sent to the printer program or device. These "pdfmark" constructs for transparencies can be interpreted by Adobe Distiller 8 (or later) and HELIOS PrintPreview. See also 16 "PDF Transparency".

5.3 More options for ImageServer users

5.3.1 Server-wide settings for PDF (OPI)

With ImageServer installed on your server, you have two more options in the "PDF HandShake Settings" dialog Fig. 5.4.



ankh:PDF HandShake Settings

☑ PDF OPI Layout Generation
☑ Antialias Screen Preview

Fig. 5.4: “PDF HandShake Settings” (ImageServer installed)

`PDF OPI Layout Generation` activates PDF services on the OPI server, meaning that it activates the generation of EPSF layouts for PDF high-resolution files. Please note that the `Create Layouts` option in the “Volume” dialog must be checked as well if you want to enable the generation of layouts (see the ImageServer manual). If `Create Layouts` is checked and `PDF OPI Layout Generation` unchecked, this would mean that the OPI server will automatically generate layouts in this volume – but not for PDF files.

Antialiasing is a method of smoothing contours. This option is only meant for screen previews of EPSF files the OPI server has generated from PDF originals.

5.3.2 Printer queue settings for PDF (OPI)

➢ Highlight a printer in the `Printers` list and then open the `ICC` settings dialog as shown in Fig. 5.3.

All options in the dialog apply to the printing of PDF files, too.

Note: See 5.2.1 “Server-wide PDF settings” above for details about the default profiles.

Please read the respective chapter in your ImageServer manual for detailed information on the printer settings.

The `Check ICC Profiles for Pictures` option will let the OPI server check for every image file or PDF document …

❍ … whether all image profiles that have been tagged to any of the image files are available, *and*

❍ … whether the images and files in the print job are tagged at all

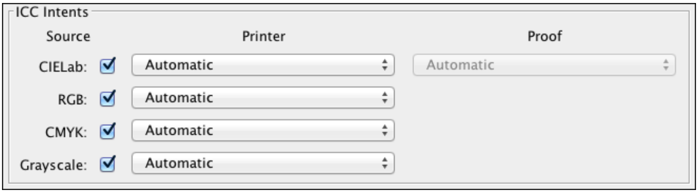
The server will automatically stop the print job if a single profile is missing or if a file is not yet tagged.

5.4 ICC rendering intent settings (HELIOS Admin)

This feature offers different "strategies" for rendering an image file from one color space to another. The following settings are available, which are freely selectable in the HELIOS Admin `ICC Intents` menu (Fig. 5.5):

- `Automatic`
- `Perceptual`
- `Relative Colorimetric`
- `Saturation`
- `Absolute Colorimetric`
- `Perceptual with BPC`[1]
- `Relative Colorimetric with BPC`
- `Saturation with BPC`

[1] Black Point Compensation



ICC Intents

Source		Printer	Proof
CIELab:	☑	Automatic	Automatic
RGB:	☑	Automatic	
CMYK:	☑	Automatic	
Grayscale:	☑	Automatic	

Fig. 5.5: HELIOS Admin “ICC Intents” dialog window

Please make sure that you have assigned a printer profile to the printer queue according to the instructions given in **Default Printer Profile** above. This ensures that the values you may specify in the “ICC Intents” section are used for printing.

Note: Please keep the intent setting on `Automatic` unless you have a special reason to change it. Your profiling software will explain the intents meanings. Reactivation of color matching for selected input color spaces by unchecking the source color space checkboxes is not supported in PDF printing.

5.4.1 Spot color editor

HELIOS Admin features a spot color editor which enables the user to define a preview color for a given spot color name. This allows PDF HandShake to include spot color objects in the composite preview, e.g. when working with pre-separated PDF documents. The spot color editor is described in the HELIOS ImageServer manual.

6 PDF HandShake utility programs

6.1 pdfcat

“pdfcat” is a command line tool that allows you to explode or concatenate PDF files on the server. Using “pdfcat” is indispensable if you have a multi-page document that is to be used in an EPSF-only or in an OPI workflow. The OPI server, for instance, will generate a layout file of the first document page only – so if you want to have a layout of document page 2, you must extract the desired page and create a new single-page PDF file. The illustration below shows the three different program modes that are all independent of one another and do exclude each other: The “concatenate” mode merges the selected PDF files into a new one, the “append” mode appends the selected files to an existing one, and the “explode” mode writes the selected pages of an existing document into new single-page files.

Please note that . . .

Every PDF file contains a list of file infos such as creator, creation date, modification date, and image profiles (optional). Furthermore, a PDF file may contain security settings (optional), a table of contents (TOC) and annotations such as text fields, buttons, etc.

The “pdfcat” program re-arranges PDF files and creates new ones. The file infos, profiles, security settings, TOC, and annotations are handled as follows:

pdfcat -o

If “pdfcat” creates a new multi-page document – as shown in the first row in the illustration – the new file will have its own creator and creation date. It will not have any ICC profiles, even if the input files were tagged, and it will not have any security settings. Tables of content – if there were any in one of the input files – will not be copied to the new file. However, annotations from the original files will find their way into the newly created document.

pdfcat -a

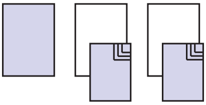
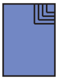
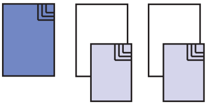
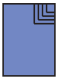
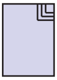
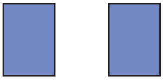
If “pdfcat” appends pages and/or documents to an existing PDF file – as shown in the second row in the illustration – the output file (*helios.pdf (new)*) will contain all the information that was already included in the original file (*helios.pdf*). File infos, profiles, security settings, TOC, and annotations remain unchanged. Information that had been included in the appended files (*Doc2* and *Doc3*) will – except for the annotations – not make it into the output file; they will be ignored.

pdfcat -e

If “pdfcat” explodes a document into several single-page documents – as shown in the last row in the illustration – the new documents inherit the file infos, profiles, and security settings from the original file. Tables of content, however, will not be copied to the output files. Please note that “pdfcat” cannot write comments into the Finder Info. Usually, if a PDF file contains profile information, the profiles are listed in the Mac OS 9 Finder Info, or via HELIOS Meta, or in WebShare. The text field may be empty for PDF files you have created with “pdfcat explode”, even though the new files contain profile information. In that case, you can use our Acrobat print plug-in or the ImageServer “HELIOS ICC Tagger” application to make the profiles visible.

For OPI users only

Automatic layout generation is not available for PDF files that have been created with “pdfcat”. You need to use our “opitouch” or “layout” programs to generate layouts from the new PDF files, or enter the `dt touch -E` command on a shell. Alternatively, the procedure can be automated by means of the ImageServer Script Server.

Action	In	Out
Concatenate	 Doc1.pdf + Doc2.pdf pages x-y + Doc3.pdf pages x-y	 = outDoc.pdf
Append	 helios.pdf + Doc2.pdf pages x-y + Doc3.pdf pages x-y	 = helios.pdf (new)
Explode	 leaflet.pdf	 = leaflet-page1.pdf + leaflet-page2.pdf

Usage:

```
pdfcat [-v] -o <outDoc> <inDoc> ...
pdfcat [-v] -a <outDoc> <inDoc> ...
pdfcat [-v] -e <prefix> <inDoc>
pdfcat -h
```

6.1.1 Options

-v This option allows you monitor the “pdfcat” conversion progress.

-o Concatenates existing PDF documents (`inDocs...`) to a new PDF document (`outDoc`). With this parameter, you have to specify one output file name and one or more input file names. It is possible to copy only selected pages from the input file to the output file by specifying page ranges. Valid ranges are listed in 6.1.2 “Operands”.

Important: If the name you choose for the output file already exists in the destination directory, the existing file will be replaced!

-a Appends one or more PDF documents or parts of them (`inDocs`...) to another – already existing – PDF document (`outDoc`).

-e Extracts from a given PDF document (`inDoc`) either all pages or the pages you select explicitly, and creates new single-page PDF files. With this parameter, you have to specify a prefix for the output files. Page number `<nnn>` of the input file will then be copied to a new PDF document named `<prefix><nnn>.pdf`.

-h Display help file.

6.1.2 Operands

`<inDoc>` is the path name of a PDF file, optionally followed by a comma-separated list of page ranges. Valid page ranges are:

<a> Page `<a>` only

<a>-<b> Page `<a>` to page `<b>`

<a>- Page `<a>` to last page

-<b> First page to page `<b>`

The character “$” stands for the last page of a document (the “$” must be escaped on a shell).

If `<inDoc>` is followed by a list of page ranges, only the specified pages of `<inDoc>` will be copied to the destination file. The default is to copy all pages from `<inDoc>`.

Example 1:

```
$ pdfcat -o new.pdf doc1.pdf doc2.pdf,2,5-7
```

Writes all pages of document “doc1.pdf” and the pages 2, 5, 6, 7 of document “doc2.pdf” to the new document “new.pdf”.

Example 2:

```
$ pdfcat -o new.pdf doc1.pdf,\$-1
```

Writes all pages of document “doc1.pdf” in reverse order to a new document called “new.pdf”.

Example 3:

```
$ pdfcat -a tmp.pdf doc1.pdf,9-6
```

Appends the pages 9, 8, 7, 6 (in this order) of document “doc1.pdf” to an existing document called “tmp.pdf”.

Example 4:

```
$ pdfcat -e page doc1.pdf,-3
```

Writes the pages 1, 2, 3 of document “doc1.pdf” to new single-page documents called “page1.pdf”, “page2.pdf”, and “page3.pdf”.

6.2 pdfform

“pdfform” allows you to output form field values of a PDF document.

Usage:

```
pdfform [-p <password>] <pdffile>
```

6.2.1 Option

-p <password>

Password to open PDF document. Required if `<pdffile>` is secured by an “Open Password”.

Example:

```
$ pdfform /laura/base11_e.pdf
  Company=HELIOS Software GmbH
  Date=03-11-04
  Priority=Urgent
```

This PDF document contains three form fields of the type text. The *Company* field has the value “HELIOS Software GmbH”, the *Date* field “03-11-04”, and the *Priority* field has the value “Urgent”.

6.3 pdfinfo

“pdfinfo” has two modes of operation: either print information about the PDF document `<pdffile>` or extract objects from it.

In information mode every output line is of the form:

```
section: key1=value1, key2=value2, ..., flag1, flag2, ...
```

Possible sections are `General`, `Security`, `Profile`, `Layer`, `Plate`, `OutputIntent`, `Color`, `Font`, `PageLabel`, `MediaBox`, `CropBox`, `BleedBox`, `TrimBox`, `ArtBox`, `Rotate`, `Transparency`, `Pattern`, `Image`, and `Form`.

The output can be restricted with the `-o` option. Note that PDF text strings such as title and author information are converted to UTF-8 encoding.

In object extraction mode arbitrary objects with object number can be extracted from the PDF document `pdffile` to standard output.

Note: The object extraction mode is only meaningful for experts with knowledge of the internals of PDF documents.

Usage:

```
pdfinfo [-f <fromPage>][-t <toPage>][-o <sections>][-l <layer>]
        [-m][-v][-p <password>] <pdffile>
```

```
pdfinfo [-x <objNo> | -s <objNo> | -d <objNo>][-i <intentNo>]
        [-p <password>] <pdffile>
pdfinfo -h
```

6.3.1 Options

-p

Password to open a PDF document. This option is required if `<pdffile>` is secured by an “Open Password”.

Information mode:

-f <from page>

First page number for font and color information. The default is 1.

-t <to page>

Last page number for font and color information. The default is the last page.

-o <sections>

Print only information for the specified sections. The default value is `-o All`, which prints all available information.

-l <layer>

Select or deselect layers (see 17 “PDF layers”).

-m

Use a different output format for values and flags, which is suitable for post-processing.

-v

Display object numbers of images and forms.

Object extraction mode:

-x <objNo>

Extract object with number `<objNo>`.

-s <objNo>

Extract a stream without decompressing it from object with number `<objNo>`.

-d <objNo>

Extract and decompress a stream from object with number `<objNo>`.

-i <intentNo>

Extract ICC profile of PDF/X output intent with number `<intentNo>`. PDF documents usually have one or no PDF/X output intent, so this number is usually 1.

Example 1:

The following commands are equivalent:

```
$ pdfinfo Doc.pdf
$ pdfinfo -o All Doc.pdf
```

Example 2:

Getting information about all images in "Doc.pdf":

```
$ pdfinfo -o Image Doc.pdf
# pdfinfo 5.0.0
Image: Page=18, BBox=207.6720/371.4596/
264.3224/443.6887, Resolution=278.3387/593.1125, BPP=8,
ColorSpace=DeviceGray, Filters=FlateDecode
Image: Page=18, BBox=225.1510/387.2033/
237.4971/397.1024, Resolution=194.9898/194.9960, BPP=8,
ColorSpace=DeviceRGB, Filters=FlateDecode
```

Example 3:

Getting information about transparencies in "Doc1.pdf":

```
$ pdfinfo -o Transparency Doc1.pdf
# pdfinfo 5.0.0
Transparency: Page=1, Transparencies=no
Transparency: Page=2, Transparencies=yes
Transparency: Page=3, Transparencies=no
Transparency: Page=ALL, Transparencies=yes
```

Example 4:

Getting information about tiling patterns in “Doc2.pdf”:

```
$ pdfinfo -o Pattern Doc2.pdf
# pdfinfo 5.0.0
Pattern: Page=1, Patterns=yes
Pattern: Page=2, Patterns=no
Pattern: Page=ALL, Patterns=yes
```

Tiling patterns are displayed as gray areas if a PDF document is converted with ImageServer or printed to a Print Preview queue, see 2.4 “Known limitations”.

6.4 pdfnote

“pdfnote” has three modes of operation: either list all annotations of a PDF document `<pdffile>` or extract the contents entry of one annotation in a PDF document or add a text annotation to a PDF document.

Usage:

```
pdfnote -l [-p <password>] <pdffile>
pdfnote -x <annot> [-n <page>][-p <password>] <pdffile>
pdfnote [-n <page>][-r <location>][-s][-t <title>][-c <contents>]
        [-p <password>] <pdffile>
pdfnote -h
```

6.4.1 Options

-p <password>

Password to open PDF document. Required if `<pdffile>` is secured by an “Open Password”.

-n <page>

Number of the page where to add the text annotation. The default is `1` (first page).

-l

List all annotations of the PDF document.

-x <annot>

Extract contents entry of annotation `<annot>` on selected PDF document page.

-r <location>

The location of the text annotation on the page. `<location>` is of the form: `<llx>:<lly>:<width>:<height>` specifying the lower left x coordinate, lower left y coordinate, width, and height of the text annotation rectangle in points.

-s

Specifies that the annotation should initially be displayed open.

-t <title>

Title string. `<title>` will appear in the annotations title bar.

-c <contents>

Contents of the text annotation. If `-c` is not specified, the contents are read from “stdin”.

-h

Help

Example:

```
$ pdfnote -r 10:400 -t "Joe" -c "OK" Doc1.pdf
$ pdfnote -r 10:10:200:200 -t "Info" Doc2.pdf < info.txt
```

6.5 pdftoeps

EPSF files can be generated by different applications, e.g. Illustrator or Photoshop, and they can be different in structure. One purpose of “pdftoeps” is to allow you to stick to PDF as a file exchange format and to use the program to generate EPSF files, that are all homogeneous in structure, from the incoming PDF files. This tool is available for ImageServer users only.

If you have an EPSF-only workflow, but receive PDF files from your customers, you may use the “pdftoeps” tool to transform PDF files into EPSF files. It allows you to specify the color space, the resolution, and the type (Mac-EPSF, PC-EPSF) of the output files. An example of the “pdftoeps” tool is given in Fig. 6.1. The illustration shows a situation where “pdfcat” and “pdftoeps” are used to convert a multi-page PDF document into several single-page EPSF files.

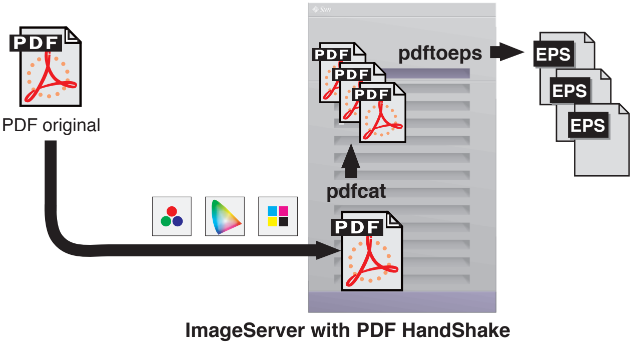


Fig. 6.1: Creating EPSF files using the “pdftoeps” tool

Font quality

We include in our software package 131 original PostScript 3 fonts. They are available automatically after installation and appropriate activation of PDF HandShake on the server. They guarantee high-quality printing, and high-quality font representation on screen and in layout files (for ImageServer). Our fonts are listed in B “The fonts we deliver”.

Usage:

```
pdftoeps [options] <pdffiles> ... <destination>
```

6.5.1 Options

-v

Displays activity reports during PDF to EPSF transformation.

-P <password>

Password to open PDF document. Required if `<pdffile>` is secured by an “Open Password”.

-m

Produces Mac EPSF files (see C “Glossary” and the option `-p` below).

-p

Generates cross-platform EPSF files (opposite of option `-m` above). If neither of the two options is specified, the default depends on the location of the selected PDF files and on whether this location is an EtherShare volume. If the volume settings in the respective volume are set to cross-platform EPSF, the resulting files will be cross-platform.

-r <resolution>

Sets resolution in dpi used for the printable preview of EPSF files. This parameter requires a floating point value as e.g. 72.0. If you do not specify `-r` the resolutions of the elements of the PDF input file will be used.

-c <colorspace>

Defines color space used for the printable part of the EPSF file. The parameter requires a string value as e.g. “CMYK”. For valid strings, see Table 6.1 below. If you do not specify `-c` the default from the OPI server will be used.

-R <resolution>

Sets resolution in dpi used for the screen preview of EPSF files. This parameter requires a floating point value as e.g. 72.0. If you do not specify `-R` the default from the OPI server will be used.

-C <colorspace>

Defines color space used for the screen preview of the EPSF file. The parameter requires a string value as e.g. "Grayscale". For valid strings, see Table 6.1 below. If you do not specify the `-c` option the default color space (RGB) will be used.

Name of color space		
None	Grayscale	Indexed
RGB	CMYK	CIELab

Table 6.1: List of supported EPSF color spaces

-b <pagebox>

Page box options: `MediaBox`, `CropBox`, `BleedBox`, `TrimBox` or `ArtBox`. Default is "CropBox".

-B

Produces EPS raster layout images.

-h

Display help file.

All options of the "pdftoeps" tool are optional. However, you should specify parameters like `-m` or `-p` explicitly whenever you are not sure about the defaults that are currently valid.

After the options, you have to specify one or more files to convert and a destination file (if converting a single file) or a destination directory (if converting several files). The destination can contain a complete UNIX path name.

Example:

```
   $ pdftoeps -m -c CMYK -C RGB file1 file2 /user/tmp
```

Pre-separated PDF documents

The “pdftoeps” program recognizes pre-separated PDF documents. It generates DCS files with default (depending on server settings) composite previews which are raster based with a maximum resolution of 150 dpi.

The DCS files are DCS-1 or DCS-2 style multifile images. The plate file suffixes for CMYK will be *.C*, *.M*, *.Y*, and *.K*. Spot color plate files will be assigned other suffixes, namely letters so far unused in alphabetical order. The suffix does not have any relation to the name of the spot color.

6.6 pdfprint

“pdfprint” allows the printing of PDF files directly from the server to a HELIOS printer queue (Fig. 6.2). Features like color matching and proof printing are available for each queue. Pre-separated files cannot be printed composite unless ImageServer is also used. For options and usage information, see 7.1 “pdfprint”.

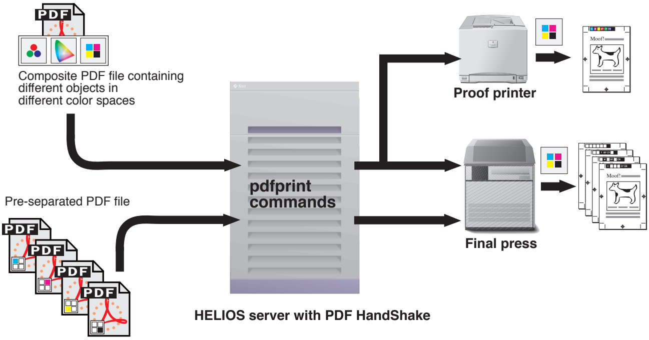

PDF
Composite PDF file containing different objects in different color spaces
Pre-separated PDF file
PDF
PDF
PDF
PDF
pdfprint commands
HELIOS server with PDF HandShake
Proof printer
Final press
Moof!

Fig. 6.2: Printing PDF files directly from the server

PDF print plug-in for Acrobat

The print plug-in for Acrobat, (see 7.2 "Print PDF files using the Acrobat plug-in") is available for Mac computers only. It is the equivalent of the "pdfprint" program and also prints to PDF HandShake printer queues (Fig. 6.3). Again, without ImageServer, pre-separated files can only be printed as separations.

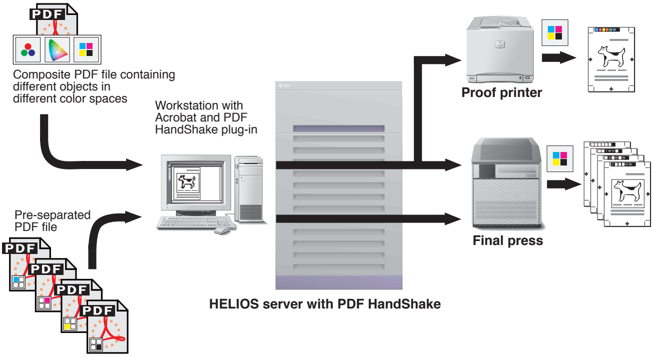

PDF
Composite PDF file containing different objects in different color spaces
Workstation with Acrobat and PDF HandShake plug-in
Pre-separated PDF file
PDF
PDF
PDF
PDF
HELIOS server with PDF HandShake
Proof printer
Final press

Fig. 6.3: Printing PDF files with the Acrobat plug-in

PDF extension for ImageServer

With PDF HandShake and ImageServer, you can use the PDF file format as an input format for the ImageServer layout generation. Moreover, you can use the "PDF HandShake" Acrobat plug-in (or the "pdfprint" command line tool) to export PDF files for further use in an imposition program. Fig. 6.4 shows these two features at a glance.

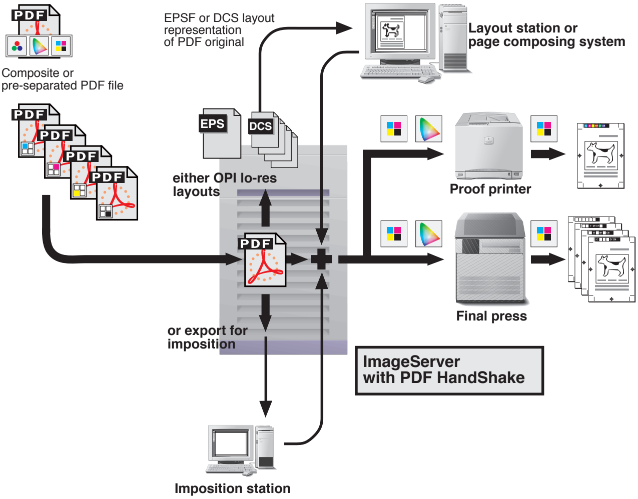

PDF
Composite or pre-separated PDF file
EPSF or DCS layout representation of PDF original
Layout station or page composing system
EPS
DCS
either OPI lo-res layouts
Proof printer
Final press
or export for imposition
ImageServer with PDF HandShake
Imposition station

Fig. 6.4: The options for PDF files on an ImageServer

The OPI server uses the first page of a PDF file for layout generation. Further pages are ignored. The file format of the layout representation is EPSF for composite PDF files and DCS for pre-separated PDF files. This allows you to place the layouts into any popular layout application, e.g. QuarkXPress or InDesign, and into any editorial or page composing system.

6.7 pdfresolve

The “pdfresolve” program allows resolving OPI references in PDF documents. Layout applications like QuarkXPress[1] or InDesign can export their native documents as PDF documents. If the native documents contain OPI references, these can be preserved during the export process, to be used to replace gray-area placeholders with the corresponding high-resolution original images prior to or during the printing process.

OPI 1.3 references contain the path name of the placed image file plus an optional Mac file ID. During OPI reference replacement “pdfresolve” searches for a matching high-resolution or low-resolution image to replace the form. An image search consists of a sequence of search methods: search by path name, search in additional search paths, search by Mac file ID, search in additional search volumes. An image search is complete when a search method succeeds. If a low-resolution image is searched, the found image is used for replacement, regardless whether it is low-resolution or high-resolution. If a high-resolution image is searched and a high-resolution image is found, it is used for replacement. If a high-resolution image is searched and a low-resolution image is found, the OPI reference of the low-resolution image is used for another high-resolution image search. A chain of low-resolution images leading to a high-resolution image must not exceed the length of seven.

Another advantage of “pdfresolve” is that transparencies that have been applied to the layout images in the layout application are preserved during the OPI image replacement process.

[1] no OPI support beginning with QuarkXPress 10

Note: OPI references in PDF documents are always embedded in form streams with OPI entries, which are called *OPI forms* in short. “pdfresolve” replaces OPI 1.3 forms in composite PDF documents, pre-separated PDF documents are not supported. Referenced files must be raster images or PDF documents. Object-based EPSF files are not supported. The maximum number of open files for a process roughly limits the number of referenced PDF documents. In PDF documents with nested OPI references only the top level is resolved.
For OPI references to PDF files, only references to the first page of the document are supported.

A description of how to set up hot folder mechanisms for the PDF native OPI workflow can be found in 11 “PDF-native OPI workflow”.

“pdfresolve” is available for ImageServer users only.

Usage:

```
pdfresolve [-lv] -P <printer> [-g <logFile>][-o <key>=<value>]...
           <inDoc> <outDoc>
pdfresolve -h
pdfresolve -P <printer> -h (help info with printer parameters)
```

6.7.1 Options

-l

High-resolution images are inserted by default. When this option is set, low-resolution images are inserted instead if they are available.

-v

Verbose mode. The file names of inserted images are printed.

-P <printer>

Replace OPI objects using preference settings of printer queue `<printer>`. OPI must be active on this printer queue. This option is mandatory.

-g <logFile>

If there are warnings or errors, then generate a log file `<logFile>` with their description.

-o <key>=<value>

Set parameter with key `<key>` to value `<value>`.

-h

Display help file.

-P <printer> -h

Help info with printer parameters.

Note: Command line options have a higher priority than file specific preference settings, and file specific preference settings have a higher priority than global preference settings which include printer queue preference settings. Command line option keys are case-insensitive, but preference keys are case-sensitive.

6.7.2 Valid parameter keys

ImageSearchPaths <strlist:"">

The list of additional search paths. The search in additional search paths locates files with matching basenames in these search paths.

ImageIDsearch <bool:TRUE>

Determines whether images are searched via Mac file ID. The search via file ID first extracts the volume specification from the file ID and then searches a file with matching file ID in the desktop database of the specified volume.

ImageSearchVolumes <strlist:"">

The list of additional search volumes. The search in additional search volumes uses their desktop databases to locate files with matching basenames, but excludes files in the network trash folder and in layout folders. This search succeeds if and only if there is exactly one matching file.

CheckImages <bool:TRUE>

If this option is `TRUE`, errors are generated for missing referenced images. If this option is `FALSE`, warnings are generated for missing referenced images and the OPI form is left unchanged.

CheckRaster <bool:TRUE>

If this option is `TRUE` and referenced image files do not contain raster data with print quality, errors are generated. If this option is `FALSE` and referenced image files do not contain raster data with print quality, neither errors nor warnings are generated.
For processing PDF documents which were assembled using object-based EPSF files you need to set this option to `FALSE`.

RemoveOPI <bool:FALSE>

Determines whether OPI references of replaced raster images are removed. The OPI references of replaced PDF documents are always removed.

ProfileRepository <str:"ICC-Profiles">

Name of the ICC profile repository volume. If an ICC profile is searched, it is searched here first, then it is searched in the directories listed in the `ProfileSearchPaths` parameter.

ProfileSearchPaths <strlist:"">

The list of additional directories where ICC profiles are searched.

CompositeColorspace <str:"CMYK">

Determines the output color space of source images with color spaces other than bilevel and grayscale. Grayscale images are affected if (and only if) they are colorized and have an ICC profile. The value of this option must be one of `None`, `Grayscale`, `RGB`, `CMYK` or `CIELab` and must be consistent with the value of the `DefaultPrinterProfile` parameter, if specified.

DefaultPrinterProfile <str:"">

Determines printer ICC profile for all source images with color spaces other than bilevel and grayscale. Grayscale images are affected if (and only if) they are colorized and have an ICC profile.

PrintRenderingIntents

Determines rendering intent used for conversion between any source and printer color space. Its syntax is:

```
<sourceA>:<printerA>:<intentA>,<sourceB>:<printerB>:<intentB>
```

	To:	**N**	**S**	**B**	**G**	**I**	**R**	**H**	**H**	**C**	**C**	**M**	**D**	**Y**	**C**	**C**	**C**	**C**	**Y**
From:	None	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	Spot	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	Bilevel	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	Grayscale	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	Index	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	RGB	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	HSV	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	HLS	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	CMY	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CMYK	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Multitone	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	Duotone	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	YCbCr	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	CIELab	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	CIEXYZ	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	CIELuv	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	CIEYxy	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3
	YCC	3	3	3	3	3	3	3	3	0	0	3	3	3	3	3	3	3	3

Valid rendering intent numbers are:

- Perceptual (0)
- Relative Colorimetric (1)
- Saturation (2)

- Absolute Colorimetric (3)
- Perceptual with BPC (4)
- Relative Colorimetric with BPC (5)
- Saturation with BPC (6)

DefaultProofProfile <str:"">

Determines proof ICC profile for all source images with color spaces other than bilevel and grayscale. Grayscale images are affected if and only if they have an ICC profile and use default OPI colorization.

ProofRenderingIntents

Determines rendering intent used for conversion between any printer and proof color space. Its syntax is:

```
<printerA>:<proofA>:<intentA>,<printerB>:<proofB>,<intentB>
```

	To:	N	S	B	G	I	R	H	H	C	C	M	D	Y	C	C	C	C	Y
From:	None	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Spot	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Bilevel	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Grayscale	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Index	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	RGB	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	HSV	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	HLS	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CMY	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CMYK	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Multitone	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	Duotone	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	YCbCr	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CIELab	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CIEXYZ	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CIELuv	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	CIEYxy	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	YCC	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3

See valid rendering intent numbers above (`PrintRenderingIntents`).

DefaultDevLinkProfile <str:"">

If `DefaultPrinterProfile` is not set, this option has no effect. This option specifies a DeviceLink ICC profile whose output color space must match the color space of `DefaultPrinterProfile`. If both `DefaultPrinterProfile` and `DefaultDevLinkProfile` are set, all source images whose color space matches the input color space of the DeviceLink ICC profile are matched using the DeviceLink ICC profile.

RenderingQuality <int:2>

Determines the CMM rendering quality. Its value must be one of `0` (*normal*), `1` (*draft*) or `2` (*best*).

CheckICCProfiles <bool:TRUE>

Determines whether to generate errors for missing ICC profiles.

IgnoreUntagged <bool:FALSE>

If `DefaultPrinterProfile` is not set or `CheckICCProfiles` is not active, this option has no effect. Otherwise this option determines whether errors are generated for color source images which are not tagged with an ICC profile and which are not excluded from color matching via `DefaultPrinterProfile`.

PureBlack <bool:FALSE>

Determines whether black pixels are excluded from color matching.

PureWhite <bool:FALSE>

Determines whether white pixels are excluded from color matching.

PureGrays <bool:FALSE>

Determines whether gray pixels are excluded from color matching.

PureCMY <bool:FALSE>

This option determines whether pure Cyan, pure Magenta, and pure Yellow pixels are excluded from color matching in CMYK to CMYK conversions. Pure Cyan pixels have 100 percent Cyan and 0 percent for other colors. Pure Magenta pixels and pure Yellow pixels are defined similarly.

PDFPageBox <crop box>

Determines the page box used in replacement of PDF high-resolution images. Page box options: `MediaBox`, `CropBox`, `BleedBox`, `TrimBox`, and `ArtBox`.

TagReplacedImages <bool:FALSE>

Determines whether to tag replaced images with an ICC profile where applicable. When this option is active and a printer profile is set, a replaced color image is tagged with the printer or proof profile. When this option is active and no printer profile is set, a replaced color image is tagged with the profile of its source image if no color conversion is necessary.

IgnoreSpots <bool:FALSE>

Determines whether to ignore all spot colors of source images. If spot colors are not ignored, they are converted to process colors unless the source image has color space CMYK with additional spot color channels and `PreserveDeviceN` is active.

CustomColorTinting <bool:TRUE>

Determines whether to generate spot colors for colorized bilevel and grayscale images if the colorization does not require the DeviceN color space. To preserve spot colors when the DeviceN color space is required, both the `CustomColorTinting` and `PreserveDeviceN` options must be active.

PreserveDeviceN <bool:FALSE>

Determines whether DeviceN images are generated for source images with color space CMYK and additional spot color channels. If DeviceN images are generated, color matching is disabled.

ColorAliases <strlist:"">

The list of color name aliases. Color name substitution is applied to process and spot color names from source images and OPI comments. Its syntax is: `<nameA>=<aliasA>,<nameB>=<aliasB>,...`

DownSampling <bool:FALSE>

Determines whether downsampling is active. Bilevel images are excluded from downsampling.

FixedSampling <bool:FALSE>

If downsampling is not active, this option has no effect. If it is active, this option determines whether upsampling is active.

FastDownSampling <bool:FALSE>

Determines whether the nearest neighbor algorithm or the mean value algorithm is used for downsampling. The nearest neighbor algorithm is fast and has low quality, the mean value algorithm is slow and has high quality.

Resolution <double:0.000000>

Determines the resolution for downsampling in dpi. The specified values must be positive.

ICMethodBilevel, ICMethodGrayscale, ICMethodRGB, ICMethodCMYK, ICMethodCIELab, ICMethodOther <str:"None">

These parameters determine the compression method for images of the corresponding output color space. Their value must be one of `None`, `Compress`, `CCITTG4`, `JPEG`, `JPEG 2000` or `Flate`.

ICQualityBilevel, ICQualityGrayscale, ICQualityRGB, ICQualityCMYK, ICQualityCIELab, ICQualityOther <see text>

These parameters determine the quality of the JPEG and JPEG 2000 compression for images of the corresponding output color space. Their type is `double` between `0` and `100`.

For JPEG:

<double:75>

Creates poor…high quality JPEG image (1…100).

For JPEG 2000:

<double:0>

Specifies the image quality of a JPEG 2000 image in relation to the uncompressed high-resolution original. The values range from 1…100. Specifying “0” means *lossless*.

Note: For a description of how to specify downsampling and output compression features in a printer queue see 11.3.1 “Downsampling and output compression”.

6.7.3 OPI settings for “pdfresolve” via preference

When setting up a dedicated printer queue for the “pdfresolve” workflow (i.e. not printing PostScript) JPEG 2000 may be used. However, for this it is required to set the corresponding preferences manually, e.g.:

```
# prefvalue -k "Printers/PRINTER/PostScriptImage/CIELab/CompressPostScript"
               -t str "JPEG 2000"

# prefvalue -k "Printers/PRINTER/PostScriptImage/CMYK/CompressPostScript"
               -t str "JPEG 2000"

# prefvalue -k "Printers/PRINTER/PostScriptImage/Grayscale/
               CompressPostScript" -t str "JPEG 2000"

# prefvalue -k "Printers/PRINTER/PostScriptImage/RGB/CompressPostScript"
               -t str "JPEG 2000"
```

The `CompressPostScript` preference is described in the ImageServer manual in the chapter “OPI-related printer queue preferences”.

7 Printing with PDF HandShake

7.1 pdfprint

The “pdfprint” program is a command line tool that allows printing PDF files directly from the server without opening an application. The Acrobat software is not required.

7.1.1 Program behavior and defaults

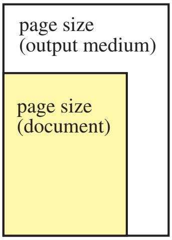
The “pdfprint” program comes with a number of parameters. You may specify, for example, the number of copies, the pages you want to print, and whether you want to print composite or separations. The parameters that are available with “pdfprint” are similar to those that are offered by the printer driver on a Mac computer.

Before describing the parameters in detail, we would like to give you some information about the program behavior and its defaults:

Please note that …

… the “pdfprint” command can only deal with one file at a time. If you want to have more than one PDF file in your print job you may use our “pdfcat” tool and write several PDF files into one (see 6.1 “pdfcat”). Of course, if ImageServer is installed, the HELIOS Script Server “printpdf” script enables multiple PDF files to be automatically queued and printed via a hot folder.

… “pdfprint” sends the print job to a HELIOS PostScript printer queue. Thus, you can make use of all the benefits offered by the HELIOS print server. The parameters you set are applied to the file before the job is sent to the queue. This is useful because errors, e.g. missing fonts, can be detected before the job is spooled.



There are two different page sizes, namely the page size of the document and the one of the output medium (paper size). By default, “pdfprint” adjusts the bottom left corner of the document page to the bottom left corner of the output medium.

“pdfprint” prints the pages you have specified according to the order you have specified; the program prints the separations for every page, and for each separation it prints the number of copies you have defined. If you print e.g. separations, and want to have two copies of pages 1 to 2, the order of printouts will be as follows:

```
1       page 1    Cyan      copy 1
2       page 1    Cyan      copy 2
3       page 1    Magenta   copy 1
4       page 1    Magenta   copy 2
5       page 1    Yellow    copy 1
6       page 1    Yellow    copy 2
7       page 1    Black     copy 1
8       page 1    Black     copy 2
9       page 2    Cyan      copy 1
10      page 2    Cyan      copy 2
```

By default, all dimensions, e.g. page size, are expected to be given in inches (`2.5` means 2.5 *inches*). You can switch to centimeters by specifying the unit explicitly, e.g. `2.5cm`.

Some print options are not available when printing pre-separated files. If you specify the respective parameters, they will be silently ignored. See 4 “Before getting started” for more details.

Usage:

```
   pdfprint [options] <pdfFilename> [-|<psFilename>]
```

7.1.2 Options

- (at the end)

Direct job to “stdout”. The optional output parameter can be either `-` for “stdout”, or, can be the output file name `<psFilename>`.

Note: You can specify several options at a time. Each option must be preceded by the `-o` option, e.g.: `-o orientation=landscape -o negativeprint`

Printer, paper size, paper tray and resolution options:

-P

Printer name is meant to direct your print job to a specific HELIOS printer queue. This parameter must be followed by the queue’s name. If you do not specify it, “pdfprint” will print to “stdout”. If you specify a queue, but add a minus (-) at the end of the command line, “pdfprint” will use the features of the selected printer, but nevertheless send the job to “stdout”.

-p

Media or Page size is used to specify a certain output medium or its size respectively. If you do not specify this parameter “pdfprint” uses the default medium of the selected printer – unless you have specified the `-I` option (see the parameters `-P`, and `-I`). The media or page size can be specified in four different ways:

❍ `Width:Height` (e.g. `14.8:21cm`). If you skip the unit, it is expected to be inches. These exact values are valid no matter whether you did specify a printer (using the `-P` parameter) or not. In case that you did specify a printer, these values together with the `-f` option can for example be used to scale your document independent of the printer’s medium size. For example, if your printer uses A4 sheets, you can scale an A4 document down to A5 by specifying `-f`, followed by `-p 14.8:21cm`.

❍ Using one of the program's medium names. The "pdfprint" program contains a list of medium names with width and height being pre-defined. You can use the following names – instead of specifying exact values as described above. Note that all these names are case-sensitive:

- `letter` (21.59 x 27.94 cm)
- `lettersmall` (21.59 x 27.94 cm)
- `legal` (21.59 x 35.56 cm)
- `b5` (17.6 x 25 cm)
- `a5` (14.8 x 21 cm)
- `a4` (21 x 29.7 cm)
- `a4small` (21 x 29.7 cm)
- `a3` (29.7 x 42 cm)
- `pdf` (PDF Page Size)

❍ If you specify an arbitrary medium name – one that is not specified in the above list, e.g. "Tabloid" or "A4" – you have to make sure that you did specify a printer as well (using the `-P` parameter). "pdfprint" will then check with the current PPD file whether the medium name is valid for one of the paper trays. If not, this can produce a PostScript error – depending on the printer's default behavior.

❍ You may specify `pdf` to make each PostScript page exactly as large as specified in the PDF page, e.g.:

```
$ pdfprint -P myPrinter -p pdf myPDF.pdf
```

Note: You can use the `-P` parameter together with the `-h` option to display the general usage of "pdfprint" and all medium names that are valid for the current printer (see example below):

```
$ pdfprint -P lw -h
...
The printer 'lw' supports the following paper sizes:
*Letter
Legal
A4
B5
```

```
LetterSmall
LegalSmall
A4Small
Com10
Monarc
```

The “*” marks the PPD’s default paper size. The default size will be used automatically if you do not specify the `-p` option *and* if you do not specify `-l` either (see the `-l` option below). Please note that in specific situations, this can cause problems: For example, if `*Letter` is the default paper size in the PPD file, but the printer only has an A4 paper tray, the job will be aborted. In such a situation you will have to specify `-p A4` on the command line explicitly.

As already indicated above, there is an interdependency between the options `-p` and `-l`. In case that you specify neither of them, the defaults from the current PPD file will be valid. If you specify only one of them, the paper size *and* paper tray entry from the PPD file will both be ignored – meaning that if you specify e.g. `B5` as paper size, the corresponding paper tray (e.g. `OptionalCassette`) will be selected automatically.

-l

Paper tray can be used to select a specific paper tray for the printer. The string you enter must be valid for the current printer. Use the `-P` parameter together with the `-h` option to display the general usage of “pdfprint” *and* all paper trays that are available (see example below):

```
$ pdfprint -P lw -h
...
The printer 'lw' supports the following paper trays:
*StandardCassette
Multipurpose
OptionalCassette
Envelope
```

The “*” marks the PPD’s default paper tray. The default tray will be used automatically if you do not specify the `-l` option *and* if you do not specify `-p` either. For details about the interdependencies between `-l` and `-p`, see the `-p` option above.

-d

Resolution allows to specify the printer's device resolution. You have to enter one of the strings that are valid for the current printer. You can prompt the list of valid strings by using the `-P` parameter together with the `-h` option:

```
$ pdfprint -P lw -h
...
The printer 'lw' supports the following resolutions:
*600dpi
300dpi
```

The "*" marks the printer's default resolution. The default value will be used if you do not specify the `-d` parameter.

PostScript language options:

-E

Export for imposition lets the program generate a PostScript file instead of printing to a spooler. This PostScript file can be used with imposition software. See 10 "Export for Imposition with ImageServer" for a detailed description.

-L

Generate PostScript Level 1 compatible code that uses some PostScript Level 2 features, provided that they are supported by the printer.

-1

Forces the program to generate PostScript Level 1 compatible code only, when transforming the PDF data into PostScript.

-2

Forces the program to generate PostScript Level 2 compatible code only, when transforming the PDF data into PostScript.

-3

Forces the program to generate PostScript 3 compatible code only, when transforming the PDF data into PostScript.

Number of copies and page selection options:

-n

Number of copies lets you specify the number of copies you want to print. The default is 1 – if you do not specify this parameter at all.

-r

Page selection lets you select the pages you want to print. There are different options available as shown in the example below:

- `2` (prints page 2 of the document only)
- `-11` (starts with page 1, prints pages 1 to 11)
- `11-` (prints pages 11 to last page)
- `2-11` (prints pages 2 to 11 – starts with page 2)
- `11-2` (also prints pages 2 to 11 – but in reverse order)

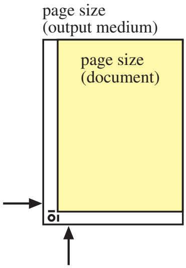
You can specify several comma-separated ranges at a time, e.g. `-r 3-6,12-`. Do not use blanks within such a specification.

If you do not specify this parameter, all pages of the document will be printed.

Orientation, margin, registration mark, shrink options:

-M

Registration marks prints registration marks. The document pages will be re-adjusted on the output medium so that the registration marks will fit in the bottom left corner. This could lead to cropping of the upper and right part of the document if the document pages *and* the registration marks do not fit on the output medium.



-b

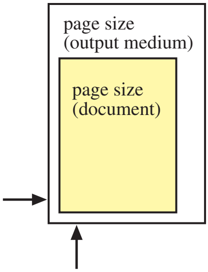
Set bleed margins `left:top:right:bottom[cm]` to specify the left, top, right, and bottom margin individually or `margin[cm]` to use the same amount for left, top, right, and bottom margin. The dimension are *inches* (default) or *centimeters*.

-f

Shrink to fit an center centers the document pages and scales them (proportionally) – if necessary – to fit on the output medium, e.g. on the paper that is used by the selected printer. Printing without the `-f` option will induce "pdfprint" to adjust the bottom left corner of the document pages to the bottom left corner of the output medium. Do not set the `-f` option if the output medium has no size specified, e.g. if you print to an imagesetter that uses film rolls. This could lead to unexpected results.

-m

Margin can be used to re-adjust the document pages on the output medium. This parameter is followed by a floating point value, e.g. `0.69`, which is used for both directions – the document pages will be shifted upwards and to the right. Using a value without a unit will imply *inches*. You can switch to *centimeters* by specifying the unit explicitly, e.g. `1.75cm`.



-o flip

`flip=horizontal` or `flip=vertical` specifies whether the PDF pages are flipped around the horizontal centerline (swap *top* and *bottom*) or around the vertical centerline (swap *right* and *left*).

-o orientation

`orientation=portrait` or `orientation=landscape` specifies whether the PDF document pages are printed to portrait or landscape orientation. The default (if this parameter is omitted) is `portrait`.

-o usemediabox

This parameter uses the “MediaBox” (always present) of a page instead of the “CropBox” for clipping the page.

-o usecropbox

This parameter uses the “CropBox” of a page for clipping the page. This is the default behavior.

-o usebleedbox

This parameter uses the “BleedBox” (if present) of a page instead of the “CropBox” for clipping the page.

In the `PDF HandShake Print...` dialog, the options `-o trimboxmarks` and `-o usebleedbox` can only be switched on together with the `Use PDF TrimBox/BleedBox` checkbox (see **Paper Settings** in 7.2 “Print PDF files using the Acrobat plug-in”).

-o usetrimbox

This parameter uses the “TrimBox” (if present) of a page instead of the “CropBox” for clipping the page.

-o useartbox

This parameter uses the “ArtBox” (if present) of a page instead of the “CropBox” for clipping the page.

PDF layer selection options:

-o layer=<selection>

Select or deselect layers. The layer selection is specified as

`[-]layer1:[-]layer2:...`

Without this option, all layers with the default state *ON* are printed. See also 17 “PDF layers”.

OPI and imagesetter options:

-o noopi

This parameter specifies whether OPI references within the PDF document are converted to PostScript or not. This can also be specified in the `PDF HandShake Print...` dialog (see **OPI Settings** in 7.2 "Print PDF files using the Acrobat plug-in").

-o gap
-o offset

`gap=value` and/or `offset=value`

These parameters are mainly meant for imagesetters. The `gap` parameter defines the length of the form feed at the end of the page. This distance will be added to the page height. The parameter is also recognized by standard desktop printers. The `offset` parameter defines the distance from the left film boundary to the left page boundary and is used to shift the whole page in X-direction. The value will not be added to the paper width. This parameter is ignored by most desktop printers. See also **Paper Settings** in 7.2 "Print PDF files using the Acrobat plug-in".

`gap` and `offset` are specified by a floating point value, e.g. `0.69`, and a unit. Using a value without a unit will imply inches. You can switch to centimeters by specifying the unit explicitly (e.g. `1.75cm`).

Color and font options:

-B

Black overprints When printing separations and "Black" appears on a colored background, usually the background separation plate will show knockouts. Setting `-B` will make sure that there are no knockouts on the background separation plates, i.e. Black will overprint the background color. This option applies to composite printing, host-based and in-RIP separations, except for pre-separated documents. This parameter is ignored for pre-separated PDF files.

-S

Spot to process lets the program convert all spot colors into process colors. The process color space depends on the printer profile that has been defined for your current printer queue and can be Grayscale, Lab, etc. Without a printer profile spot colors are converted to CMYK. This parameter is ignored for pre-separated PDF files.

-i

Ignore missing fonts induces the program to ignore fonts that are not available in the PDF file or on the server. The job will be sent to the queue and – depending on the printer's default settings for missing fonts – the job will be aborted by the printer or the missing fonts will be replaced, e.g. with *Courier*. If you do not specify this option, missing fonts will induce "pdfprint" to abort the job before it is spooled.

-s

Separations lets you specify the list of separations you wish to print. The `-s` parameter must be followed by a specification, as e.g.:

- `-s all` (prints all separations that occur on a specific document page, at least Cyan, Magenta, Yellow and Black)
- `-s Black` (prints the Black plate only)
- `-s "Pantone 387 CV","Pantone 647 CV"` (prints the specified spot colors only)

Color names that contain blanks must be quoted (see the example above), several names must be separated by a comma. Color names are case-sensitive, meaning that e.g. the CMYK color names must begin with a capital letter (`Cyan,Magenta,Yellow,Black`). Additionally, you can set up halftoning for each separation plate. For that purpose, you have to add an angle and a frequency to each color name.

More details about halftone settings as well as an example are given in section 7.1.3 “Set up halftoning”.

Note that printing separations is pre-defined for pre-separated PDF files. e.g. `-s Black` will be ignored for a pre-separated file, and all separation plates will be printed.

-o inripseparation

Prints separations using in-RIP separation instead of host-based separations. This option requires the `-s` option. If the PDF file is pre-separated, this option is ignored.

Note: This option can only be used when the printing device supports in-RIP separation.

Example 1: (Print all separations)

```
$ pdfprint -o inripseparation -s all -Pprints
```

Example 2: (Print only the black plate and one spot color)

```
$ pdfprint -o inripseparation -s "Black","HKS13" -Pprints
```

-o negativeprint

Set this parameter to negative print the whole document. Usually, this is only applied when printing separations.

-o preservedevicen

This parameter preserves DeviceN color objects in PDF files instead of converting them to the output color space. This also means that DeviceN color objects are excluded from any color transformation if color matching is active.

Note: This option can only be set when printing to a PostScript 3 device.

ICC profile and PPD file options:

-G

`Grayscale Profile` lets you specify a path to a Grayscale profile that should be used as default source profile for color matching. The default profile will be used for PDF files which are not yet “tagged” with profile information.

It is possible to specify that Grayscale colors should print with their values from the PDF document without any color matching applied. This is achieved by specifying “None” as the file name of the Grayscale input profile.

-R

`RGB Profile` lets you specify a path to an RGB profile that should be used as default source profile for color matching. The default profile will be used for PDF files which are not yet “tagged” with profile information.

It is possible to specify that RGB colors should print with their values from the PDF document without any color matching applied. This is achieved by specifying “None” as the file name of the RGB input profile.

-C

`CMYK Profile` lets you specify a path to a CMYK profile that should be used as default source profile for color matching. The default profile will be used for PDF files which are not yet “tagged” with profile information.

It is possible to specify that CMYK colors should print with their values from the PDF document without any color matching applied. This is achieved by specifying “None” as the file name of the CMYK input profile.

-o ignoreinputprofiles

Ignore all ICC profiles within the PDF document.

The following tables summarize the “pdfprint” color matching strategy. The term “`-C/-R` option” means for CMYK color data that the `-C` option was specified, and for RGB color data that the `-R` option was specified:

	no option `-C/-R` **no** option `-o ignoreinputprofiles`	option `-C/-R` **no** option `-o ignoreinputprofiles`
Color data without ICC profile	Color matching only with color space change using server default profile	Color matching with option `-C/-R` profile
Color data with ICC-based color space (PDF 1.3)	Color matching with ICC-based profile	Color matching with ICC-based profile
Color data with ICC-based color space (PDF 1.3) and with HELIOS ICC tag	Color matching with ICC-based profile	Color matching with ICC-based profile
Color data with HELIOS ICC tag	Color matching with HELIOS ICC tag profile	Color matching with HELIOS ICC tag profile

	no option `-C/-R` option `-o ignoreinputprofiles`	option `-C/-R` option `-o ignoreinputprofiles`
Color data without ICC profile	Color matching only with color space change using server default profile	Color matching with option `-C/-R` profile
Color data with ICC-based color space (PDF 1.3)	Color matching only with color space change using server default profile	Color matching with option `-C/-R` profile
Color data with ICC-based color space (PDF 1.3) and with HELIOS ICC tag	Color matching only with color space change using server default profile	Color matching with option `-C/-R` profile
Color data with HELIOS ICC tag	Color matching only with color space change using server default profile	Color matching with option `-C/-R` profile

The following tables summarize the “pdfprint” color matching strategy for Grayscale (option `-G`) data:

	no option `-G` **no** option `-o ignoreinputprofiles`	option `-G` **no** option `-o ignoreinputprofiles`
Color data without ICC profile	No color matching, output data is Grayscale	Color matching with option `-G` profile
Color data with ICC-based color space (PDF 1.3)	Color matching with ICC-based profile	Color matching with ICC-based profile
Color data with ICC-based color space (PDF 1.3) and with HELIOS ICC tag	Color matching with ICC-based profile	Color matching with ICC-based profile
Color data with HELIOS ICC tag	Color matching with HELIOS ICC tag profile	Color matching with HELIOS ICC tag profile

	no option `-G` option `-o ignoreinputprofiles`	option `-G` option `-o ignoreinputprofiles`
Color data without ICC profile	No color matching, output data is Grayscale	Color matching with option `-G` profile
Color data with ICC-based color space (PDF 1.3)	No color matching, output data is Grayscale	Color matching with option `-G` profile
Color data with ICC-based color space (PDF 1.3) and with HELIOS ICC tag	No color matching, output data is Grayscale	Color matching with option `-G` profile
Color data with HELIOS ICC tag	No color matching, output data is Grayscale	Color matching with option `-G` profile

-D

PPD file name allows specifying a server path for a specific PPD file. With this parameter you can select for this particular job a PPD file other than the one that has been specified for your current HELIOS printer queue.

Halftone options:

-H

Preserve PDF halftones will use the halftone information that is contained in a PDF file for printing.

A PDF file can contain halftone information for individual elements or for all elements. When setting the `-H` parameter, the available halftone information will be used. Else, if you do not specify this parameter, all halftone information in the PDF file will be ignored. Note that it depends on the Distiller's job options whether a PDF file contains any halftone information at all.

For more explanations about halftone settings, see 7.1.3 “Set up halftoning”.

-a

Select halftone angle lets you specify a global halftone angle for printing. The parameter must be followed by a floating point value, e.g. `84.8`. If you print separations you can set a different halftone angle for each separation plate. For that purpose you have to use the `-s` parameter.

-l

Select halftone frequency lets you specify a halftone frequency for printing. The parameter `-l` is followed by a floating point value, e.g. `33.39`. Using the value without a unit will imply lines per inch. You can switch to lines per centimeter by specifying the unit explicitly (e.g. `84.8lcm`).

You can list the frequency options from the current PPD file by using the `-P` parameter together with the `-h` option.

Example:

```
$ pdfprint -P lw -h
...
The predefined halftones of printer 'lw' are:

60lpi at 300dpi
53lpi at 300dpi
*85lpi at 600dpi
71lpi at 600dpi
```

The “*” marks the PPD’s default setting. If you wish to select another option from the PPD file you must specify both the `-l` and the `-d` parameter. For example, to select the first entry from the PPD file, you must specify:

```
pdfprint ....-l 60 -d 300dpi
```

With the -h option, you can also display the complete halftone settings for your current print job, the "*" marks the halftone settings for spot colors:

```
$ pdfprint -P lw -h
...
The default halftone frequencies and angles are:

Black at 84.85 lpi and 45.00 degrees
Cyan at 94.87 lpi and 71.57 degrees
Magenta at 94.87 lpi and 18.44 degrees
Yellow at 30.00 lpi and 0.00 degrees
*CustomColor at 84.85 lpi and 45.00 degrees
```

The values that are displayed above are related to the default of the current PPD file (85 lpi at 600 dpi). If you switch for example to 60 lpi at 300 dpi and use -h again, the list will be updated accordingly (see example below):

```
$ pdfprint -P lw -l 60 -d 300dpi -h
...
The default halftone frequencies and angles are:

Black at 84.85 lpi and 45.00 degrees
Cyan at 94.87 lpi and 15.00 degrees
Magenta at 94.87 lpi and 75.00 degrees
Yellow at 30.00 lpi and 0.00 degrees
*CustomColor at 84.85 lpi and 45.00 degrees
```

If your PPD file did not contain any halftone specifications at all, the above list would show the defaults we set with PDF HandShake. These PDF HandShake defaults would be used if you did not specify any frequency or angle values (with -s, -l or -a), if you did not specify -H or -u, and if there were no information in the PPD file.

Find more details about halftone settings in 7.1.3 "Set up halftoning".

-u

Use printer's default screening is only relevant for printing composite. If you print separations this parameter will be ignored.

With -u, the default halftone settings of the final output device will be used when printing the PDF document. If you use -u together with `-H` the printer's halftone settings will only be applied to elements that do not have their own halftone information.

For more explanations about halftone settings, see 7.1.3 "Set up halftoning".

Help, job title and feature options:

-h

Display help. Additionally, you can list all medium names, paper trays, resolutions, halftone information, and features that are valid for your current printer by combining `-h` with the `-P` parameter that specifies a particular printer. Examples are given below; see options `-p`, `-I`, `-d`, `-l`, and `-F`.

-T <Title>

Title uses `<Title>` for this print job in an "lpr" printer queue. The default title is the document file name.

-F <Feature>

Feature allows selecting printer features. `<feature>` is constructed of a key name and an optional option, separated by '='. You can only specify valid features. The list of valid features for the current printer is prompted by using the `-P` parameter together with the `-h` option (see example below):

```
$ pdfprint -P lw -h
...
The features of printer 'lw' are:
InstalledMemory=None
InstalledMemory=16Meg
InstalledMemory=32Meg
OptionalCassette1=True
OptionalCassette1=False
OptionalCassette1=Preferred
OptionalEnvelopeFeeder=True
OptionalEnvelopeFeeder=False
Smoothing=True
Smoothing=False
BitsPerPixel=4
BitsPerPixel=None
TraySwitch=True
TraySwitch=False
```

You can specify several features at a time. Each feature must be preceded by `-F`. Note that some features do not have any effect on your output results.

Check your printer's manual for more details.

7.1.3 Set up halftoning

Printing composite (for composite PDF files only):

The PDF HandShake software has its own default halftone settings. When printing composite, the program's defaults can be overwritten by the PPD file (if it contains halftone settings) or by specifying the `-l` and/or the `-a` option. Note that halftoning will be set for CMYK if you print composite to a color printer, and it will be set for *Black* only if you print composite to a B/W printer.

Neither you, nor PDF HandShake, nor the PPD file will be able to set any values at all if you specify `-u`. The device defaults will be valid then.

Finally, the `-H` option will make sure that the current halftone settings are only applied to those elements in a PDF file that do not have their own halftone information.

Printing separations:

There are two differences between printing composite and printing separations: First, the `-u` parameter is not available for printing separations. Second, you can use the `-s` parameter to specify individual frequency and angle values for one or more separation colors.

7.1.4 Examples

In the following examples, the file “abc.pdf” in the HELIOS volume “data” is being printed to the HELIOS printer queue “lw”:

```
$ pdfprint -i -n 3 -r -6,85- -P lw /usr/data/abc.pdf
```

prints three copies of pages 1 to 6 and pages 85 to last page and ignores missing fonts, if there are any.

```
$ pdfprint -s "Pantone 387 CV" -r 1 -P lw /usr/data/abc.pdf
```

prints one plate of spot color “Pantone 387 CV” of document page 1.

```
$ pdfprint -S -s all -r 1 -P lw /usr/data/abc.pdf
```

converts all spot colors into process and prints page 1 of the resulting separation plates (here: CMYK).

```
$ pdfprint -f -r 22-1 -P lw -p a5 /usr/data/abc.pdf
```

prints pages 1 to 22 in reverse order and scales them to size A5 (5.76 x 8.26 inches).

```
$ pdfprint -P lw -R /usr/ICC-Profiles/Scanner/"TOPAZ Durchsicht Fuji ICC"
                  /usr/data/abc.pdf
```

prints the complete document and uses the profile “TOPAZ Durchsicht Fuji ICC” as default RGB source profile for color matching.

```
$ pdfprint -P lw -r 1-3 -l 71 -d 600dpi -H /usr/data/abc.pdf
```

prints pages 1 to 3 with 71 lpi, 600 dpi, and default halftone angles, while preserving the halftone information that is already included in the document.

```
$ pdfprint -P lw -r 1 -s all -l 60 -a 44 -B /usr/data/abc.pdf
```

prints all separation plates of page 1 with 60 lpi and 44°. Black overprints any background color.

7.2 Print PDF files using the Acrobat plug-in

Our Acrobat print plug-in allows printing PDF files from your Acrobat application with many options that are not available with the built-in print dialog.

Access to the plug-in

Make sure that the "PDF HandShake" plug-in for Acrobat is already available in the "Plug-Ins" subfolder of your Acrobat program folder. Otherwise, see 3 "Installation".

➢ Start your Acrobat application. The `PDF HandShake Print` dialog is opened from the `File` menu as shown in Fig. 7.1.

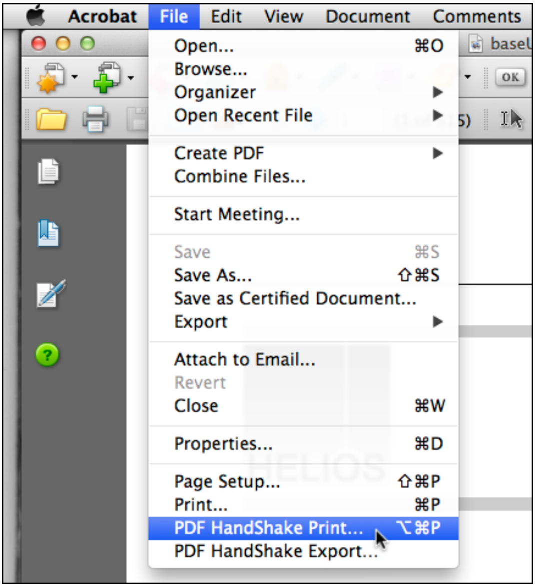


Fig. 7.1: Selecting `PDF HandShake Print...` from the menu

Note: If you made any changes in your document you must save them before printing. The PDF HandShake plug-in always prints the last saved document version. If this differs from the version that is currently open, access to the print dialog is impossible and an error message appears instead.

PDF HandShake Print

Fig. 7.2 shows the “PDF HandShake Print” dialog. The selected printer queue is stated in the dialog’s title bar.

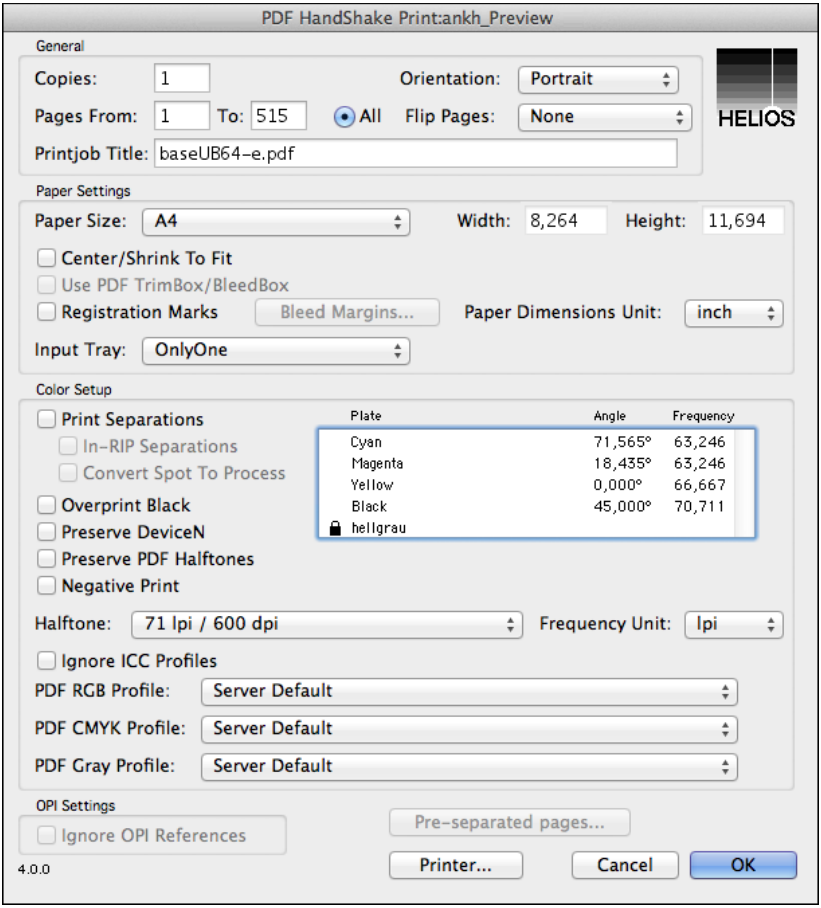
General

In the “General” section of the dialog, you can specify the number of copies, the page range you want to print, the orientation (Portrait/Landscape), and whether you want to flip pages.

Paper Settings

The “Paper Settings” section contains the default settings of the PPD file that is currently selected for your spooler.

From the `Paper Size` pop-up menu you can choose one of the printer’s paper sizes (as shown in Fig. 7.2) or `Custom`. `Custom` *must* be selected if you want to enter individual values in the `Height` and `Width` text fields (Fig. 7.3). Dimension units can be displayed either in `inch` or `cm`. Note that if you print with custom values these values will be saved. They will be available the next time you open the dialog. The `Paper Size` pop-up menu contains the entry `(PDF Page Size)`. This entry can be selected to make sure that the pages in the PostScript file are exactly as large as the PDF pages in the document. If you choose (`PDF Page Size`), the `Orientation` and `Center/Shrink To Fit` options will be disabled.



PDF HandShake Print:ankh_Preview

General

Copies: 1 Orientation: Portrait

Pages From: 1 To: 515 ● All Flip Pages: None

Printjob Title: baseUB64-e.pdf

Paper Settings

Paper Size: A4 Width: 8,264 Height: 11,694

□ Center/Shrink To Fit

□ Use PDF TrimBox/BleedBox

□ Registration Marks Bleed Margins... Paper Dimensions Unit: inch

Input Tray: OnlyOne

Color Setup

□ Print Separations

□ In-RIP Separations

□ Convert Spot To Process

□ Overprint Black

□ Preserve DeviceN

□ Preserve PDF Halftones

□ Negative Print

Plate	Angle	Frequency
Cyan	71,565°	63,246
Magenta	18,435°	63,246
Yellow	0,000°	66,667
Black	45,000°	70,711
🔒 hellgrau		

Halftone: 71 lpi / 600 dpi Frequency Unit: lpi

□ Ignore ICC Profiles

PDF RGB Profile: Server Default

PDF CMYK Profile: Server Default

PDF Gray Profile: Server Default

OPI Settings

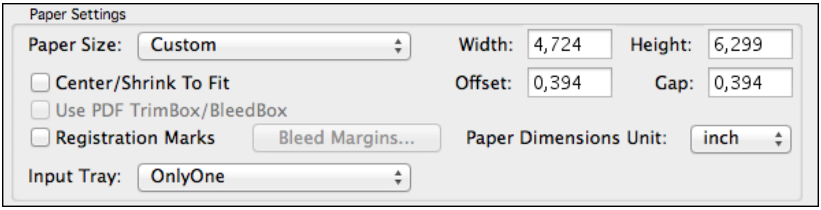
□ Ignore OPI References

Pre-separated pages...

4.0.0

Fig. 7.2: The “PDF HandShake Print” dialog

If you select `Custom` from the `Paper Size` pop-up menu, two new options in the dialog will become available, namely `Offset` and `Gap`. The `Offset` value defines the distance from the left boundary of the print medium, e.g. a film on an imagesetter, to the left page boundary of the document, and is used to shift the whole page in X-direction. The `Offset` value will not be added to the page width. If your width is e.g. 15 cm, it will still be 15 cm after specifying an offset of e.g. 1 cm. In that case, the document page must fit on the remaining 14 cm. This parameter is ignored by most standard desktop printers. The `Gap` value defines the length of the form feed at the end of the page. This distance will be added to the page height. This parameter is recognized by desktop printers as well.



Paper Settings

Paper Size: Custom

Width: 4,724 Height: 6,299

□ Center/Shrink To Fit

Offset: 0,394 Gap: 0,394

□ Use PDF TrimBox/BleedBox

□ Registration Marks

Paper Dimensions Unit: inch

Input Tray: OnlyOne

Fig. 7.3: Setting custom paper sizes

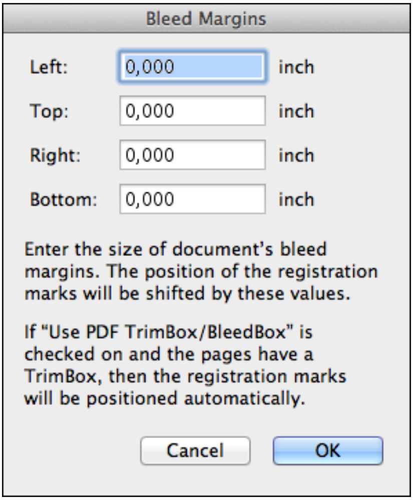
If you specify one of the printer's paper sizes you must make sure that the selected input tray uses this paper size. Otherwise, the printer may be pending until you either submit the right print medium or delete the job and restart the printer. `Center/Shrink To Fit` can be selected to fit your document pages on the selected paper size.

PDF HandShake includes a feature called "Bleeding", which is used by direct printouts via the PDF HandShake plug-in or the UNIX "pdfprint" command (see also 7.1.2 "Options" above in this chapter). With PDF 1.2 based files and layout applications, e.g. QuarkXPress, the customer needs to create a larger page size (e.g. A4 + 3 mm bleeding) or create PostScript from QuarkXPress with registration marks.

The PDF HandShake plug-in allows specifying the bleeding parameters for the four page-borders (top, left, right, bottom). The correct bleeding parameters will be used to adjust the registration marks in order to represent the real document size.

Beginning with PDF 1.3, PDF documents can include a “TrimBox” to define the intended dimensions of the finished page. They can also include a “BleedBox” to define the region to which the content should be clipped when output in a production environment, including extra “bleed area”. When printing such a document, enable the `Use PDF TrimBox/BleedBox` checkbox.

Activate the `Registration Marks` option to print PDF documents with registration marks. The document pages will be re-adjusted on the output medium so that the registration marks will fit in the bottom left corner. This could lead to cropping of the upper and right part of the document if the document pages *and* the registration marks do not fit on the output medium. When this option is active, you can click the `Bleed Margins...` button and additionally specify `Bleed Margins` for `Left`, `Top`, `Right`, and `Bottom` (Fig. 7.4).



Bleed Margins

Left: 0,000 inch

Top: 0,000 inch

Right: 0,000 inch

Bottom: 0,000 inch

Enter the size of document’s bleed margins. The position of the registration marks will be shifted by these values.

If “Use PDF TrimBox/BleedBox” is checked on and the pages have a TrimBox, then the registration marks will be positioned automatically.

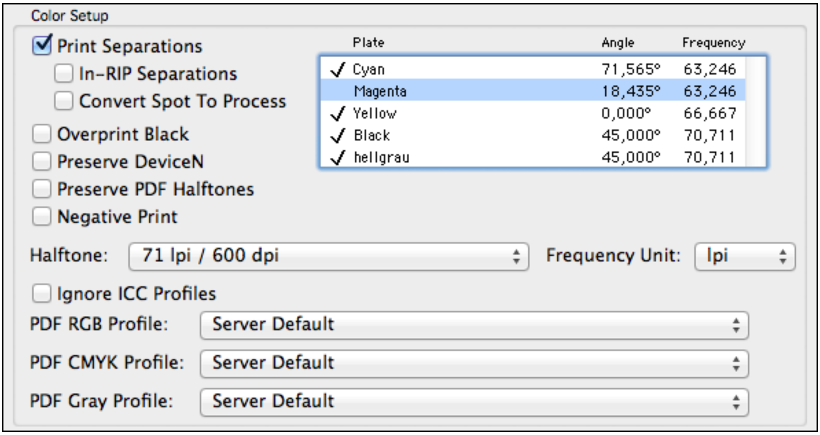
Fig. 7.4: “Bleed Margins” dialog

If you specify an `Input Tray` keep in mind that the selected paper size must be valid for this particular tray.

Color Setup

With the `Print Separations` option selected, individual plates to print and individual angles/frequencies for each plate can be specified.

A check mark (√) adjacent to each plate color indicates that it will be printed. You can click on a check mark to unselect the respective separation color. An example is given in Fig. 7.5 below.



Color Setup

☑ Print Separations

□ In-RIP Separations

□ Convert Spot To Process

□ Overprint Black

□ Preserve DeviceN

□ Preserve PDF Halftones

□ Negative Print

	Plate	Angle	Frequency
✓	Cyan	71,565°	63,246
	Magenta	18,435°	63,246
✓	Yellow	0,000°	66,667
✓	Black	45,000°	70,711
✓	hellgrau	45,000°	70,711

Halftone: 71 lpi / 600 dpi

Frequency Unit: lpi

□ Ignore ICC Profiles

PDF RGB Profile: Server Default

PDF CMYK Profile: Server Default

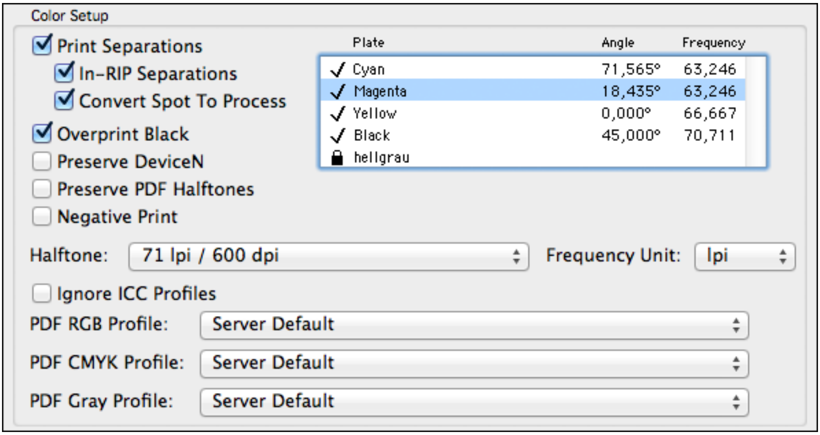
PDF Gray Profile: Server Default

Fig. 7.5: Printing separations

`In-RIP Separations`: If separations are selected for composite PDF documents, PDF HandShake offers two ways to separate color plates from each composite PDF page. The default host-based separation generates a monochrome PostScript page for each composite PDF page and for each separation plate. The PostScript file consisting of these monochrome pages is sent to the printer. The other method, called in-RIP separation, generates one composite PostScript page for each composite PDF page. The PostScript file consisting of these composite pages is sent to the printer, which is configured to make separations from it. The `In-RIP Separations` option is ignored for pre-separated PDF documents. If your printer does not support in-RIP separation, it generates composite output.

PDF HandShake will convert all spot colors to process colors if `Convert Spot To Process` is checked. The spot colors in the text field will then be marked by the "locked" symbol to indicate that they will not be printed on their own separation plates (Fig. 7.6). The process color space depends on the printer ICC profile that has been defined for your current printer queue. By default, (i.e. if the `Default Printer Profile` in the HELIOS Admin `ICC` is set to `None`), the spot colors are converted into CMYK.

Whenever Black appears on a colored background, the background separation plate will usually show knockouts. `Overprint Black` prevents this by overprinting the background color.



Color Setup

☑ Print Separations
☑ In-RIP Separations
☑ Convert Spot To Process
☑ Overprint Black
□ Preserve DeviceN
□ Preserve PDF Halftones
□ Negative Print

	Plate	Angle	Frequency
✓	Cyan	71,565°	63,246
✓	Magenta	18,435°	63,246
✓	Yellow	0,000°	66,667
✓	Black	45,000°	70,711
🔒	hellgrau		

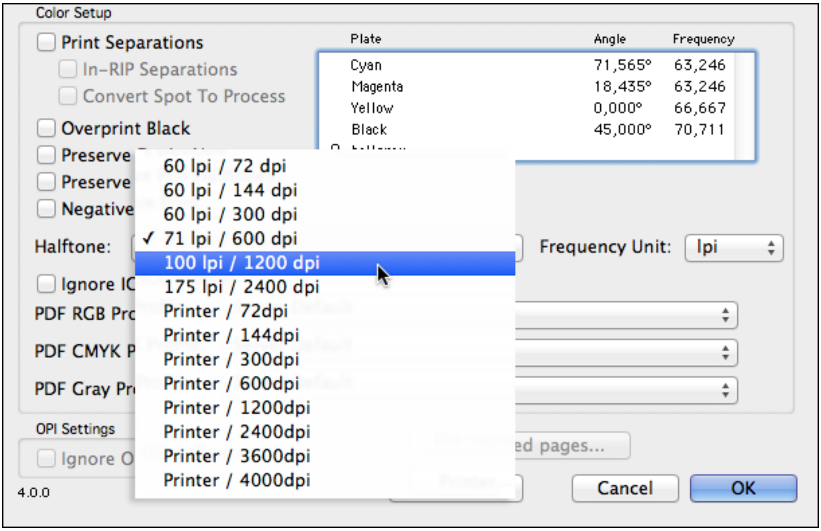
Halftone: 71 lpi / 600 dpi
Frequency Unit: lpi

□ Ignore ICC Profiles
PDF RGB Profile: Server Default
PDF CMYK Profile: Server Default
PDF Gray Profile: Server Default

Fig. 7.6: Printing separations with `Spot To Process`

A PDF file can contain halftone information for individual elements or for all elements. When specifying `Preserve PDF Halftones`, the available halftone information will be used. Else, if you do not check this option, all halftone information in the PDF file will be ignored. Note that it depends on the Distiller job options whether a distilled PDF file contains any halftone information at all. From the `Halftone` pop-up menu you may select global halftone settings. These settings will be used for all elements in the PDF file unless you have checked `Preserve PDF Halftones`, and they will be used for all separation plates unless you enter individual values for a specific plate (this is described hereinafter). Fig. 7.7 shows the Halftone pop-up menu. It contains all entries that are available in the PPD file. The `Printer / *dpi` entries at the end of the list are only available if you print *composite*. These options can be used to set a resolution, but leave the halftoning setup to the final output device.

All angles and frequencies that correspond to the halftone setup are listed in the text field after the color names. Frequencies can be displayed either in `lpi` (*lines per inch*) or in `l/cm` (*lines per centimeter*). You can select the desired unit from the `Frequency Unit` pop-up menu.



Color Setup

□ Print Separations

□ In-RIP Separations

□ Convert Spot To Process

□ Overprint Black

□ Preserve

□ Preserve

□ Negative

Plate	Angle	Frequency
Cyan	71,565°	63,246
Magenta	18,435°	63,246
Yellow	0,000°	66,667
Black	45,000°	70,711

Halftone: ✓ 71 lpi / 600 dpi

Frequency Unit: lpi

□ Ignore IC

PDF RGB Pro

PDF CMYK P

PDF Gray Pro

OPI Settings

□ Ignore O

4.0.0

Fig. 7.7: Setting up halftoning

You can overwrite the global halftone settings by individual settings for each separation plate. For that purpose, double-click the desired color in the text field and then edit the dialog that is shown in Fig. 7.8.

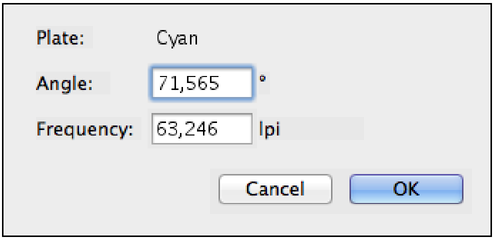


Plate: Cyan

Angle: 71,565 °

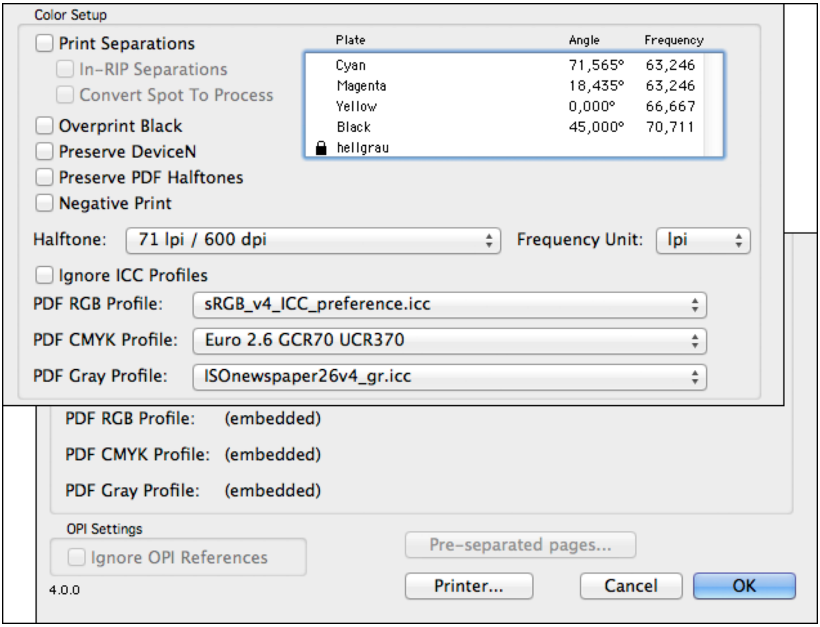
Frequency: 63,246 lpi

Fig. 7.8: Changing halftone settings for a single separation plate

The `Negative Print` option affects the whole document and is usually only applied when printing separations.

The `PDF RGB Profile`, `PDF CMYK Profile`, and `PDF Gray Profile` pop-up menus are used for color matching. Together with the printer profile that may be specified for your printer queue (see 5.2.2 “Printer queue settings for PDF”), accurate color matching for all RGB, CMYK, and Grayscale objects in your PDF document can be achieved.

When opening the print dialog for a document that does not contain any profile information, you may set the profile pop-up menus to `Server Default` to use the default server source profiles (see 5.2.2 “Printer queue settings for PDF”). Switch to `Choose` to select other profiles from the “ICC-Profiles” volume or any other server location you may use. For documents that are already tagged with profile information, the behavior is different: Explicitly stated profiles in the dialog (path and file name) indicate that your current document has been tagged by reference. The statement “embedded” indicates that the profiles themselves are contained in the document. Examples are given in Fig. 7.9.



Color Setup

□ Print Separations

□ In-RIP Separations

□ Convert Spot To Process

□ Overprint Black

□ Preserve DeviceN

□ Preserve PDF Halftones

□ Negative Print

Plate	Angle	Frequency
Cyan	71,565°	63,246
Magenta	18,435°	63,246
Yellow	0,000°	66,667
Black	45,000°	70,711
hellgrau		

Halftone: 71 lpi / 600 dpi

Frequency Unit: lpi

□ Ignore ICC Profiles

PDF RGB Profile: sRGB_v4_ICC_preference.icc

PDF CMYK Profile: Euro 2.6 GCR70 UCR370

PDF Gray Profile: ISOnewspaper26v4_gr.icc

PDF RGB Profile: (embedded)

PDF CMYK Profile: (embedded)

PDF Gray Profile: (embedded)

OPI Settings

□ Ignore OPI References

4.0.0

Fig. 7.9: Profile information for tagged documents

You can also choose `None` from the `PDF RGB Profile`, `PDF CMYK Profile`, and `PDF Gray Profile` pop-up menus to suppress color matching during output. For example, the printer profile that has been specified in the queue's `ICC` settings dialog (in HELIOS Admin) is a CMYK profile, you may set the `PDF CMYK Profile` to `None` to print the CYMK colors in your document without any color matching. This can be desired if the CYMK colors in the document are already matched to the printer. If you select `Server Default` from the pop-up menu, the default profile that has been set in the HELIOS Admin `PDF HandShake Settings` dialog will be used as input profile for color matching.

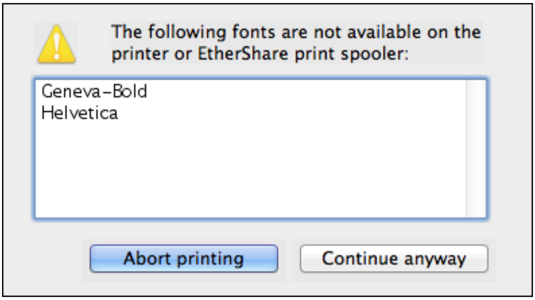
It does not make any sense to set the `PDF RGB Profile`, the `PDF CMYK Profile`, and the `PDF Gray Profile` to `None` because color matching will always be applied to objects that are not in the printer's color space. For example, if your printer profile is a CMYK profile, all RGB and Lab colors in your document will definitely be matched.

OPI Settings

Printing PDF documents can be done without resolving OPI references. This is useful e.g. if the high-resolution images are already embedded in the PDF document and the images are not stored on the HELIOS server where the PDF document is printed. If you wish to ignore OPI references that are embedded in the PDF document, activate the `Ignore OPI References` checkbox.

The button `Pre-separated pages` in the print dialog opens a window that allows you to check the separation plate information of all pages in your document. The button is activated for pre-separated documents only; for composite documents it is grayed out.

Finally, the plug-in will look for missing fonts. In the case that a non-embedded document font is not available on the HELIOS server or on the final output device, there will be a warning that allows you to abort the job or to print anyway (Fig. 7.10). Obtain more details about font handling in A "About fonts".



The following fonts are not available on the printer or EtherShare print spooler:

Geneva-Bold
Helvetica

Fig. 7.10: Information about missing fonts

Whenever you click `Print`, most of the `PDF HandShake Print` settings are saved and preserved for the next time you open the dialog. This does not affect the "General" section; the entries in this section will always be reset to default.

Pre-separated PDFs

Pre-separated PDF documents are automatically recognized by the "PDF HandShake Print" dialog. When printing pre-separated documents with PDF HandShake, the following rules apply:

- `Print Separations` is turned on
- The options `In-RIP Separations`, `Convert Spot To Process` and `Overprint Black` are disabled
- Color matching (`PDF RGB/CMYK/Gray Profiles`) is disabled
- The halftone settings for each color are applied to the pages with the corresponding plate colors

8 Create PDF Server

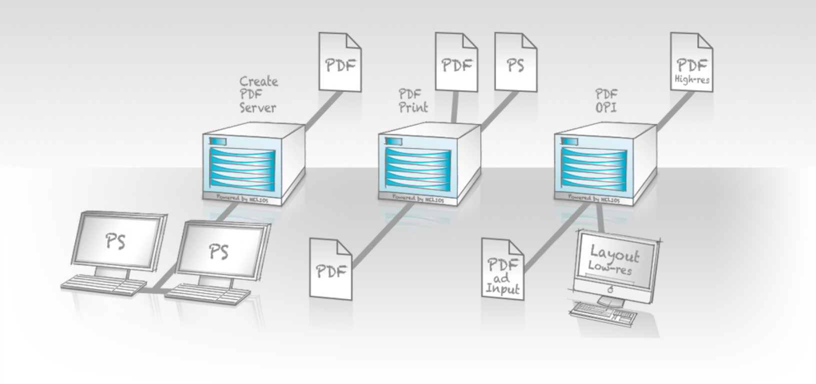


Fig. 8.1: Workflow: Create PDF Server

PDF HandShake has a feature to automate the PDF creation of any PostScript job. This is done by creating a new printer queue in HELIOS Admin, with the connection type `Create PDF`. This queue prints the PostScript job over the network to a dedicated Mac or Windows based workstation which is used to create the PDF file. After a PDF file is successfully created, it is returned to the HELIOS printer queue.

Create PDF Server utilizes Adobe Acrobat Distiller, which must be purchased separately. It is important to comply with the licensing terms of Adobe Acrobat. Specific terms related to using Acrobat on a server can be found in the Acrobat license file. The purpose of Create PDF Server is to use Acrobat Distiller more efficiently, reduce administrative and training costs, and to ensure consistent, reliable PDF job production.

Alternatively, you may use Ghostscript to convert the PostScript print job into a PDF file. Ghostscript is license-independent and is available for major platforms. However, it can only generate low-end PDF documents.

Ghostscript binary

Ghostscript ("gs" binary) is available in various distributions, free of charge. HELIOS supports all current Ghostscript 8.x and 9.x versions. The Ghostscript executable ("gs" on UNIX, "gswin32c.exe" on Windows) must be installed in some standard directory, or the **ghostscript** preference must be set to the executable path.

The benefits of Create PDF Server are:

- ❍ Server-based automatic PDF creation from any UNIX, Mac or Windows client by printing to a HELIOS "Create PDF" print queue
- ❍ Create PDF Server works without any local configuration of the Mac/ Windows server, the Distiller settings are attached to the HELIOS printer queue
- ❍ PDF creation requires a lot of computing which will be off-loaded from the host to a dedicated workstation, e.g. a Mac with a fast network connection
- ❍ Office PDFs: Ghostscript PDF generation support

Note: There is no speed up for the Adobe Distiller or Ghostscript from multi-processor machines.

- ❍ There is no temporary PostScript file stored on the dedicated server
- ❍ There is only the spooled job stored on the HELIOS server. Font or OPI resolving is done *on-the-fly* and does not use additional disk space on the HELIOS server.

- Error Messages will be saved to the HELIOS printer accounting log file and sent back to the user by e-mail
- The created PDF document will be attached to the hold queue, similar to the HELIOS PrintPreview application
- The created PDF document can be saved to any HELIOS volume or into a subfolder which can be created for each user name
- Job distribution to the next available Create PDF Server – HELIOS supports balance queues which allow a cluster of "dedicated" servers which are used to distill many print jobs in parallel
- Distiller security settings can be specified individually for each queue with HELIOS Admin
- No shared folder/volume needed; Create PDF Server behaves like a network printer
- Job-pipelining via multiple queues allows the creation of PDF files with different settings, e.g. one input job produces web, press and archive optimized PDF files
- Create PDF Server fail safety; when multiple Create PDF Servers are used, automatic load balancing (cluster) ensures availability in case one server node should fail

Section 8.2 "Set up a "Create PDF" printer queue with HELIOS Admin", presents and explains the features within HELIOS Admin which are specific to Create PDF Server. However, other features which are already described in the corresponding chapter in the HELIOS Base manual are merely referenced.

8.1 Installation

8.1.1 Prerequisites

Mac OS 9

- A dedicated Power Mac running Mac OS 9, with a working TCP/IP connection and at least 64 MB RAM
- Acrobat 4.0.5 (Acrobat 4.0 will *not* work!), Acrobat 5

OS X

- A Mac (Intel or PPC) with a working TCP/IP connection, running OS X 10.3.5 or later,
- Acrobat 6/7/8/9/X/XI

Windows

- A PC with a working TCP/IP connection, running Windows Vista, Windows Server 2008 or newer
- Acrobat 4.0.5/5/6/7/8/9/X/XI

All platforms

- A host running PDF HandShake and EtherShare, PCShare or WebShare

In order to ensure trouble-free server operation it is most important that the “dedicated” server is not employed with tasks other than running the Create PDF Server process.

Ghostscript

During the installation of the HELIOS software modules, “createpdf” is created in the “HELIOSDIR/sbin” directory. It is started every time a print job is sent to port 2018 (see “TCP ports used by HELIOS” in the HELIOS Base manual).

8.1.2 Server memory requirements

The performance of Create PDF Server on the dedicated server increases with the size of the available RAM memory.

8.1.3 Mac server installation

➢ Mount the “HELIOS Applications” volume and open *MacOS > PDF Tools > Create PDF Server*.

Important: For instructions on how to install Create PDF Server on different Mac systems, refer to the “README.txt” document!

8.1.4 Windows server installation

➢ Connect the “HELIOS Applications” network drive and open *Windows > PDF Tools > Create PDF Server*.

Important: For instructions on how to install Create PDF Server on different Windows systems, refer to the “README.txt” document!

There are two ways to make use of Create PDF Server, by installation as a service or by use in a console window:

Installation

Advantages:

- Runs as background application
- No user login is needed

- Error reporting in “Windows Event Log” file

➢ Double-click “InstallPDFService”, which installs “HELIOS Create PDF Server” as a Windows Service. This means that it is started each time the Windows PC is booted. Calling this install script will also start the server immediately.

Verification of the installation:

➢ Start the Windows “cmd” program and change to the “Create PDF Server” directory. Type `"Create PDF Server.exe" -s` and press RETURN (`-s` = “status”).

Your Create PDF Server service installation was successful if the line `Service is installed`. appears in the Windows “cmd” program command line.

Alternatively, you can do the following:

➢ Open “Start > Control Panel > Administrative Tools > Services”. Then double-click the “Create PDF Server” entry.

Removal of the service:

➢ In the Create PDF Server folder double-click “RemovePDFService.bat”.

Use in a console window

Advantage:

- Job processing can be monitored

➢ To start Create PDF Server manually (in a console window), double-click the Create PDF Server icon.

➢ To stop Create PDF Server use “Ctrl-Pause” (*Break* key functionality) or close the console window.

Important: Create PDF Server can run as a service or in a console window – but never simultaneously!

Error logging

In the event of an error, respective messages are written to the “Windows Event Log” and can be displayed with the Windows `Event Viewer`.

Adopt settings (Acrobat 5 and later)

If Create PDF Server is used as a service, you need to run the “AdoptSettings.exe” program each time you change the `Font Locations...` settings within Acrobat Distiller. This is because Acrobat stores its settings in user-dependent trees of the Windows Registry. In order to make these settings available to the service, “AdoptSettings.exe” copies the settings to a global node.

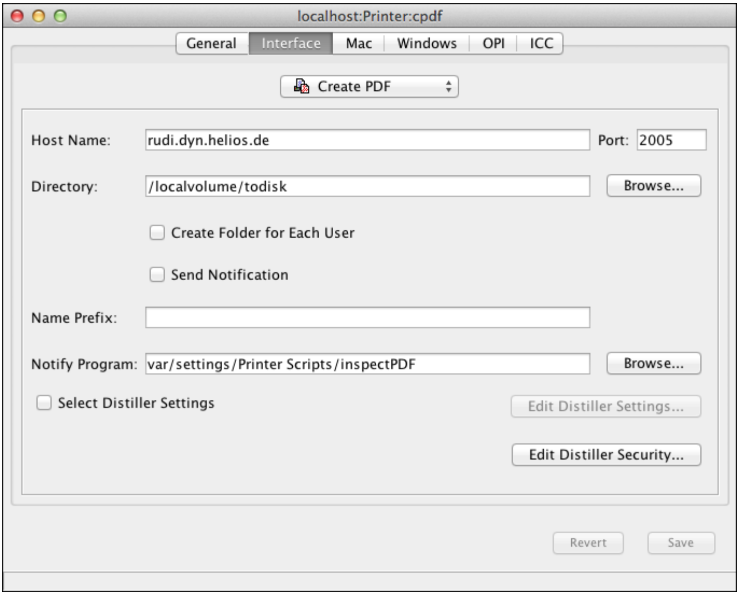
Note: When running Create PDF Server on Windows as a system service, jobs may get stuck if Adobe Acrobat tries to display a dialog. System services on Windows run as user “SYSTEM”, so Acrobat’s dialogs about registration, activation, demo expire time or updates are not shown to the user that is currently logged-in. To make dialogs being shown to the currently logged-in user, open the service’s properties (`Computer Management > Services > HELIOS Create PDF Server`), select the tab `Log on`, and check `Allow service to interact with desktop`. To avoid these dialogs and stuck jobs, you should activate Acrobat and disable registration and automatic updates.

8.2 Set up a “Create PDF” printer queue with HELIOS Admin

Creating or editing printer queues with HELIOS Admin, which is essential for the following, is described in detail in the HELIOS Base manual. In this chapter, the focus is on the “Create PDF” printer queue only.

8.2.1 Create PDF connection

➢ In the HELIOS Admin `Printers` tab select `File > New` and specify a new printer according to the instructions given in the Base manual. Select `Create PDF` from the `Connection` pop-up menu, so the `Interface` tab comes into the foreground (Fig. 8.2).



Host Name: rudi.dyn.helios.de Port: 2005

Directory: /localvolume/todisk

□ Create Folder for Each User

□ Send Notification

Name Prefix:

Notify Program: var/settings/Printer Scripts/inspectPDF

□ Select Distiller Settings

Fig. 8.2: `Create PDF` printer connection

➢ Enter the IP address of the host where Create PDF Server and Acrobat Distiller are installed in the `Host Name` field, and specify the port number in the `Port` field.

Note: The port number allows you to specify whether the PDF is generated via Create PDF Server (2005) or Ghostscript (2018).

Instead of the IP address you may also specify the server host name on the network. However, this requires a configured and working DNS server on the network.

The directory path to the location where the processed PDF print jobs are saved must be specified in the `Directory` field. You may also use the `Browse...` button to select the spool directory. If the option `Create Folder for Each User` is active, a subfolder for each print job “owner” is created in the directory path where the print jobs are stored. The user name is the `Owner Name` as stated in the `%% For: (name)` string inside the PostScript job.

Note: If the user name is not known to the HELIOS server, the PDF will be stored inside a folder “nobody”.

If the `Send Notification` checkbox is activated, the “Create PDF” interface reports events to all listening services.

Note: Note that with the option `Send Notification` activated, layouts are generated if the target volume is configured to create layouts.

If desired, you can enter a `Name Prefix` and a `Notify Program`. The prefix serves to identify the files that are coming from this specific printer queue (in case you have several “Create PDF” printer queues which print to the same destination). The `Notify Program` option lets you enter a path that leads to a specific program. Again, you may also use the `Browse...` button to select a notify program which will be started automatically after printing has been successfully finished. A notify program can perform any arbitrary action.

Select Distiller Settings

To set up the preferred Distiller settings for the “Create PDF” printer queue, do the following:

➢ Start Acrobat Distiller and select a pre-defined PDF job option from the `Default Settings` pop-up menu in the `Adobe PDF Settings` section.

➢ Open `Settings > Edit Adobe PDF Settings...`. Modify the settings according to your needs (see 15 “Create PDF files using Acrobat Distiller”) and save your changes – under a new name – to the Distiller “Settings” folder by clicking `Save as...`. Optionally, the job options file can be saved to the HELIOS “Settings” volume, in the “Distiller Settings” folder. Then highlight your “Create PDF” printer queue in the HELIOS Admin `Printers` tab and select `Settings` from the `Printer` menu. In the `Interface` tab activate the `Select Distiller Settings` checkbox and enter the path to the job options file in the text field or click `Browse...` to search for it.

Now your job option settings have been saved to the “Create PDF” printer queue on the host and are stored in the “SETTINGS” file of the spool directory.

Edit Distiller Settings

In the `Interface` tab of the “Create PDF” printer queue you can display the contents of the Distiller settings file in a text window. “root” and members of the “SysAdm” or “QueueAdm” group can edit it:

➢ Click the `Edit Distiller Settings...` button.

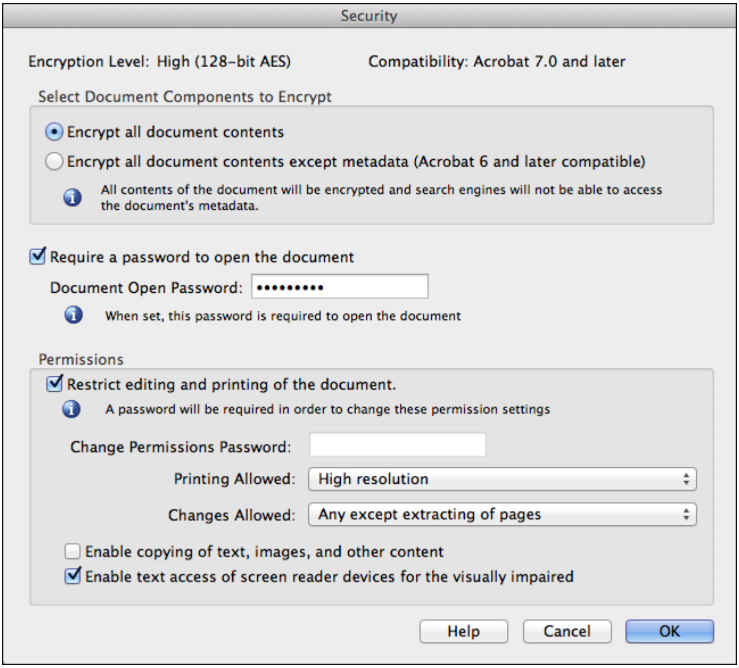
Note: It is not recommended to edit the “SETTINGS” file manually because this can introduce syntax errors.

Edit Distiller Security

You may want to protect your created PDF documents against unauthorized use, e.g. from being opened, printed, or changed.

The different types of protection can be set in any Acrobat Distiller:

➢ Start Acrobat Distiller and go to `Settings > Security...`. In the "Security" window (Fig. 8.3) specify the required options, and finally confirm your entries with the `OK` button.



Security

Encryption Level: High (128-bit AES)   Compatibility: Acrobat 7.0 and later

Select Document Components to Encrypt

● Encrypt all document contents

○ Encrypt all document contents except metadata (Acrobat 6 and later compatible)

All contents of the document will be encrypted and search engines will not be able to access the document's metadata.

☑ Require a password to open the document

Document Open Password: •••••••••

When set, this password is required to open the document

Permissions

☑ Restrict editing and printing of the document.

A password will be required in order to change these permission settings

Change Permissions Password:

Printing Allowed: High resolution

Changes Allowed: Any except extracting of pages

□ Enable copying of text, images, and other content

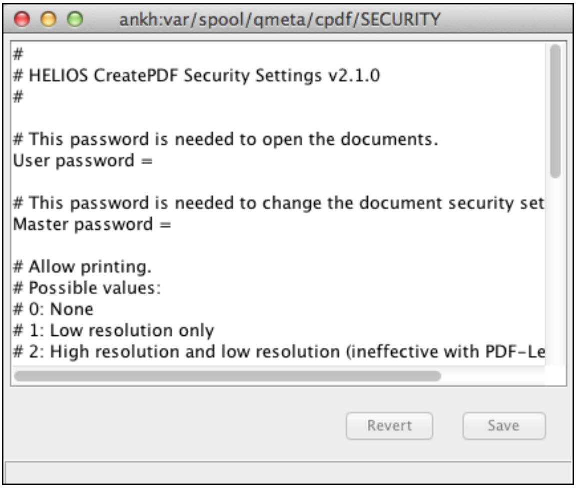
☑ Enable text access of screen reader devices for the visually impaired

Fig. 8.3: Acrobat Distiller "Security" dialog

Likewise, Create PDF Server can tag PDF documents, during the generation, with security settings which restrict the opening, printing, changing (i.e. editing) of a PDF document.

Create PDF Server stores the security information in “HELIOSDIR/var/spool/qmeta/<printer>/SECURITY”. To use security settings with Create PDF Server, proceed as follows:

➢ Start HELIOS Admin and select the desired “Create PDF” printer queue. Open the security settings window (Fig. 8.4) of the respective “Create PDF” printer queue by clicking the `Edit Distiller Security...` button in the `Interface` tab of the printer queue configuration window.



```
#
# HELIOS CreatePDF Security Settings v2.1.0
#

# This password is needed to open the documents.
User password =

# This password is needed to change the document security set
Master password =

# Allow printing.
# Possible values:
# 0: None
# 1: Low resolution only
# 2: High resolution and low resolution (ineffective with PDF-Le
```

Fig. 8.4: Select “SECURITY” file

➢ Apply your changes in the “SECURITY” window, then close the window and confirm your changes.

Create PDF Server will produce PDF files which behave according to the specifications in the “SECURITY” file.

Note: If no security settings are specified in the “SECURITY” file, Create PDF Server does not put up any restrictions. However, if security settings are defined in the used Distiller they will be applied! So make it a habit to remove all PDF security related restrictions in the Distiller settings, and apply such restrictions via Create PDF Server only.

8.2.2 Composite PDF without color space transforms

When printing composite PostScript to a “Create PDF” queue, if color matching is turned off (i.e. the `Default Printer Profile` in the HELIOS Admin tab `ICC` is set to `None`), then no color matching will take place, and all objects will be sent in their original color space to be Distilled. However, color space conversion may still occur if the Distiller “Adobe PDF Settings” are set to do so.

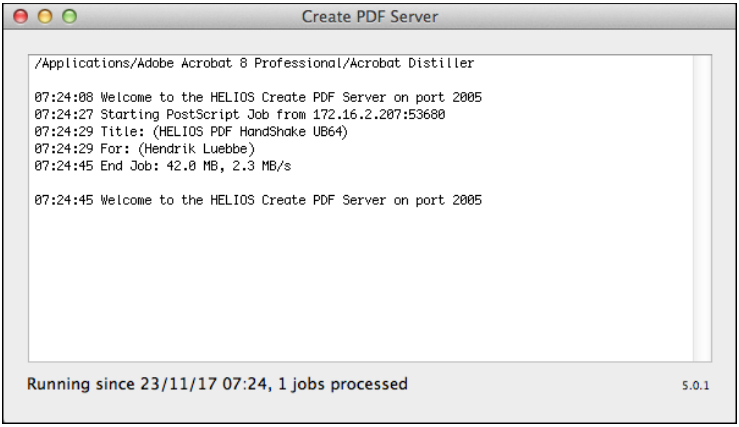
8.3 Print to the “Create PDF” printer queue

Printing to a “Create PDF” printer queue is almost the same as printing to any other HELIOS printer queue. The only difference is that here a Mac or Windows host with Acrobat Distiller acts as a “Create PDF” printer. So any (authorized) user on the network can benefit from the functionality of Create PDF Server.

➢ In the application from which you want to print, open the print menu and choose the “Create PDF” printer queue. There, specify the required settings (e.g. paper size and PostScript options) and start the job.

Create PDF Server then starts processing the jobs – one at a time – according to their order in the spool queue. You can monitor the process and the state of each print job in the “Create PDF Server” window.

While idling, the window states with a "Welcome" message that the server is ready to receive print jobs. When a print job is being processed the window displays information about where the job comes from and via which port as well as the total print job size and the transfer rate. After Create PDF Server has completed the job and the PDF file has been created, the window shows the size of the PDF file and the transfer rate (Fig. 8.5).



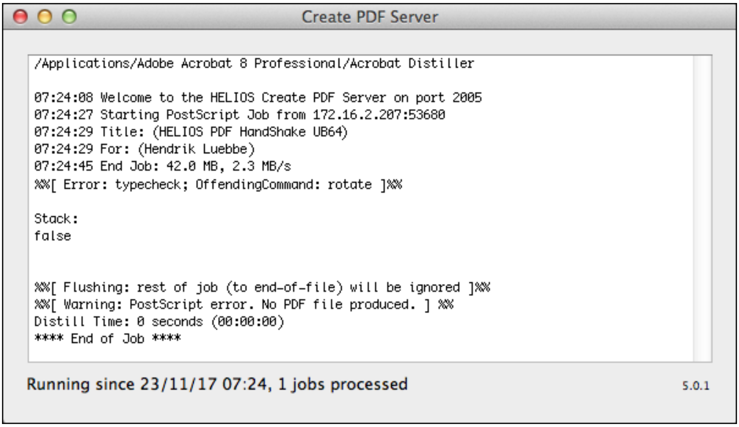
```
/Applications/Adobe Acrobat 8 Professional/Acrobat Distiller

07:24:08 Welcome to the HELIOS Create PDF Server on port 2005
07:24:27 Starting PostScript Job from 172.16.2.207:53680
07:24:29 Title: (HELIOS PDF HandShake UB64)
07:24:29 For: (Hendrik Luebbe)
07:24:45 End Job: 42.0 MB, 2.3 MB/s

07:24:45 Welcome to the HELIOS Create PDF Server on port 2005
```

Running since 23/11/17 07:24, 1 jobs processed

5.0.1

Fig. 8.5: "Create PDF Server" window

If there should arise a problem during the print processing, a warning message appears in the "Create PDF Server" window specifying the cause of the problem (Fig. 8.6).



```
/Applications/Adobe Acrobat 8 Professional/Acrobat Distiller

07:24:08 Welcome to the HELIOS Create PDF Server on port 2005
07:24:27 Starting PostScript Job from 172.16.2.207:53680
07:24:29 Title: (HELIOS PDF HandShake UB64)
07:24:29 For: (Hendrik Luebbe)
07:24:45 End Job: 42.0 MB, 2.3 MB/s
%%[ Error: typecheck; OffendingCommand: rotate ]%%

Stack:
false


%%[ Flushing: rest of job (to end-of-file) will be ignored ]%%
%%[ Warning: PostScript error. No PDF file produced. ] %%
Distill Time: 0 seconds (00:00:00)
**** End of Job ****
```

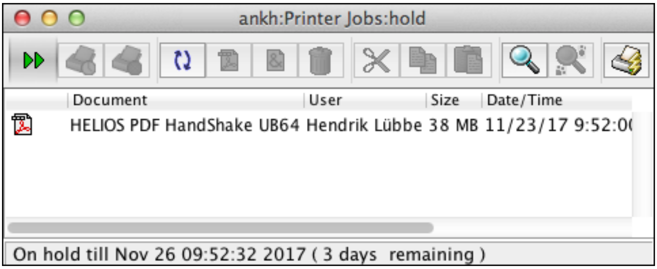
Running since 23/11/17 07:24, 1 jobs processed

Fig. 8.6: “Create PDF Server” window

After that, Create PDF Server is idle again and ready to accept new print jobs.

Obtain the PDF file

There are two ways to obtain the printed PDF file: from the “hold” queue of HELIOS Admin, provided that the “Create PDF” printer queue is configured to pass printed jobs to the “hold” queue.



Document	User	Size	Date/Time
HELIOS PDF HandShake UB64	Hendrik Lübbe	38 MB	11/23/17 9:52:0(

On hold till Nov 26 09:52:32 2017 (3 days remaining)

Fig. 8.7: “Create PDF” print job on the “hold” queue

➢ Double-click the “hold” queue in the `Printers` list to access the file in the job window (Fig. 8.7). The PDF icon in front of the file name indicates that PDF generation has been successful. Double-clicking a job’s PDF icon in the “hold” queue automatically starts your Acrobat application. Also, you can select the job in the “hold” queue window and go to `Save as...` in the `File` menu of HELIOS Admin.

You can also obtain the created PDF file via the HELIOS volume (which should be described by the path in the `Directory` entry) and get access to the file there or, if you checked `Create Folder for Each User`, in the directory’s subfolder.

Note: Please note that the PDF file still remains in the HELIOS directory even if the print job is erased from the “hold” queue. This is because the printer queue only references the PDF file.

8.3.1 Balance “Create PDF” print jobs (Balance queue)

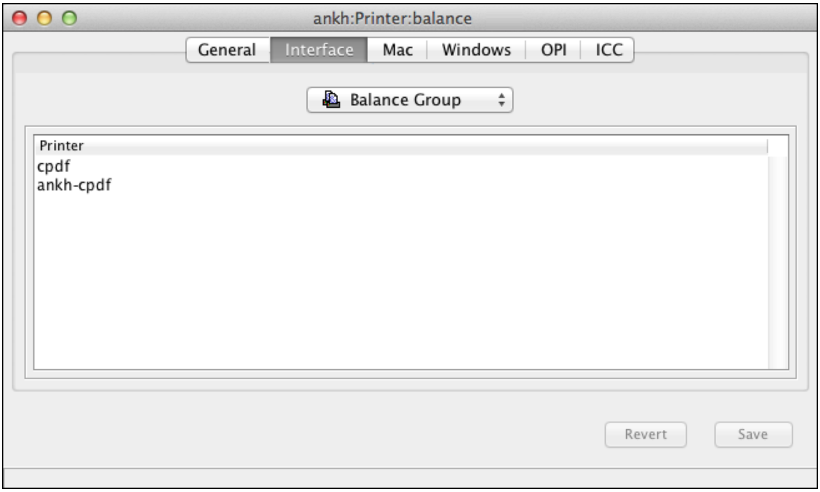
To enhance the data throughput you may group two or more “Create PDF” queues in one balance queue. The user then prints to this “balance” queue where the print server distributes the job to the next idle “Create PDF” queue.

Because the PDF job options and security settings are sent along with the print job, the print job can be sent to any of the Create PDF Server hosts.

For comprehensive and more detailed information about how to set up a balance queue, see the respective chapters in the HELIOS Base manual.

➢ Activate the `Printers` list and choose `New` from the `File` menu. HELIOS Admin then opens a new printer data window, which – to some extent – is already filled out with default values from the “Server Settings” window.

➢ Fill out the `Printer Name`, select a PPD, and if desired, assign “hold” and “error” queues. In the `Connection` pop-up menu select `Balance Group`. From the Printers list drag the “Create PDF” queues you want to group for load balancing into the Printer field (Fig. 8.8). To finish, enter the TCP and SMB names in the Mac and Windows tabs, and click Save.



Printer
cpdf
ankh-cpdf

Fig. 8.8: Grouping “Create PDF” queues for balance queue

Important: It is most important that the “Create PDF” printer queues which form the “Balance Group” have been assigned the same Distiller and Security settings (see **Select Distiller Settings** and **Edit Distiller Security**). Also select the same PPD for the “Balance Queue” which is already assigned to the “Create PDF” printer queues in the Printers field. Otherwise the printing results may not be predictable.

8.4 Possible problems

Acrobat Distiller sometimes produces strange error messages if it is running out of memory. Sometimes there is an error message *no PDF file created* or a PostScript error message. Please verify the failed PostScript job by dropping it on the Distiller application icon, maintaining the same Distiller `Settings` and `Memory` allocation.

If no PDF file is created the reasons could be:

- PostScript errors in the print job
- Incomplete Acrobat Distiller installation
- Missing fonts

The status in the "Printer" window issues "Waiting for printer…". The possible cause is:

- Host or Create PDF Server is not up and running or wrong IP address in printer configuration

9 Use PDF files in a PostScript OPI workflow

PDF HandShake is tightly integrated with other HELIOS products. When PDF HandShake is used together with ImageServer, many additional options become available. One option is the use of PDF files in a traditional PostScript OPI workflow. Another option, **Export for Imposition in Acrobat**, is described in 10 “Export for Imposition with ImageServer”. A third option is the PDF-native OPI workflow, which entails OPI references and image replacement within PDF files, without printing to PostScript. See 11 “PDF-native OPI workflow”.

Placing PDF documents directly into page layouts is not fully supported by many page layout applications. Using PDF HandShake with ImageServer overcomes this limitation, by automatically creating low-resolution EPS “layout” files from the PDF, for placement in a page layout document. At print time, the OPI server replaces each layout image with the original high-resolution PDF page residing on the server.

In addition, layout images (of high-resolution images or PDF documents) can be placed in documents that are saved as PDF. When the resulting PDF file is printed to a HELIOS printer queue with OPI active, once again, the low-resolution layout images will be replaced with the corresponding high-resolution originals.

Using PDF HandShake together with ImageServer enables in-RIP separation workflows using DeviceN color spaces for color objects with arbitrary combinations of CMYK process colors and spot colors.

9.1 Tag PDF files

PDF files can contain a multitude of objects (images, fonts, line art, etc.) in a variety of color spaces (RGB, CMYK, CIE Lab, etc.). In order to successfully print a PDF file, all of the content must, at some point, be converted into the color space of the output device. PDF HandShake does include a high quality CMM (*Color Management Module*) for ColorSync ICC based color space transforms. If necessary, PDF HandShake can perform on-the-fly color space transforms to provide proper color matched output for a given device at print time. This on-the-fly conversion has the advantage that the original PDF document is never modified.

In order to achieve optimal color space conversions, it is important to use the appropriate source ICC profiles (an output profile is specified for each printer queue). Many PDF documents do already include ICC profiles, and these can be used by PDF HandShake. In cases where document ICC (source) profiles are not included with a PDF file, the `Default RGB Profile` and `Default CMYK Profile` that have been defined in HELIOS Admin will be used. See 5 “Set up PDF HandShake with HELIOS Admin”. When printing, these default profiles can be overridden if desired by means of the `PDF HandShake Print` dialog, or via the “pdfprint” command.

When PDF HandShake is used together with ImageServer, the “HELIOS ICC Tagger” program can also be used to tag (i.e. assign or change) source profiles to a PDF document or to image files. If you are not yet familiar with the “HELIOS ICC Tagger” program, you should read the corresponding chapter in the HELIOS ImageServer manual first.

The “HELIOS ICC Tagger” program is located in the “MacOS > ImageServer Tools” folder in the “HELIOS Applications” volume.

Display tagging information

If you choose `Open Tag Info` from the `File` menu, you can drag and drop several files at the same time into the window, as shown in Fig. 9.1.

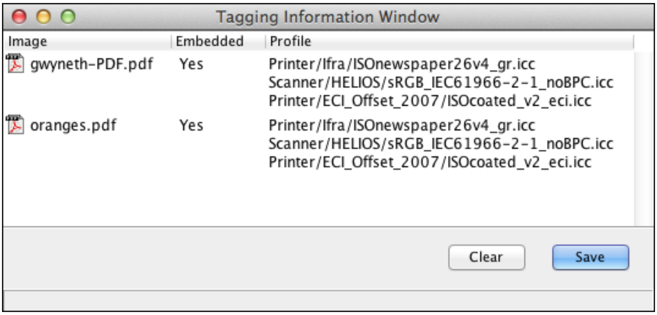


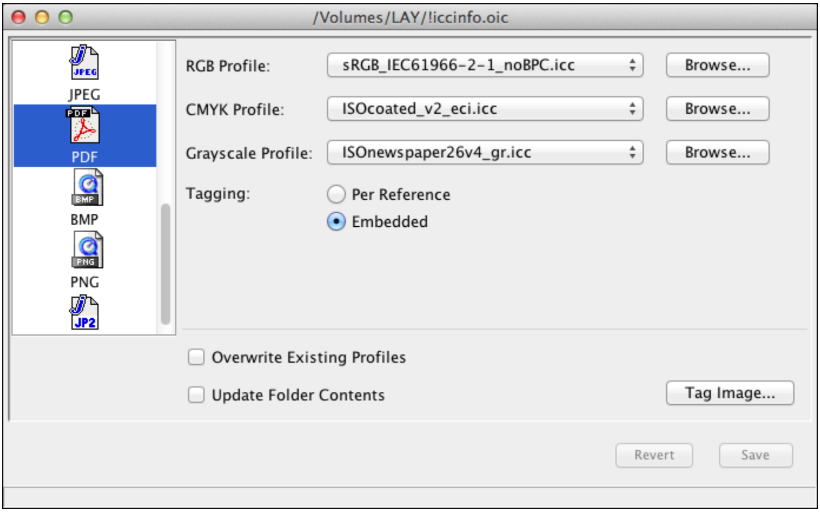
Image	Embedded	Profile
gwyneth-PDF.pdf	Yes	Printer/Ifra/ISOnewspaper26v4_gr.icc Scanner/HELIOS/sRGB_IEC61966-2-1_noBPC.icc Printer/ECI_Offset_2007/ISOcoated_v2_eci.icc
oranges.pdf	Yes	Printer/Ifra/ISOnewspaper26v4_gr.icc Scanner/HELIOS/sRGB_IEC61966-2-1_noBPC.icc Printer/ECI_Offset_2007/ISOcoated_v2_eci.icc

Fig. 9.1: Entries in the „Tagging Information Window“

The buttons in the information window serve to either delete (`Clear`) all entries or save them into a text file (`Save`) for further usage.

Tagging information of an image and the content of the “!iccinfo.oic” file can also be displayed by selecting the object via the `File > Open` menu.

If you wish to create a new ICC info file (for automatic tagging of files), please follow the instructions given in the corresponding chapter of your ImageServer manual. Note that the tagging dialog includes the PDF file format (Fig. 9.2). Unlike “pure” images, such as TIFF or Photoshop files, PDF documents may contain several color spaces. Therefore you should always specify an RGB, a CMYK *and* a Grayscale profile for the PDF file format.



RGB Profile: sRGB_IEC61966-2-1_noBPC.icc

CMYK Profile: ISOcoated_v2_eci.icc

Grayscale Profile: ISOnewspaper26v4_gr.icc

Tagging: ○ Per Reference ● Embedded

□ Overwrite Existing Profiles

□ Update Folder Contents

Fig. 9.2: Setting up tagging instructions for PDF files

9.2 About the handling of PDF files (OPI server)

9.2.1 Automatic layout generation

To enable the layout generation for PDF high-resolution files, activate the following options in HELIOS Admin:

- `Create Layouts` in the volume settings window (`OPI` tab)
- `PDF OPI Layout Generation` in *PDF HandShake Settings > PDF Layout*

If both options are activated, the OPI server will automatically generate layouts for all PDF files that are saved in the HELIOS server volume. The specified parameters in the `ImageServer Settings` dialog will then be valid for PDF files as well, e.g. layout resolution. See the corresponding chapter in the HELIOS ImageServer manual for details.

By default, the file format of the PDF layout representations will be EPSF for composite PDF files and DCS for pre-separated PDF files.

The OPI server uses the first page of the PDF documents only to generate layouts. If you have a multi-page document and want to have layouts from several pages, you can use our "pdfcat" tool to split the document into several single-page files. Find more information in 6.1 "pdfcat".

PDF files may contain several images, illustrations and text on one page, meaning that the pages could become very complex. Depending on the `Layout Delay` time you have specified in the `OPI Server Settings`, and on how complex the given PDF files are, the generation of layouts can take some time (in some cases even more than a minute).

9.2.2 PDF-native OPI replacement

This PDF HandShake functionality replaces layout images directly in the PDF document. This is done without the need to convert the PDF document into PostScript first, and then back to PDF. The PDF itself remains unchanged because only the images within the document are replaced. Some more features of PDF OPI resolve are:

❍ Transparencies are preserved

❍ Metadata (e.g. job tickets, XMP) is preserved

❍ It is fast and reliable

Note: PDF-native OPI resolve requires HELIOS ImageServer to be installed and licensed on the same host.

All settings are configurable by an OPI printer queue:

❍ ICC profiles and output parameters

PDF OPI can be used via command-line tools or "hot folders" to repurpose an InDesign low-resolution PDF job, so that there is no need for multiple exports of an InDesign document for different needs, e.g.:

- Optimized for press
- Optimized for digital print
- Optimized as 144 dpi RGB color matched customer proof
- Optimized for archiving with JPEG 2000 image compression

See 11 "PDF-native OPI workflow" for further details.

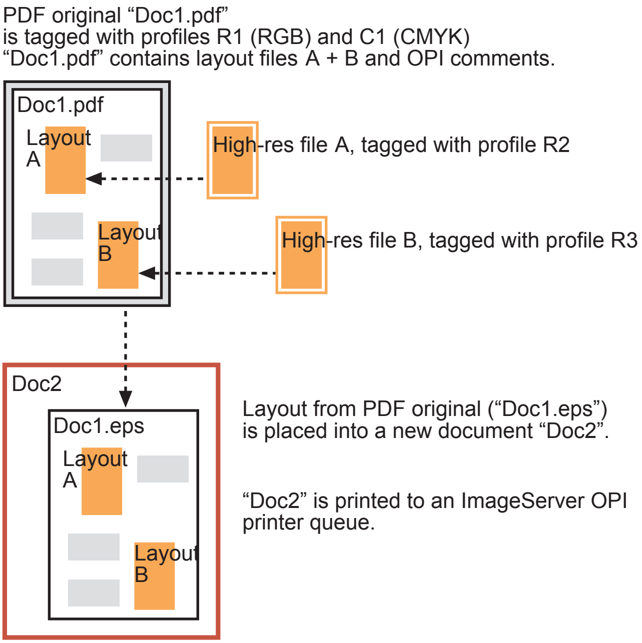
9.2.3 Resolve OPI comments in PDF files

PDF files can already contain OPI comments. For example, if you design an InDesign document, import some layout images (by reference) into this document, and then print into a PostScript file, the resulting PostScript file contains OPI comments referencing the high-resolution images. The OPI comments can be preserved when distilling the PostScript – depending on the Acrobat Distiller job options you define (see 15 "Create PDF files using Acrobat Distiller").

Whenever OPI comments are included in a PDF file, the OPI server tries to resolve these comments. That means that if you place an EPSF layout file (generated from a PDF original) into a layout application and then print your new document, image replacement will possibly be performed recursively: the OPI server replaces the EPSF layout in your document with the PDF original and replaces the layout files in the PDF document with the respective originals. An example is given in Fig. 9.3 below.

In the example illustrated below, color matching will also be performed on two different layers. All objects of "Doc1.pdf" that will not be replaced again, will be matched according to the profiles R1 and C1 – depending on the color space of the respective object. The files A and B will be matched according to the image profiles that are tagged to the high-resolution originals (here: profiles R2 and R3).

In certain situations, resolving OPI comments in PDF documents can cause problems, especially if you send your PDF documents to another production site. In that case – if the PDF document contains layout images and OPI comments pointing to the originals – make sure that you also send these originals and all the profiles that are needed. Sometimes, it could be wiser to embed everything in the PDF file and switch off the option `Preserve OPI Comments` in the Distiller (see 15 "Create PDF files using Acrobat Distiller").



PDF original “Doc1.pdf”
is tagged with profiles R1 (RGB) and C1 (CMYK)
“Doc1.pdf” contains layout files A + B and OPI comments.

High-res file A, tagged with profile R2

High-res file B, tagged with profile R3

Layout from PDF original (“Doc1.eps”)
is placed into a new document “Doc2”.

“Doc2” is printed to an ImageServer OPI
printer queue.

Fig. 9.3: Resolve OPI comments in PDF high-resolution files

9.2.4 Pre-separated PDF files and ImageServer

Pre-separated PDF documents are recognized by ImageServer. They are handled as follows:

❍ The “layout” and “pdftoeps” server command line programs generate DCS files with default (depending on the server settings) composite previews which are raster-based. In order to change the screen preview resolution, use the `-R` option (see 6.5 “pdftoeps”). The “layout” program creates single file DCS-style layout images while the “pdftoeps” program creates DCS-1 or DCS-2 style multi-file images. The plate file suffixes for CMYK will be *.C*, *.M*, *.M*, and *.K*. Spot color plate files will be assigned other suffixes, namely letters so far unused in alphabetical order. The suffix does not have any relation to the name of the spot color.

❍ Printing separations of documents containing placed layouts of pre-separated PDF files to an OPI printer queue with image replacement is fully supported for all output devices.

Note: Composite printing of documents containing placed layouts of pre-separated PDF files to a Print Preview queue will lead to blank picture boxes in the preview.

9.3 OPI layout images of PDF files

The “layout” program (see the respective chapter in the ImageServer manual) offers some options which relate to OPI layout generation of PDF files.

9.3.1 PDF specific options for the “layout” program

-o PDFPageBox <str:"CropBox">

This option can be used to override the default PDF preference **PDFPageBox** which is described in 13 “Preferences”.

-o PDFNativeWorkflow <boolean:FALSE>

Enforces the generation of PDF-native workflow compatible layout images from vector-based EPS images, even if TIFF layout images are supposed to be generated.

-o ImageProfilePaths <strlist:"">

If PDF input files are not yet tagged with ICC profiles, this parameter can be set for attaching a profile to the RGB, CMYK or Grayscale objects within the document. Each string has the format `<color space>=<Pathname>` and defines the ICC profile path name for the given color space. This setting overrides the default profiles in the HELIOS Admin `PDF HandShake Settings`.

-o PDFTransparency <boolean:FALSE>

This option can be used to override the default PDF preference **PDFTransparency** which is described in 13 “Preferences”.

The following conversion options need to be called with the `-l` option:

-o PureWhite <boolean:FALSE>

This option can be used to override the default PDF preference **PureWhite** which is described in 13 “Preferences”.

-o PureGrays <boolean:FALSE>

This option can be used to override the default PDF preference **PureGrays** which is described in 13 “Preferences”.

-o PureBlack <boolean:FALSE>

This option can be used to override the default PDF preference **PureBlack** which is described in 13 “Preferences”.

-o PureCMY <boolean:FALSE>

This option can be used to override the default PDF preference **PureCMY** which is described in 13 “Preferences”.

-o PureVectorWhite <boolean:FALSE>

This option can be used to override the default PDF preference **PureVectorWhite** which is described in 13 “Preferences”.

-o PureVectorGrays <boolean:FALSE>

This option can be used to override the default PDF preference **PureVectorGrays** which is described in 13 “Preferences”.

-o PureVectorBlack <boolean:FALSE>

This option can be used to override the default PDF preference **PureVectorBlack** which is described in 13 “Preferences”.

-o PureVectorCMY <boolean:FALSE>

This option can be used to override the default PDF preference **PureVectorCMY** which is described in 13 “Preferences”.

9.3.2 Attributes for “layout” options

The attributes are entered as command line options using the `layout -o Attributes=<attributes>=<value>` syntax. Additional attributes are delimited by commas.

AntiAlias <boolean:TRUE>

This PDF only attribute smooths the screen preview. It can be used to override the default PDF preference **AntiAlias** which is described in 13 “Preferences”.

PageNumber <uint32:1>

This attribute lets you specify a certain page of a PDF document for the layout process.

TextQuality <str:"high">

Determines the text rendering quality when PDF is converted to raster images. It can be set to override the PDF preference **TextQuality** which is described in 13.2 “PDF manager preferences”.

10 Export for Imposition with ImageServer

“Export for Imposition” enables significantly faster utilization of PostScript imposition applications. It works by exporting the PDF document as a series of PostScript placeholder pages. E.g., a 48 page, 48 MB PDF file can be exported to a 70 kB PostScript file or 1 MB with thumbnail images. This exported file can be imposed and printed to the server orders of magnitude faster than otherwise possible. Once the imposed PostScript is printed to an ImageServer print queue, the original PDF pages are restored according to the imposition instructions.

The “Export for Imposition” feature is integrated into the “PDF HandShake” Acrobat plug-in for Mac, and it can also be used by “pdfprint” on the server. The imposition software output must be printed to an ImageServer printer queue in order to perform proper OPI replacement.

10.1 Export for Imposition in Acrobat

“Export for Imposition” can be used in Adobe Acrobat. It is done via the separate `PDF HandShake Export` dialog, which is similar to the one used for `PDF HandShake Print`.

➢ Start your Acrobat application and open the `Export for Imposition` dialog from the `File` menu as shown in Fig. 10.1.

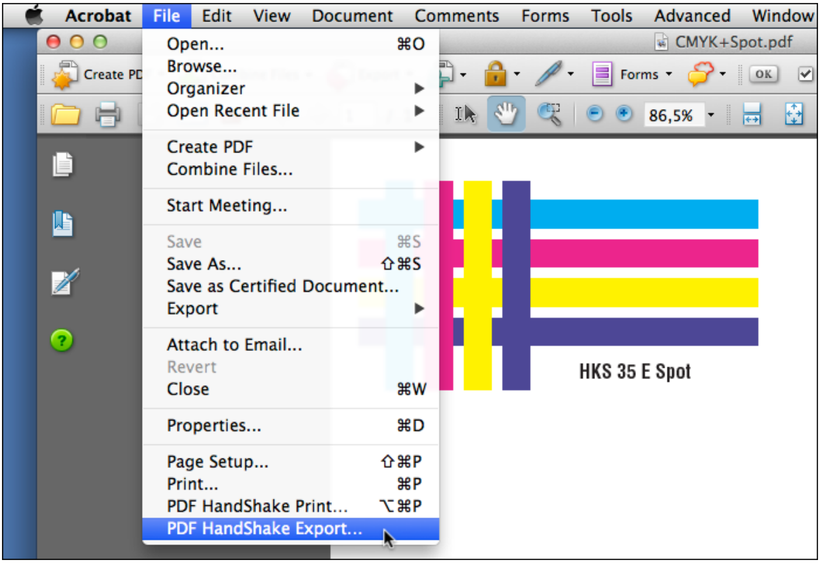
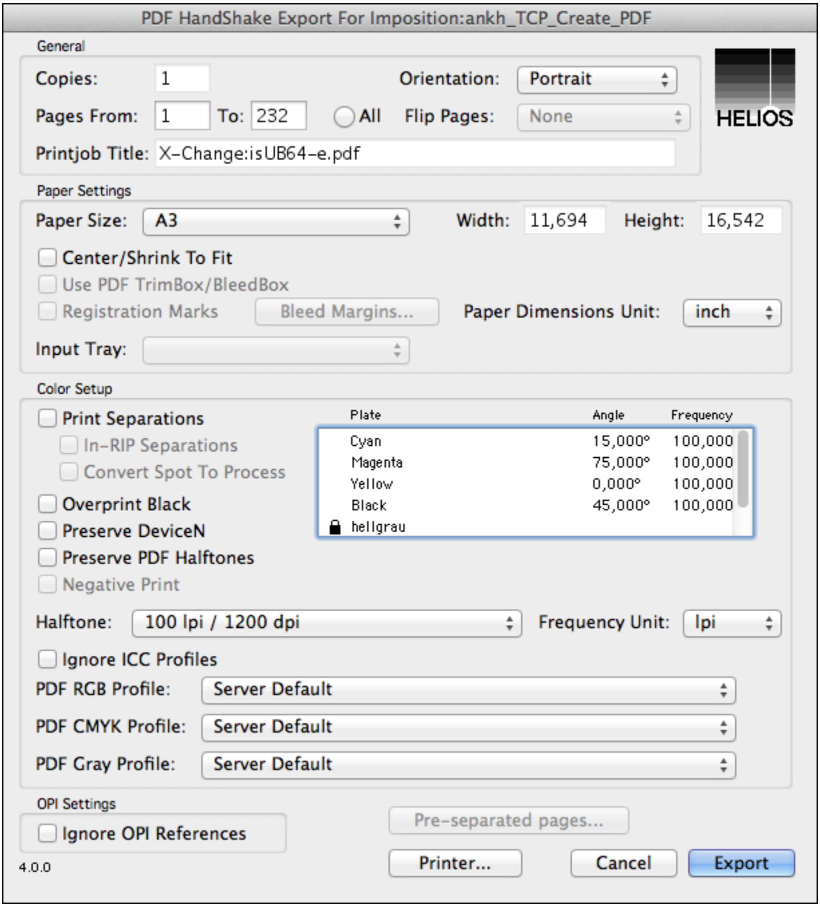


Fig. 10.1: Opening the “Export for Imposition” dialog

Fig. 10.2 shows an example of the export dialog. Note that the dialog – just like the `PDF HandShake Print` dialog – may differ slightly, depending on whether or not your document is pre-separated. Fig. 10.2 shows the dialog as it would appear for a composite PDF document.



PDF HandShake Export For Imposition:ankh_TCP_Create_PDF

General

Copies: 1
Orientation: Portrait
Pages From: 1 To: 232 ○ All
Flip Pages: None
Printjob Title: X-Change:isUB64-e.pdf

Paper Settings

Paper Size: A3
Width: 11,694
Height: 16,542
□ Center/Shrink To Fit
□ Use PDF TrimBox/BleedBox
□ Registration Marks
Paper Dimensions Unit: inch
Input Tray:

Color Setup

□ Print Separations
□ In-RIP Separations
□ Convert Spot To Process
□ Overprint Black
□ Preserve DeviceN
□ Preserve PDF Halftones
□ Negative Print

Plate	Angle	Frequency
Cyan	15,000°	100,000
Magenta	75,000°	100,000
Yellow	0,000°	100,000
Black	45,000°	100,000
🔒 hellgrau		

Halftone: 100 lpi / 1200 dpi
Frequency Unit: lpi
□ Ignore ICC Profiles
PDF RGB Profile: Server Default
PDF CMYK Profile: Server Default
PDF Gray Profile: Server Default

OPI Settings

□ Ignore OPI References

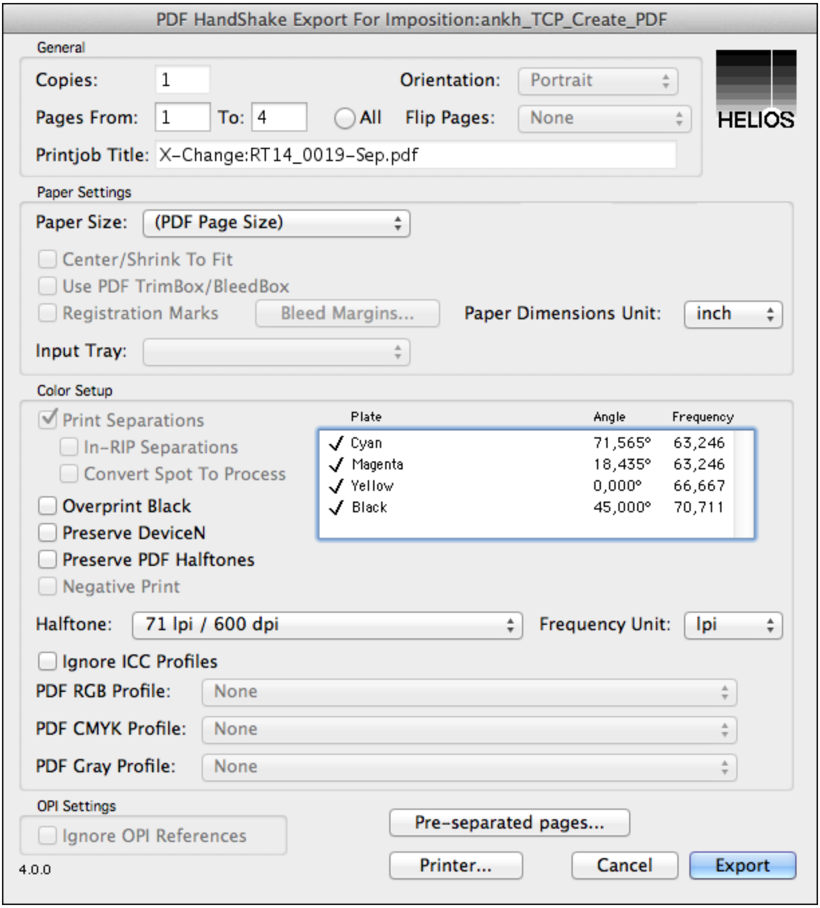
4.0.0

Fig. 10.2: The “Export for Imposition” dialog (composite file)

The options `Flip Pages`, `Input Tray`, `Registration Marks`, `Use PDF TrimBox/BleedBox`, and `Negative Print` are disabled because it makes much more sense to set these options in the imposition software later before the whole document is sent to the final output device. Neither `Offset` nor `Gap` are available for custom page sizes (`Custom`).

The `Paper Size` pop-up menu contains the entry `(PDF Page Size)`. This entry can be selected to make sure that the pages in the exported PostScript file are exactly as large as the PDF pages in the document. If you choose `(PDF Page Size)`, the `Orientation` and `Center/Shrink To Fit` options (Fig. 10.3) are disabled.

For pre-separated files – as shown in Fig. 10.3 – `Print Separations` is switched on automatically. `In-RIP Separations` and `Convert Spot To Process` as well as color matching, are disabled. This is equivalent to the settings in the `PDF HandShake Print` dialog for pre-separated documents.



PDF HandShake Export For Imposition:ankh_TCP_Create_PDF

General

Copies: 1

Orientation: Portrait

Pages From: 1 To: 4 ○ All

Flip Pages: None

Printjob Title: X-Change:RT14_0019-Sep.pdf

Paper Settings

Paper Size: (PDF Page Size)

□ Center/Shrink To Fit

□ Use PDF TrimBox/BleedBox

□ Registration Marks

Paper Dimensions Unit: inch

Input Tray:

Color Setup

☑ Print Separations

□ In-RIP Separations

□ Convert Spot To Process

□ Overprint Black

□ Preserve DeviceN

□ Preserve PDF Halftones

□ Negative Print

Plate	Angle	Frequency
✓ Cyan	71,565°	63,246
✓ Magenta	18,435°	63,246
✓ Yellow	0,000°	66,667
✓ Black	45,000°	70,711

Halftone: 71 lpi / 600 dpi

Frequency Unit: lpi

□ Ignore ICC Profiles

PDF RGB Profile: None

PDF CMYK Profile: None

PDF Gray Profile: None

OPI Settings

□ Ignore OPI References

4.0.0

Fig. 10.3: The “Export for Imposition” dialog (pre-separated file)

The button `Pre-separated pages ...` in the export dialog opens a window that allows you to check the separation plate information of all pages in your document.

10.2 Use the export option in “pdfprint”

The “pdfprint” command line program can also be used to export files instead of printing them:

```
pdfprint -E [... other options] <PdfFilename> <PSFilename>
```

If `<PSFilename>` is not specified, the export file will be sent to “stdout”.

Some options are only available for printing but cannot be exported. If you specify one of these options by mistake, they will be ignored by “pdfprint”. This applies to the following parameters:

- `-o flip=`
- `-o negativeprint`
- `-o gap=`
- `-o offset=`
- `-o usetrimbox`
- `-o usebleedbox`
- `-I (Input Tray)`
- `-M (Registration Marks)`

If you *do not* set the `-p` parameter, the “PDF Page Size” will be applied by default. All pages in the exported PostScript file will then be exactly as large as the pages in the PDF document.

10.3 Features of the exported document

The PostScript files that are generated by “Export for Imposition” only contain references to the PDF document and to optionally selected ICC profiles. In your imposition software, you will at least see a black frame with the file name, page number, and plate name (for pre-separated files only) of the PDF page. If the PDF document contains thumbnail images (created in Acrobat), these are also included in the PostScript file.

When the PostScript file or the imposed print job is printed to an ImageServer print spooler, the original high-resolution pages are inserted. The PDF document and the optional profiles must reside in a HELIOS volume to be found by the OPI system. The usual OPI image search strategies apply.

10.4 Imposition software products

Until now, the following imposition software packages have been successfully tested with the PDF HandShake "Export for Imposition" feature:

- Imation PressWise
- INposition (for QuarkXPress)
- ScenicSoft Preps
- Farrukh Imposition Publisher
- Heidelberger Signa
- STRIP IT
- ULTIMATE IMPOSTRIP (with Adobe DSC settings)

11 PDF-native OPI workflow

HELIOS ImageServer is well-known as a reliable, colortrue and fast OPI workflow solution. For more information about OPI workflows see the ImageServer manual. All classic OPI workflows use the print function of page layout applications (e.g. InDesign) to produce PostScript output with OPI references. These OPI references are placeholders for high-resolution images. To obtain a full-resolution document, e.g. for the generation of a printing plate, the OPI references within the PostScript output are resolved, i.e. replaced with high-resolution images. Then this resolved PostScript output is used for further processing with a trapping, imposition or rasterizing solution.

The basic idea behind PDF HandShake's new PDF-native OPI workflow is to use the InDesign or QuarkXPress[1] *PDF export* function to produce PDF output with OPI references instead of PostScript output with OPI references. As mentioned above, to obtain a full-resolution document, the OPI references within the exported PDF document are resolved, i.e. replaced with high-resolution images, and the result is a new PDF document containing the high-resolution images. The command line utility "pdfresolve" (see 6.7 "pdfresolve") provides this new OPI image replacement functionality for PDF documents. This functionality can also be used by means of a hot folder utilizing the Script Server example script "pdfresolve.pl".

For more information on Script Server and the example script "pdfresolve.pl" see the ImageServer manual. This workflow needs neither PostScript documents nor conversions from PDF to PostScript or vice versa. The development of "pdfresolve" follows the current trend in workflow design to replace PostScript with PDF.

[1] QuarkXPress 3-9

One major benefit of using PDF instead of PostScript is that it offers many features that can be expressed in PDF only but not in PostScript. In a classic PostScript based OPI workflow, much valuable information like transparency, ICC profiles, PDF/X output intents, metadata, and trapping information cannot be embedded within the document file. It is a difficult and nearly impossible task to store and maintain this information elsewhere, and once this information is lost somewhere within the workflow, it is often gone for good. Another major advantage of PDF over PostScript is that PDF viewing applications are available for many platforms whereas viewing PostScript accurately enough for prepress needs is a sophisticated task which involves high software costs.

11.1 Appropriate file formats for PDF-native OPI

You can place both high-resolution and layout images. When generating layout images via ImageServer, you can choose between EPSF and TIFF layouts. It is strongly recommended not to use high-resolution images and to generate TIFF only layouts in a PDF-native OPI workflow. You can force the recommended generation of TIFF layouts by copying the high-resolution images to a folder in a HELIOS volume with automatic layout generation enabled and using the "%t" folder naming option (see the chapter "Defining folder-specific OPI settings" in the HELIOS ImageServer manual.

What are the benefits of placing layouts instead of high-resolution images? First of all, layouts are nearly always much smaller than high-resolution images, making production possible on a limited bandwidth network or making production substantially faster. This is also the main reason why OPI is used in the first place.

What are the benefits of placing TIFF layouts rather than EPSF layouts? First of all, the current versions of InDesign and QuarkXPress (up to version 9) do not embed the Mac file ID of high-resolution images in the exported PDF documents when using EPSF layouts. Furthermore, ImageServer generates TIFF and EPSF layouts from high-resolution images with clipping paths while preserving these clipping paths, but e.g. InDesign copies the clipping path to the exported PDF documents only if TIFF layouts are used. “pdfresolve” does not insert clipping paths during OPI image replacement either. Hence clipping paths are lost within the PDF-native OPI workflow when using EPSF layouts.

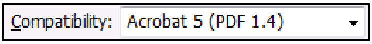
The preference **PDFNativeWorkflow** (see 13.1 “Global PDF preferences”) enforces the generation of PDF-native workflow compatible layout images. So vector-based EPS images will lead to EPS layout images, even if TIFF layout images are supposed to be generated.

11.2 Export PDF

11.2.1 InDesign

When you have loaded a document in InDesign and want to export it as PDF for a PDF-native OPI workflow, you have to specify certain options in the “Export PDF” options dialog.

If you want to use the transparency features, which include drop shadow and feather effects, you must select `Acrobat 5 (PDF 1.4)` or higher from the `Compatibility` pop-up menu (Fig. 11.1) in the “General” settings section of the “Export Adobe PDF” options dialog. Prior versions of PDF do not support transparencies.

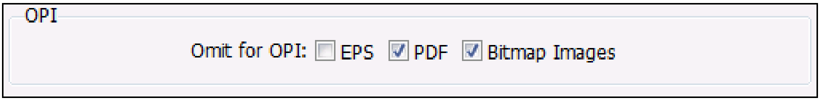


Compatibility: Acrobat 5 (PDF 1.4)

Fig. 11.1: InDesign – Export PDF compatibility

Note: Make sure to activate the option `Crop Image Data to Frames` in the “Compression” settings section.

If you use TIFF layout files only, as recommended, you must check the `Omit for OPI` checkbox for `Bitmap images` in the “Advanced” settings section of the `Export PDF` options dialog (Fig. 11.2):



OPI

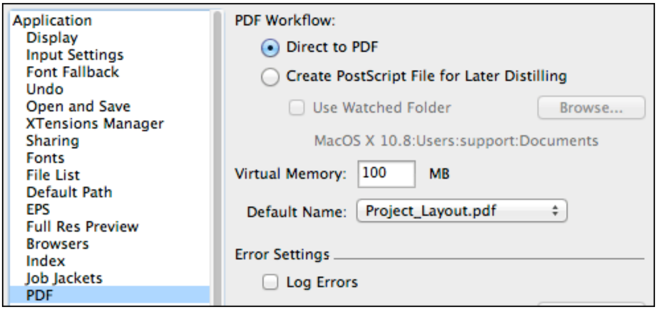
Omit for OPI: □ EPS ☑ PDF ☑ Bitmap Images

Fig. 11.2: InDesign – Export omit bitmap images

If you want to use EPSF layouts despite their disadvantages, you must check the `EPS` checkbox here, too.

11.2.2 QuarkXPress (up to version 9)

First of all, it is recommended to activate the `Direct to PDF` option in the QuarkXPress “PDF Workflow” preferences (Fig. 11.3).



PDF Workflow:

● Direct to PDF

○ Create PostScript File for Later Distilling

□ Use Watched Folder Browse...

MacOS X 10.8:Users:support:Documents

Virtual Memory: 100 MB

Default Name: Project_Layout.pdf

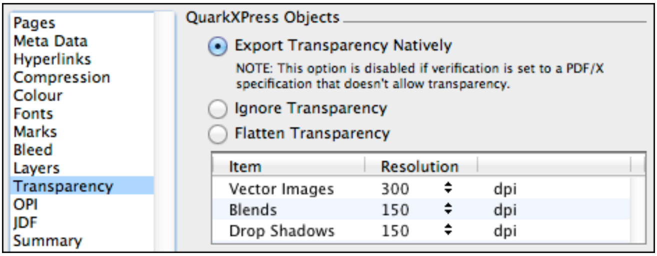
Error Settings

□ Log Errors

Fig. 11.3: QuarkXPress – Preferences

If you want to export a document as PDF for a PDF-native OPI workflow, you have to specify certain options in the “Export as PDF” dialog.

For using the transparency features, which include drop shadow and feather effects, you must select `Export Transparency Natively` in the "QuarkXPress Objects" section (Fig. 11.4).



QuarkXPress Objects

● Export Transparency Natively

NOTE: This option is disabled if verification is set to a PDF/X specification that doesn't allow transparency.

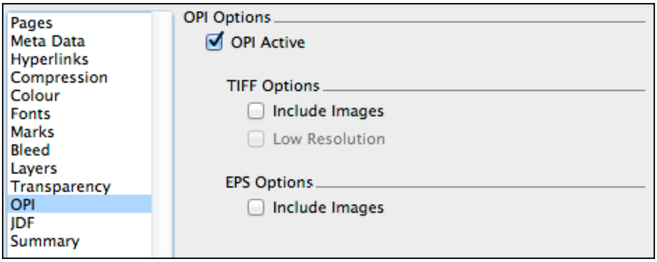
○ Ignore Transparency

○ Flatten Transparency

Item	Resolution	
Vector Images	300	dpi
Blends	150	dpi
Drop Shadows	150	dpi

Fig. 11.4: QuarkXPress – PDF export options

If you use TIFF layout files only, as recommended, you must leave the `Include Images` checkbox unchecked in the "TIFF Options" section (Fig. 11.5):



OPI Options

☑ OPI Active

TIFF Options

□ Include Images

□ Low Resolution

EPS Options

□ Include Images

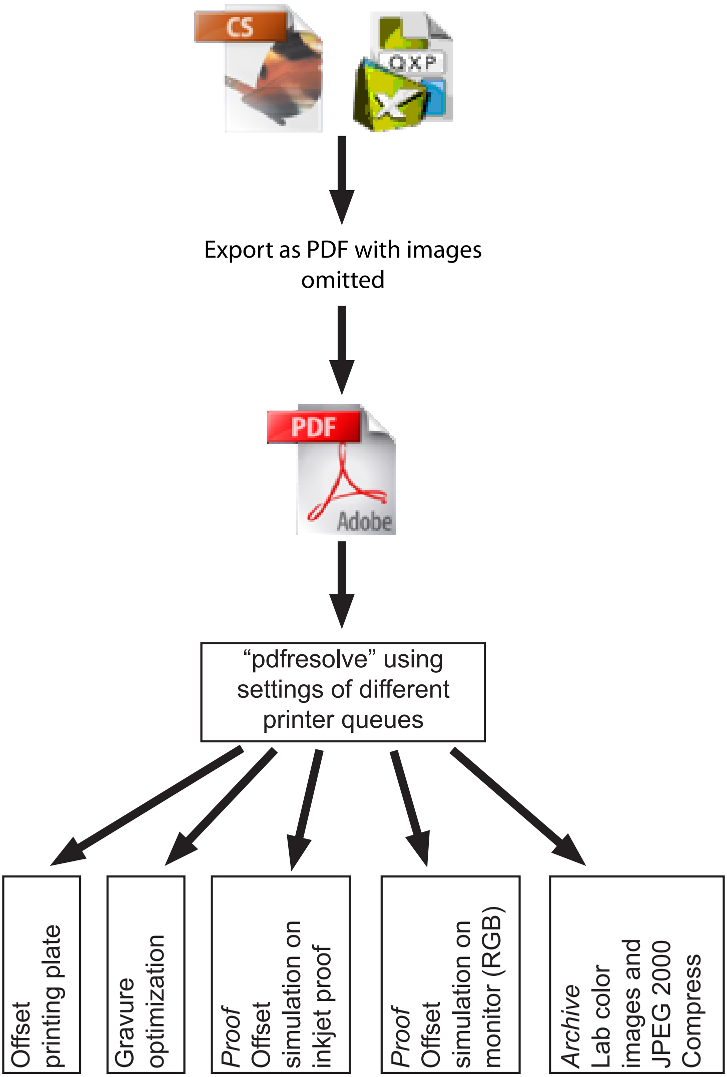
Fig. 11.5: QuarkXPress – Do not include images

11.3 Configure “pdfresolve” via printer queue settings

By default, “pdfresolve” adopts all of its image replacement options from a specified printer queue. Additional command line options override printer queue settings. Printer queue settings can be edited using HELIOS Admin, but two “pdfresolve” options can only be specified via the command line, they cannot be specified in HELIOS Admin. These are the `TagReplacedImages` option and the options to compress replaced images with JPEG 2000 compression format. For more information on the command line options of “pdfresolve” please refer to 6.7 “pdfresolve”. Here is a short overview of the `TagReplacedImages` option:

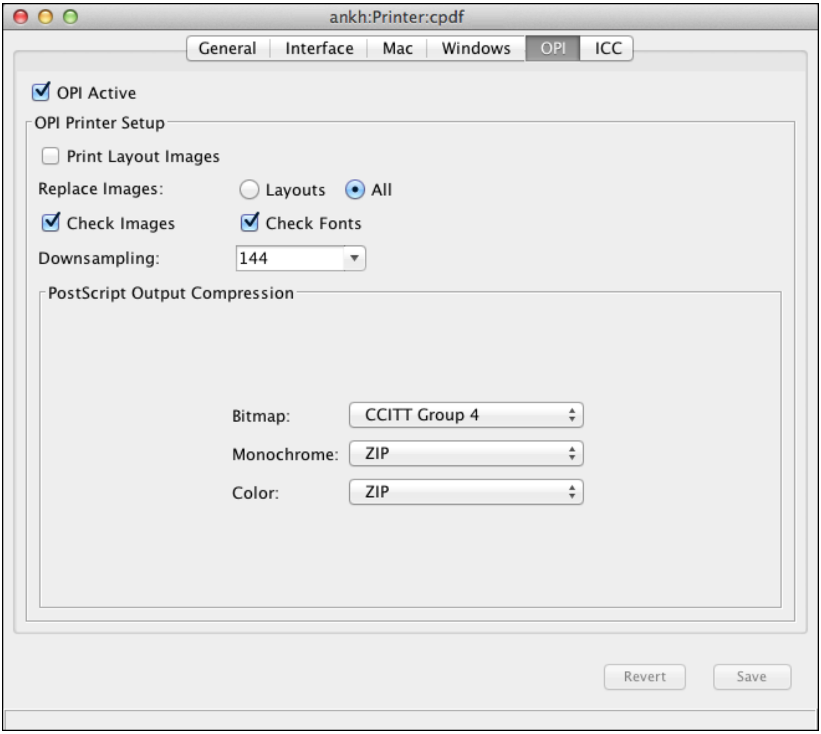
TagReplacedImages	FALSE	TRUE
No printer profile specified	No color matching, no tagging of replaced images	No color matching, tagging replaced images with source image profiles
Printer profile specified	Color matching active, no tagging of replaced images	Color matching active, tagging replaced images with printer or proof profiles

Using “pdfresolve” with settings of different printer queues allows easy generation of PDF output documents for a broad range of applications, from a single PDF document exported by InDesign or QuarkXPress:


CS
QXP
Export as PDF with images omitted
PDF
Adobe
"pdfresolve" using settings of different printer queues
Offset printing plate
Gravure optimization
Proof Offset simulation on inkjet proof
Proof Offset simulation on monitor (RGB)
Archive Lab color images and JPEG 2000 Compress

11.3.1 Downsampling and output compression

In a PDF-native OPI workflow with large high-resolution images it is advantageous to specify downsampling and compression options for the used printer queue. File sizes of resolved PDF documents can be reduced by 50% or more, to substantially shorten printing and other document processing times. These options can easily be specified in HELIOS Admin (Fig. 11.6):



☑ OPI Active

OPI Printer Setup

□ Print Layout Images

Replace Images: ○ Layouts ● All

☑ Check Images ☑ Check Fonts

Downsampling: 144

PostScript Output Compression

Bitmap: CCITT Group 4

Monochrome: ZIP

Color: ZIP

Fig. 11.6: HELIOS Admin – Downsampling and compression

- ➢ In the HELIOS Admin printer queue settings select the `OPI` tab and from the `Downsampling` pop-up menu choose a resolution (in dpi) which will be applied to objects in the PDF document that exceed the specified value.

- ➢ Likewise, specify an output compression method from the `Bitmap`, `Monochrome`, and `Color` pop-up menus.

It is recommended to choose a lossless compression type such as “ZIP” or “CCITT Group 4”.

11.3.2 Hot folder setup

Please refer to the HELIOS ImageServer manual for how to set up the Script Server and how to use the example script “pdfresolve.pl” to set up a PDF-native OPI hot folder.

11.4 Limitations

- ❍ Raster images and PDF documents are supported as high-resolution images only, object-based EPSF documents are not supported
- ❍ Using EPSF layouts leads to reduced high-resolution image file search functionality and loss of clipping paths and transparency, see above
- ❍ Ultimately, all PDF documents need to be rastered, e.g. for the generation of a printing plate. All current RIP solutions can rasterize PDF documents with transparencies, alternatively Adobe Acrobat Professional can be used to print and flatten PDF documents with transparencies

11.5 Remote PDF-native OPI workflows

The basic concept of a remote PDF-native OPI workflow is:

- Generate TIFF layout images from high-resolution images on a server system running ImageServer and PDF HandShake
- Transfer the layout images only to the designers, e.g. via Internet and HELIOS WebShare
- The designers place these layout images in InDesign documents and export the document as PDF, with `Omit for OPI` selected
- The designers transfer the exported PDF documents back to the server system, e.g. via Internet using HELIOS WebShare
- The exported PDF documents are resolved via PDF-native OPI “hot folders” for different applications, e.g. for offset printing plate generation or website publishing

12 PDF-to-PDF color conversion

What is PDF-to-PDF color conversion?

PDF documents are very versatile and are utilized in multiple manners. For example, a PDF product catalog might need to be printed on an offset press and/or on a digital press. In addition, it may be distributed online for reading on computers and mobile devices. But different printing and display devices have different color rendering requirements. So it is often necessary to create multiple versions of a document with the color in each optimized for a different output method. But doing this manually wastes time and is prone to errors.

This is where the HELIOS PDF-to-PDF color conversion solution comes into play. One master PDF document is created and saved into a hot folder (see "HELIOS Script Server" in the HELIOS ImageServer manual), which will automatically produce color-matched versions for different purposes, such as for offset printing, for digital printing, to sRGB for Internet distribution or to the HELIOS profile optimized for iPads.

The HELIOS workflow automation system will extract each object within the PDF document and perform an ICC color space transform into the specified output color space. This allows colors to be rendered consistently across different media. The output is a new PDF where the entire content is color matched to the desired color space.

Fonts, transparencies, images and vector data are fully preserved and the new color matches the desired purpose.

Preparation

➢ First, mount a HELIOS server volume and create a hot folder in which color conversion will take place.

Note: Of course you can create several hot folders, one for each purpose, according to your needs.

We assume three hot folders for three different purposes (see Fig. 12.1):

- One to produce a CMYK PDF optimized for offset printing
- Another to produce color-matched output for iPad displays
- A third hot folder to convert content to the sRGB color space for viewing on other mobile devices and on monitors

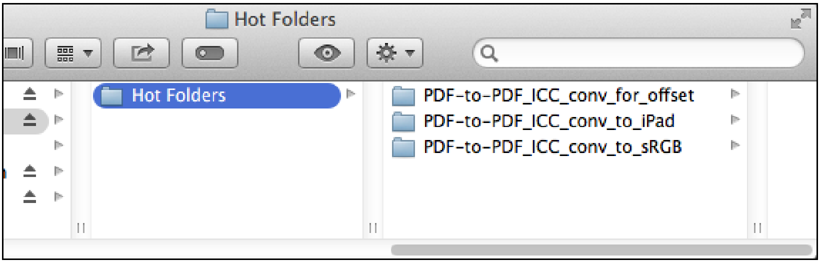

Hot Folders
Hot Folders
PDF-to-PDF_ICC_conv_for_offset
PDF-to-PDF_ICC_conv_to_iPad
PDF-to-PDF_ICC_conv_to_sRGB

Fig. 12.1: Hot folders in a HELIOS server volume

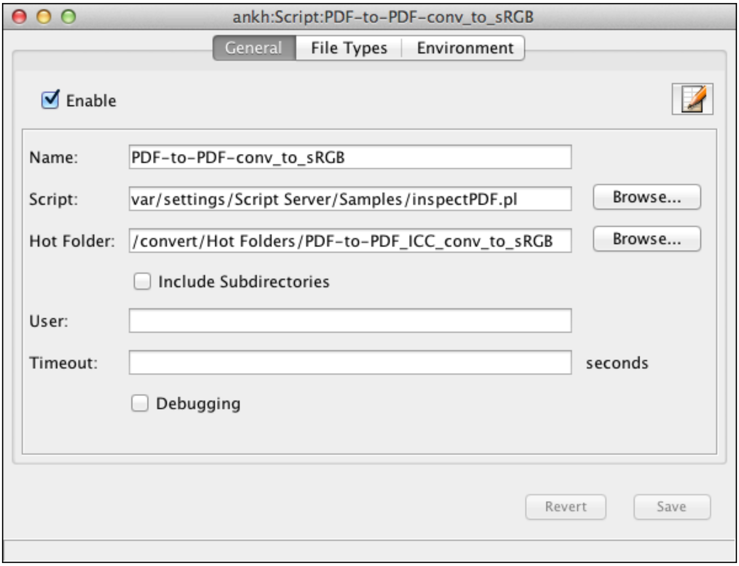
Setting up a hot folder can easily be done with HELIOS Admin:

➢ Activate the `Scripts` tab and choose `New` from the `File` menu.

➢ In the `Name` text field enter a descriptive name.

➢ Then click the `Browse ...` button to select the “inspectPDF.pl” script which is located is the “Settings” volume.

➢ Next, click on `Browse ...` to select the hot folder that you just created (see Fig. 12.2).



ankh:Script:PDF-to-PDF-conv_to_sRGB

General | File Types | Environment

☑ Enable

Name:	PDF-to-PDF-conv_to_sRGB	
Script:	var/settings/Script Server/Samples/inspectPDF.pl	Browse...
Hot Folder:	/convert/Hot Folders/PDF-to-PDF_ICC_conv_to_sRGB	Browse...
	□ Include Subdirectories	
User:		
Timeout:		seconds
	□ Debugging	

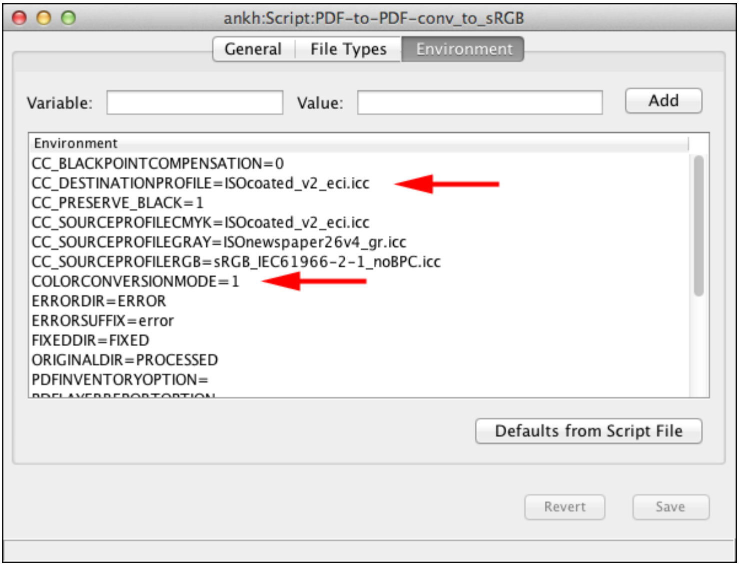
Fig. 12.2: `General` tab

➢ Now click the `Environment` tab.

Two environment variables must be adjusted here:

➢ Double-click the line `COLORCONVERSIONMODE` and set the value from `0` to `1` to enable the color conversion. Then click `Add`.

➢ In the same manner set `CC_DESTINATIONPROFILE` to the appropriate ICC profile for the output device.



```
CC_BLACKPOINTCOMPENSATION=0
CC_DESTINATIONPROFILE=ISOcoated_v2_eci.icc
CC_PRESERVE_BLACK=1
CC_SOURCEPROFILECMYK=ISOcoated_v2_eci.icc
CC_SOURCEPROFILEGRAY=ISOnewspaper26v4_gr.icc
CC_SOURCEPROFILERGB=sRGB_IEC61966-2-1_noBPC.icc
COLORCONVERSIONMODE=1
ERRORDIR=ERROR
ERRORSUFFIX=error
FIXEDDIR=FIXED
ORIGINALDIR=PROCESSED
PDFINVENTORYOPTION=
```

Fig. 12.3: `Environment` tab

In our example we will use the “ISOcoated_v2_eci.icc” ICC profile (see Fig. 12.3).

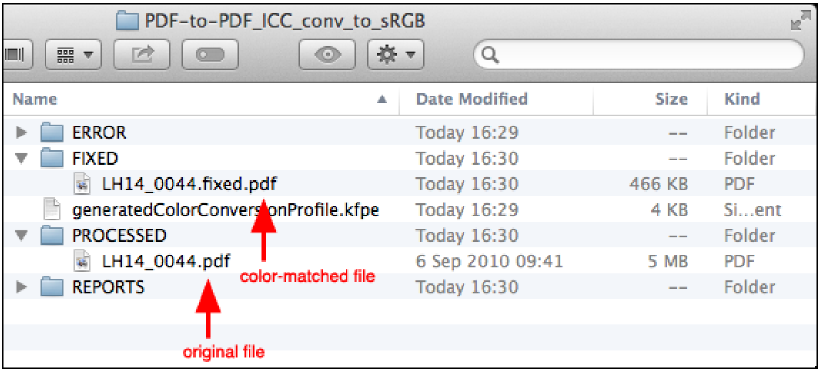
If desired, you can also change the output folder names and the default source ICC profiles.

➢ Repeat the process to create a new hot folder for any other color conversion task.

PDF color conversion

After saving a PDF document into the appropriate hot folder, the conversion is automatically performed. When finished, the hot folder contains additional objects (see Fig. 12.4):

ERROR	Empty, if the conversion was successful
FIXED	Contains the converted PDF file
PROCESSED	Contains the original PDF file
REPORTS	Contains conversion reports in text format and HTML



PDF-to-PDF_ICC_conv_to_sRGB

Name	Date Modified	Size	Kind
ERROR	Today 16:29	--	Folder
FIXED	Today 16:30	--	Folder
LH14_0044.fixed.pdf	Today 16:30	466 KB	PDF
generatedColorConversionProfile.kfpe	Today 16:29	4 KB	Si...ent
PROCESSED	Today 16:30	--	Folder
LH14_0044.pdf	6 Sep 2010 09:41	5 MB	PDF
REPORTS	Today 16:30	--	Folder

color-matched file

original file

Fig. 12.4: Hot folder after successful color conversion

In addition, the “generatedColorConversionProfile.kfpe” file is stored in the hot folder.

HELIOS Admin allows monitoring the conversion progress of all jobs in progress. Details are given in the “HELIOS Script Server” chapter in the HELIOS ImageServer manual.

13 Preferences

This chapter lists all preferences that are pertinent to PDF HandShake. Find a description of how to set, view, change or delete preferences, with the HELIOS utility programs “prefdump”, “prefvalue”, and “prefrestore”, in the HELIOS Base manual.

Important: Make sure that preference keys *DO NOT* start or end with a slash (“/”) character, and note that they are case-sensitive! Also, if any preference key or preference value includes spaces, that key or value must be enclosed in quotes.

13.1 Global PDF preferences

```
Key:  Global/Opi/<preference>
```

PDFNativeWorkflow `bool` `FALSE`

Enforces the generation of PDF-native workflow compatible layout images from vector-based EPS images, even if TIFF layout images are supposed to be generated, e.g. via a “%t” folder syntax.

This key requires a restart of “opisrv” to take effect.

PDFPageBox `str` `"CropBox"`

Specifies the page box of PDF documents that should be used for output. PDF page objects always contain a MediaBox entry and may specify boxes within the MediaBox as CropBox, BleedBox, TrimBox and ArtBox. Valid values of this preference are: `"MediaBox"`, `"CropBox"`, `"BleedBox"`, `"TrimBox"`, `"ArtBox"`.

IgnoreUntagged `bool` `FALSE`

If **DefaultPrinterProfile** is not set or the ImageServer preference `CheckICCProfiles` is set to `FALSE`, this preference has no effect. Otherwise this option determines whether errors are generated for color source images that are not tagged with an ICC profile and which are not excluded from color matching via **PreserveDeviceN**.

AutomaticLayout `bool` `TRUE`

(HELIOS Admin option `PDF OPI Layout Generation`)

Enables the automatic generation of layout files from PDF files in volumes where layout generation is enabled. You can set the preference to `FALSE` if you want PDF files to be excluded from automatic layout generation.

For this preference, the key must be extended to:

```
Global/Opi/Formats/PDF ///AutomaticLayout
```

13.2 PDF manager preferences

```
Key:  Library/OpenImage/ManagerPrivate/HeliosPDF/<preference>
```

AntiAlias `bool` `TRUE`

(HELIOS Admin option `Antialias Screen Preview`)

If set to `TRUE`, the screen preview in layout files, generated from PDF documents, is smoothed to offer an enhanced readability. This is recommended for monitor screen resolutions up to 96 dpi.

RGBProfilePath `str` `"sRGB_IEC61966-2-1_noBPC.icc"`

(HELIOS Admin option `Default RGB Profile`)

Specifies the profile which is used as source profile for color data transformation of all RGB objects in PDF files that have not been tagged with a specific RGB profile.

CMYKProfilePath `str` `"ISO Coated 2 FOGRA39"`

(HELIOS Admin option `Default CMYK Profile`)

Specifies the profile which is used as source profile for color data transformation of all CMYK objects in PDF files that have not been tagged with a specific CMYK profile.

VMSize `uint` `61440000`

This parameter controls the PDF HandShake virtual RIP memory (in bytes) for creating screen previews in layout files from PDF files. You should use the parameter to enter a value higher than the default (60000 kB) – lower values are not recommended.

GVMSize `uint` `10240000`

This parameter controls the PDF HandShake global virtual RIP memory (in bytes) for creating screen previews in layout files from PDF files. You should use the parameter to enter a value higher than the default (10000 kB) – lower values are not recommended.

HairLineWidth `double` `0.25`

Hair lines are lines with a width of zero, meaning the thinnest line that a PostScript device can produce. The results of rendering such “zero-width” lines are device-dependent, and on high-resolution devices they are nearly invisible. If this preference is set to a non-negative value the width of all hair lines in PDF files is set to that value in PostScript points (1/72") for printing and layout generation.

TextQuality `str` `"high"`

Determines the text rendering quality when PDF is converted to raster images. The preference can be set to `high` or `low`. `high` means that text is rendered more accurately. `low` has the advantage of faster rendering.

13.3 PDF printer queue preferences

```
Key:  Printers/<printer queue>/<preference>
```

PDFTransparency bool FALSE

(HELIOS Admin option PDF Tranparency)

If a PDF document that contains transparencies is sent to a printer queue with this preference set to TRUE, PostScript with “pdfmark” constructs is generated. With the default setting (FALSE) an error is generated. For more information on PDF transparencies refer to 16 “PDF Transparency”.

PreserveDeviceN bool FALSE

(HELIOS Admin option PostScript 3/DeviceN Output)

This preference applies only to OPI image replacement. If you want to print a PDF document with output optimized to use the PostScript 3/DeviceN feature, you must print it using the “pdfprint” command line tool with the appropriate option (see 7.1 “pdfprint”):

```
pdfprint -o preservedevicen
```

Another possibility is to print via the PDF HandShake Print plug-in for Acrobat 8, with activated Preserve DeviceN checkbox.

DefaultPrinterProfile str ""

(HELIOS Admin option Default Printer Profile)

This preference specifies the path name of the default ICC profile describing the printing device.

DefaultProofProfile str ""

(HELIOS Admin option Default Proof Profile)

This preference specifies the path name of the default ICC profile describing the proof device. The string will only be recognized if a default printer profile is specified.

CompositeColorSpace `str` `"CMYK"`

This preference sets the color space used while printing composite. The default is CMYK. Setting this option to `"None"` causes all color images in a print job to be kept and printed in their original color space. This applies to CMYK, CIELab and RGB images only. Valid strings are `"CMYK"`, `"RGB"`, and `"CIELab"`. Please note that the value you choose for this preference can be overridden by using the `DefaultPrinterProfile` option.

Note: In a "Create PDF" queue, the default behavior is "None".

PureWhite `bool` `FALSE`

(HELIOS Admin option `Preserve Raster Colors - White`)

Exclude white colored `raster` objects in PDF files from color transformation, so that they remain `white` even if color matching is done. CMYK values will be `zero`, i.e. no ink is applied to the plates.

PureGrays `bool` `FALSE`

(HELIOS Admin option `Preserve Raster Colors - Gray`)

Exclude gray colored `raster` objects in PDF files from color transformation, so that they remain `gray` even if color matching is done. Gray raster objects in Gray/RGB/CIELab/Indexed color spaces are detected and converted to `Gray` only for CMYK output.

PureBlack `bool` `FALSE`

(HELIOS Admin option `Preserve Raster Colors - Black`)

Exclude black colored `raster` objects in PDF files from color transformation, so that they remain `black` even if color matching is done. Black raster objects in Gray/RGB/CIELab/Indexed color spaces are detected and converted to `Black` only for CMYK output.

PureCMY `bool` `FALSE`

(HELIOS Admin option `Preserve Raster Colors – CMY`)

Applies to CMYK to CMYK conversions of pixels of raster images, and preserves the input color *only if* black is 0% and exactly one CMY primary color is 100% and all other primary colors are 0%.

PureVectorWhite `bool` `FALSE`

(HELIOS Admin option `Preserve Vector Colors – White`)

Exclude white colored `vector` objects in PDF files from color transformation, so that they remain `white` even if color matching is done. CMYK values will be `zero`, i.e. no ink is applied to the plates.

PureVectorGrays `bool` `FALSE`

(HELIOS Admin option `Preserve Vector Colors – Gray`)

Exclude gray colored `vector` objects in PDF files from color transformation, so that they remain `gray` even if color matching is done. Gray text and vectors in Gray/RGB/CIELab/Indexed color spaces are detected and converted to `Gray` only for CMYK output.

PureVectorBlack `bool` `FALSE`

(HELIOS Admin option `Preserve Vector Colors – Black`)

Exclude black colored `vector` objects in PDF files from color transformation, so that they remain `black` even if color matching is done. Black text and vectors in Gray/RGB/CIELab/Indexed color spaces are detected and converted to `Black` only for CMYK output.

PureVectorCMY `bool` `FALSE`

(HELIOS Admin option `Preserve Vector Colors – CMY`)

Applies to CMYK to CMYK conversions of text and vector objects, and preserves the input color *only if* black is 0% and exactly one CMY primary color is 100% and all other primary colors are 0%.

createlog `bool` `TRUE`

Creates log files for Create PDF workflows. If set to `FALSE`, no log file is generated after a Create PDF print job.

This preference can also be set globally, with the key:

`Global/Printers/createlog`

13.3.1 "pdfif" preference keys

distillhost `str` `""`

(HELIOS Admin option `Host Name`)

This preference specifies the host name of the machine where the CreatePDF Server/Acrobat Distiller is installed.

distillservice `str` `"2005"`

Specifies the TCP port number of the Create PDF Server. If you change the port number to another value, you must also change the port number of the "Create PDF Processor".

distillprefix `str` `""`

(HELIOS Admin option `Name Prefix`)

This preference will only be recognized if **distilldir** is also specified. With "string", you can define a file name prefix for all PDF files that are sent to the selected directory. This can help you classify your PDF files if you have several "Create PDF" queues and save all PDF files into the same directory.

distilldir `str` `""`

(HELIOS Admin option `Directory`)

This preference specifies the path of the directory where the generated PDF files are saved.

createpdf `bool` `TRUE`

This preference determines if a PDF file is created at all.

distillnotifyprog `str` `""`

(HELIOS Admin option `Notify Program`)

This preference will only be recognized if **distilldir** is specified as well. With this preference the path of the "distillnotify" program is specified.

E.g.: `/usr/local/bin/crpdfnotify`

"pdfif" exports the following environment variables during the program call:

```
HELIOSDIR=
   <environment variable for the HELIOS install path>
HELIOS_JOBFOR=
   <user name (if available) of print job creator>
HELIOS_JOBUSER=
   <user name (if available) as known to "lpd">
HELIOS_JOBHOST=
   <name (if available) of server which hosts the printer queue>
HELIOS_JOBTITLE=
   <print job title (if available)>
HELIOS_PRINTERNAME=
   <logical (UNIX) name of the printer queue>
HELIOS_JOBPAGES=
   <expected number of pages. This value may differ slightly
   from the actual number>
HELIOS_JOBSIZE=
   <expected print job size (in bytes). The stated value is
   often smaller than the actual size>
HELIOS_VFFILE=
   <name of created PDF file>
```

distillresolve `bool` `TRUE`

Specifies if any resolving is done within a PostScript job. When a print job is processed to a "Create PDF" queue, and the switch is set to `FALSE`, the print job remains "untouched" and is passed on without any changes.

userdir `bool` `FALSE`

(HELIOS Admin option `Create Folder for Each User`)

Specifies if the PDF files are stored in a folder per user.

writesize `int` `131072 (128 x 1024)`

Specifies the TCP write buffer size in bytes.

readsize `int` `131072 (128 x 1024)`

Specifies the TCP read buffer size in bytes.

keepepsfsize `bool` `TRUE`

If set to `TRUE`, this preference determines that the PDF page size is taken automatically from the BoundingBox if the job is an EPSF file rather than a PostScript file. This preference is only meaningful if **distillresolve** is set to `TRUE` as well.

SendEvent `bool` `FALSE`

If this preference is set to `TRUE`, the "Print To Disk" interface reports events to all listening services.

13.4 Ghostscript Create PDF preference keys

The following keys require a restart (see "srvutil" in the HELIOS Base manual) of the "createpdf" service to take effect:

```
Key:  Programs/createpdf/<preference>
```

ghostscript `str` `(see description)`

Full path to the Ghostscript executable. If the preference is not defined, the executable ("gs" on UNIX, "gswin32c" on Windows) is searched in all directories of the default search path and in the following locations:

UNIX:

```
/usr/bin; /usr/sbin; /usr/local/bin; /bin; /opt/bin;
/opt/local/bin
```

Windows:

```
C:\gs\bin; C:\Program Files\gs\bin; C:\Programme\gs\bin;
```

Example (UNIX):

```
# prefvalue -k Programs/createpdf/ghostscript -t str
  /sw/bin/gs
```

("gs" is installed in "/sw/bin".)

Example (Windows):

```
# prefvalue -k Programs/createpdf/ghostscript -t str
  "C:\\Program Files\\gs\\gs8.64\\bin\\gswin32c.exe"
```

port `int` `2018`

Specifies the TCP/IP port number where to listen for jobs.

ipaddress `str` `(see description)`

TCP/IP interface address used by the "createpdf" service. If the preference is not set, "createpdf" listens on the address "any", which means that connections on all interfaces are accepted.

Example:

```
# prefvalue -k Programs/createpdf/ipaddress -t str
  "127.0.0.1"
```

Restricts access to the local interface. In this example, only "Create PDF" printer queues running on the same host can access this "createpdf" server.

TempDir `str` `"HELIOSDIR/var/tmp"`

Specifies the path to a directory where temporary files are created.

14 Example workflow

- Do your customers use a different DTP application than you do?
- How do you exchange documents?
- And what about the colors on the printouts?

With PDF HandShake there are no application incompatibilities anymore. Here is an example of how you can get high-quality printouts of a document that was originally created with Microsoft Word and then transformed into PDF. The example production environment is illustrated in Fig. 14.1.

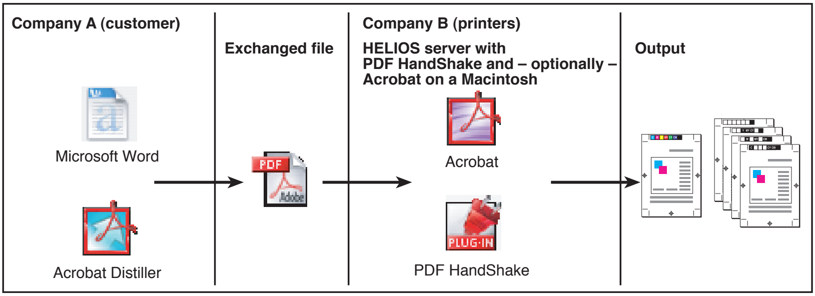


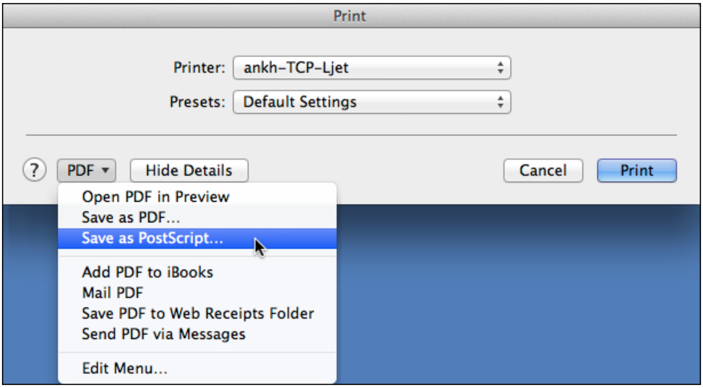
Fig. 14.1: Example workflow

The corresponding workflow is described below:

Company A (customer):

➢ Create a complete document using Microsoft Word. Import graphics, e.g. Illustrator EPSF graphics and images or Photoshop PICT images, if desired.

➢ Set up your printer driver as shown in Fig. 14.2 and print your document into a PostScript file.



Printer: ankh-TCP-Ljet

Presets: Default Settings

Fig. 14.2: Printing the Word document into a PostScript file

➢ Convert the PostScript file into PDF. (Check 15 "Create PDF files using Acrobat Distiller" for advice on the best-suited Distiller options.) Embed all fonts your printers do not have.

➢ Send the PDF file to your printers.

Company B (printers):

➢ Start the "pdfprint" program on your server or install our Acrobat plug-in on a Mac client and open the `PDF HandShake Print...` dialog in your Acrobat application (Fig. 14.3).

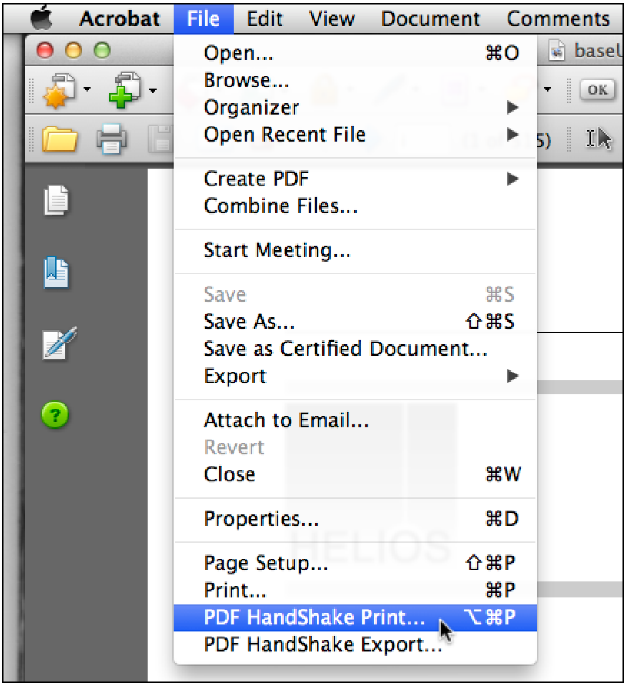


Fig. 14.3: Printing with “PDF HandShake Print”

➢ Set up all required parameters and print the PDF document you have received. Note that for best color matching results, it might be helpful to get ICC profiles from Company A. For example, if the document contains several Illustrator illustrations, the RGB profile of the designer’s monitor would guarantee the best results. However, it is also possible to use the standard profiles we deliver with PDF HandShake.

Note: Of course, the application on the input side is exchangeable. Microsoft Word is *only one* option.

In the case that both of your companies are using either EtherShare, PCShare or WebShare and have PDF HandShake installed, you can both make use of the more than one hundred fonts we deliver.

Your printers will then have the same fonts available on the server. So you do not have to include them in your PostScript job, and you do not have to embed them when distilling the file. The exchanged files will become smaller.

15 Create PDF files using Acrobat Distiller

PDF HandShake offers the opportunity of using PDF as an exchange format in the printing business. The results on the output side, however, depend on the quality on the input side. So for PDF generation you should follow the instructions given below or communicate the instructions to every customer who submits PDF files to you. We recommend to use Acrobat Distiller for PDF generation, and to consult the Acrobat online help.

In the following, we briefly discuss the Distiller job options that might be relevant for the PDF HandShake workflows.

Note: Please note that some of the Distiller settings described below are best suited for printing only – they may be disadvantageous for other purposes such as web production or re-editing of files using Adobe Acrobat.

➢ Start Acrobat Distiller, open `Settings > Edit Adobe PDF Settings...` and select the `General` tab to open the dialog shown in Fig. 15.1.

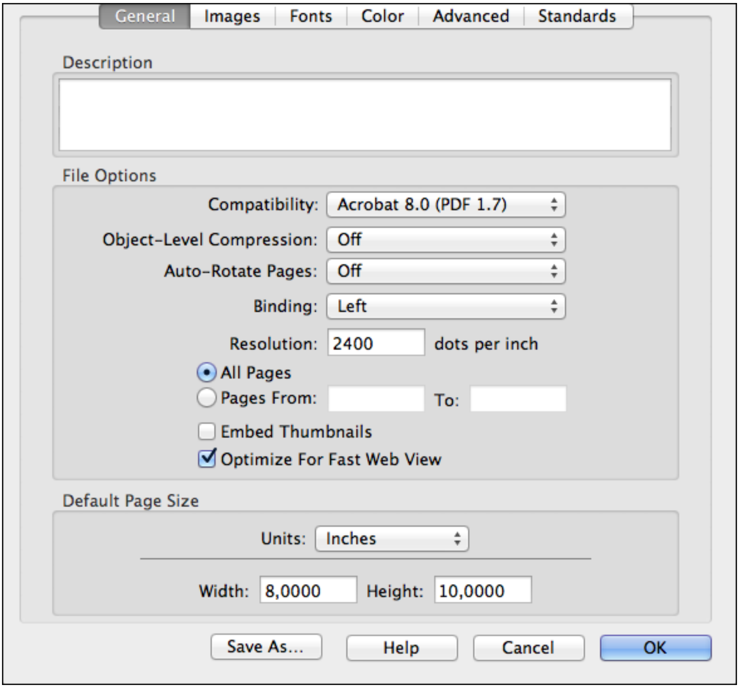
File Options

`Compatibility` should be set to `Acrobat 4` (PDF 1.3) in order to provide a high grade of compatibility with different Acrobat versions. For “Create PDF” printer queues with the `PDF Transparency` option enabled, compatibility should be set to Acrobat 5 (PDF 1.4) or higher.

Default Page Size

The default document page size can be arbitrarily specified in this section.

➢ In the `Units` pop-up menu select `Centimeters` or `Inches`.



Description

File Options

Compatibility: Acrobat 8.0 (PDF 1.7)

Object-Level Compression: Off

Auto-Rotate Pages: Off

Binding: Left

Resolution: 2400 dots per inch

● All Pages

○ Pages From: To:

□ Embed Thumbnails

☑ Optimize For Fast Web View

Default Page Size

Units: Inches

Width: 8,0000 Height: 10,0000

Fig. 15.1: Setting up `General` settings

➢ Then select the `Images` tab as shown in Fig. 15.2.

Color Images

`Sampling` should be switched off in the pop-up menu, and `Compression` should be `ZIP`. Even though PDF HandShake is able to uncompress every kind of PDF file, any type of `JPEG`, `JPEG 2000` or `Automatic` compression may entail data loss and thus reduce quality on the output side.

Grayscale Images

Here, the same settings as described in `Color Images` above should be applied.

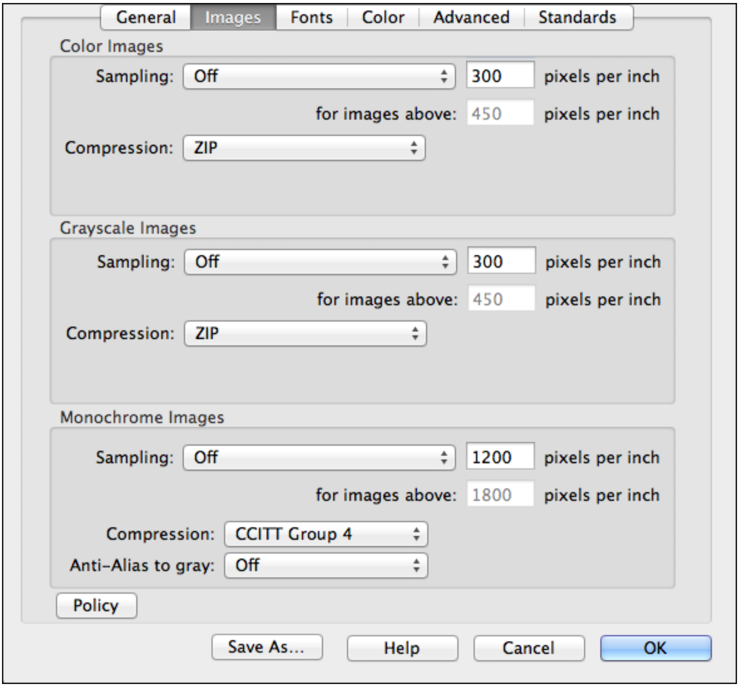
Monochrome Images

If you intend to decrease the file size for monochrome bitmap images switch `Sampling` off and select `CCIT Group 4` or `Run Length` from the corresponding pop-up menu.

➢ Switch to the `Fonts` tab to specify the fonts you want to include in your PDF file. Fig. 15.3 shows the settings that are best suited for printing.

`Embed All Fonts` guarantees that all fonts used in the original document will be contained in the resulting PDF file. This is especially important if you plan to use the PDF file with PDF HandShake on a different system later. Note that the Distiller software can only embed fonts that are either embedded in the PostScript file or available in one of your system's font locations. TrueType fonts will only be embedded if they are already included in the PostScript file.

If you are fairly sure that the fonts you used for document creation are available on the system that will be used for production, you may uncheck `Embed All Fonts`.



Color Images

Sampling: Off 300 pixels per inch

for images above: 450 pixels per inch

Compression: ZIP

Grayscale Images

Sampling: Off 300 pixels per inch

for images above: 450 pixels per inch

Compression: ZIP

Monochrome Images

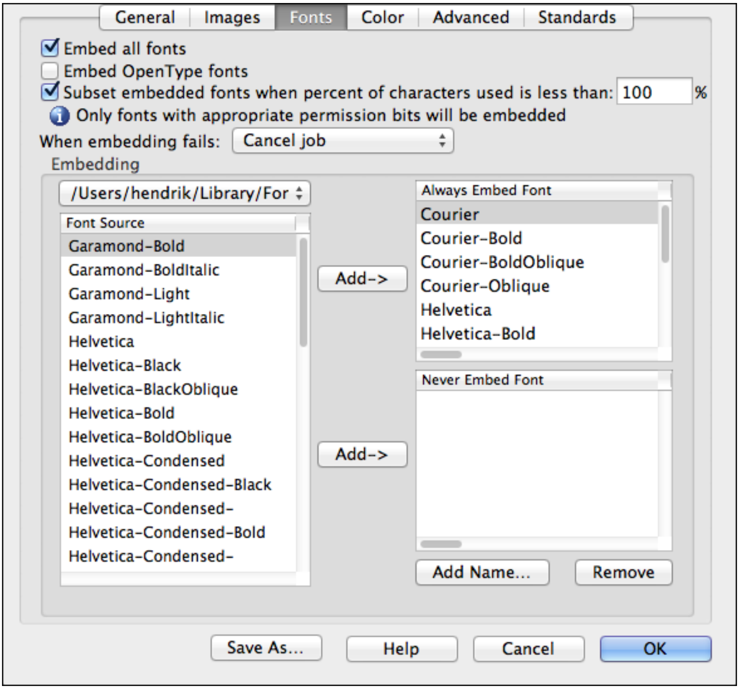
Sampling: Off 1200 pixels per inch

for images above: 1800 pixels per inch

Compression: CCITT Group 4

Anti-Alias to gray: Off

Fig. 15.2: Setting up `Images` settings



☑ Embed all fonts

□ Embed OpenType fonts

☑ Subset embedded fonts when percent of characters used is less than: 100 %

Only fonts with appropriate permission bits will be embedded

When embedding fails: Cancel job

Embedding

/Users/hendrik/Library/For

Font Source
Garamond-Bold
Garamond-BoldItalic
Garamond-Light
Garamond-LightItalic
Helvetica
Helvetica-Black
Helvetica-BlackOblique
Helvetica-Bold
Helvetica-BoldOblique
Helvetica-Condensed
Helvetica-Condensed-Black
Helvetica-Condensed-
Helvetica-Condensed-Bold
Helvetica-Condensed-

Always Embed Font
Courier
Courier-Bold
Courier-BoldOblique
Courier-Oblique
Helvetica
Helvetica-Bold

Never Embed Font

Fig. 15.3: Setting up `Fonts` settings

The `Subset Embedded Fonts When Percent Of Characters Used Is Less Than` option should be used, and set to 100%, to make sure that the fonts embedded in the PDF file are really used for printing.

When printing with the Acrobat plug-in or “pdfprint”, all fonts that are embedded in a PDF file will be sent to the printer. However, depending on the printer’s default settings, a font from the document might be rejected in case the printer already has a font of the same name. Problems can arise if the font in the PDF file, e.g. *Chicago*, is not identical to the printer’s *Chicago* font.

If you use `Subset Embedded Fonts When Percent Of Characters Used Is Less Than` and set it to the highest value – `100%` – you can make sure that *every* embedded font in the document will be assigned a new and unique font name (e.g. "CMFONL+Chicago" for a Chicago subset font) that does not coincide with the standard font names on a printer.

Note: If you work with `Subset Embedded Fonts When Percent Of Characters Used Is Less Than` and set it to a rather high value you should keep in mind that the Adobe Acrobat cannot re-edit text – in a PDF file – that uses subset fonts.

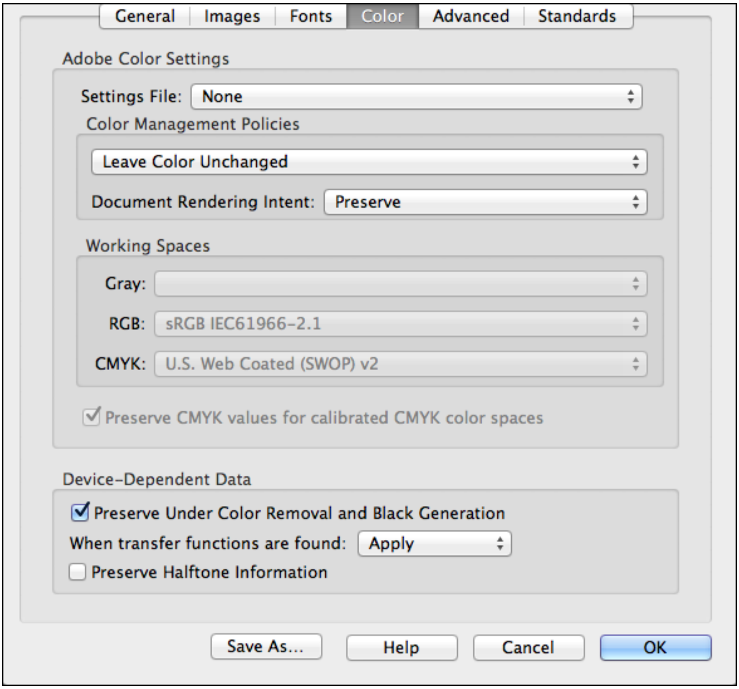
The `Color` tab can be used to apply color settings. We recommend to use the settings shown in Fig. 15.4.

➢ Choose `Leave Color Unchanged` from the pop-up menu, as shown in the example, unless you intend to edit the colors in the PDF document.

The profiles pop-up menus for `Gray`, `RGB` and `CMYK` remain grayed out as long as you maintain `Leave Color Unchanged`.

`Preserve Under Color Removal` and `Preserve Halftone Information` can be checked if required. If your PostScript file contains e.g. halftone information you want to use for later printing, check `Preserve Halftone Information`. The "pdfprint" program and the Acrobat plug-in both have an option that allows to use for printing the halftone information that is included in the PDF file.

➢ Switch to the `Advanced` tab as shown in Fig. 15.5.



Adobe Color Settings

Settings File: None

Color Management Policies

Leave Color Unchanged

Document Rendering Intent: Preserve

Working Spaces

Gray:

RGB: sRGB IEC61966-2.1

CMYK: U.S. Web Coated (SWOP) v2

☑ Preserve CMYK values for calibrated CMYK color spaces

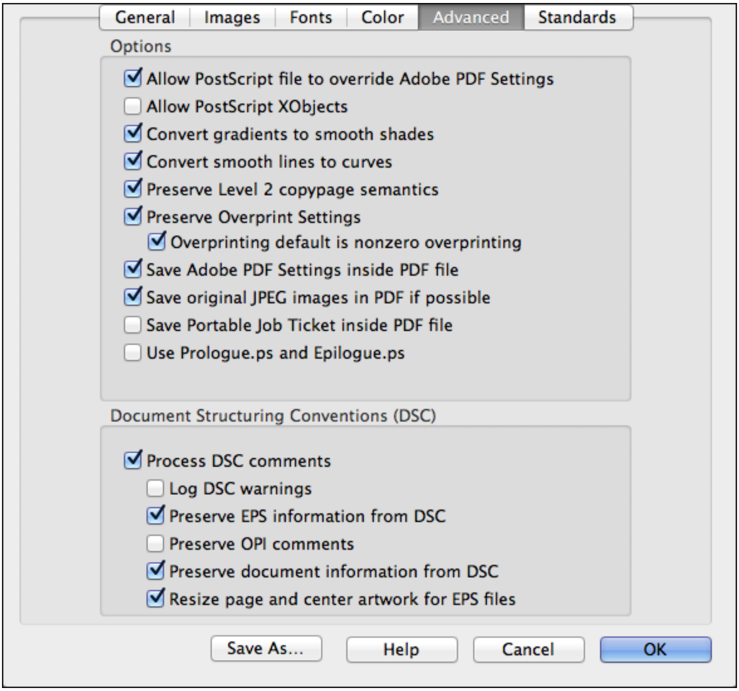
Device-Dependent Data

☑ Preserve Under Color Removal and Black Generation

When transfer functions are found: Apply

□ Preserve Halftone Information

Fig. 15.4: Setting up `Color` settings



Options

☑ Allow PostScript file to override Adobe PDF Settings
□ Allow PostScript XObjects
☑ Convert gradients to smooth shades
☑ Convert smooth lines to curves
☑ Preserve Level 2 copypage semantics
☑ Preserve Overprint Settings
  ☑ Overprinting default is nonzero overprinting
☑ Save Adobe PDF Settings inside PDF file
☑ Save original JPEG images in PDF if possible
□ Save Portable Job Ticket inside PDF file
□ Use Prologue.ps and Epilogue.ps

Document Structuring Conventions (DSC)

☑ Process DSC comments
  □ Log DSC warnings
  ☑ Preserve EPS information from DSC
  □ Preserve OPI comments
  ☑ Preserve document information from DSC
  ☑ Resize page and center artwork for EPS files

Fig. 15.5: Setting up `Advanced` settings

`Preserve OPI Comments` is only relevant if you are using ImageServer. Both choices are possible: If you are fairly sure that your input document is complete, meaning that all required fonts and high-resolution images are embedded, you should uncheck the option. In that case OPI comments are no longer relevant. If you check `Preserve OPI Comments` this would mean that OPI comments that are already included in your EPSF or PostScript file will be included in the PDF file as well; and they will be resolved by the OPI server later. For example, if the PDF document contains a TIFF layout file and a reference to the high-resolution original, the OPI server will be able to perform image replacement according to the reference in the PDF file. Problems can arise if you send PDF files with OPI comments to another company and do not also send the high-resolution images. In that case, the OPI comments cannot be resolved and the PDF files cannot be printed correctly.

Note: Creating PDF files from separated PostScript files using Acrobat Distiller works just as well as with composite files.

➢ Finally, switch to the `Standards` tab as shown in Fig. 15.6.

The ISO (*International Organization for Standardization*) has issued several PDF/X standards by now. Each PDF/X standard describes a specific subset of an Adobe PDF standard version. The goal of all PDF/X standards is to achieve high interoperability and reproducibility without appearance variations. Unrestricted PDF documents are known for color, font, and trapping problems, which PDF/X eliminates.

PDF/X, a subset of the PDF specification created by Adobe, eliminates many of the color, font, and trapping variables that lead to printing problems.

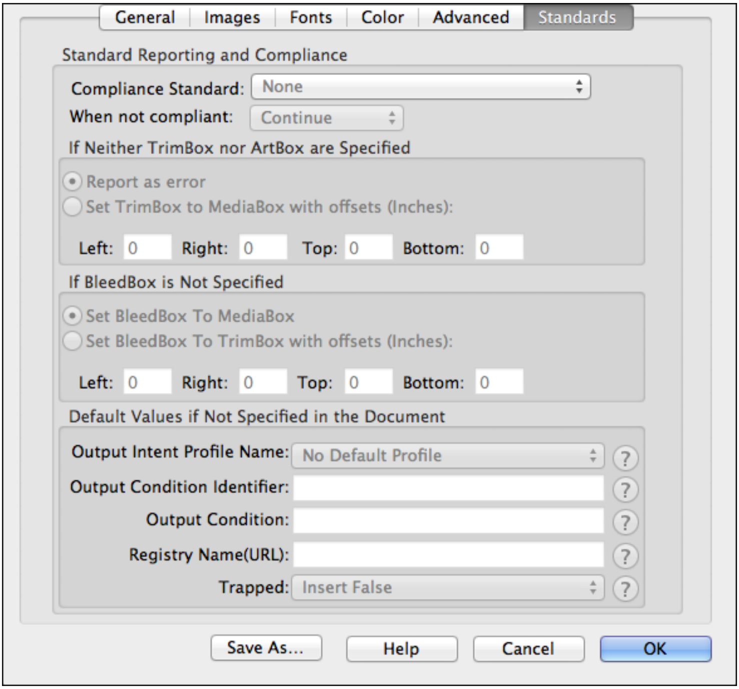
There are two commonly used versions of PDF/X, both approved ISO standards:

PDF/X-1a

PDF/X-1a is a specification for the blind exchange of PDF files which contain CMYK and spot color information.

PDF/X-3

The second, PDF/X-3 is very similar to PDF/ X-1a, except it allows using device independent color (Lab, color profiles and calibrated color spaces).



Standard Reporting and Compliance

Compliance Standard: None

When not compliant: Continue

If Neither TrimBox nor ArtBox are Specified

● Report as error

○ Set TrimBox to MediaBox with offsets (Inches):

Left: 0 Right: 0 Top: 0 Bottom: 0

If BleedBox is Not Specified

● Set BleedBox To MediaBox

○ Set BleedBox To TrimBox with offsets (Inches):

Left: 0 Right: 0 Top: 0 Bottom: 0

Default Values if Not Specified in the Document

Output Intent Profile Name: No Default Profile

Output Condition Identifier:

Output Condition:

Registry Name(URL):

Trapped: Insert False

Fig. 15.6: Setting up `Standards` settings

`Standard Reporting and Compliance` defines the PDF/X specification used, what Distiller is to do when it finds files that are not compliant, etc.

The `Compliance Standard` pop-up menu is used to specify which version of PDF/X is desired. PDF/X-1a is used for device CMYK and spot color workflows; PDF/X-3 is used for color managed workflows.

When not compliant – Distiller can either fail a non-compliant job or it can continue to create a PDF file that is as close to PDF/X compliance as possible. Using `Continue` is not recommended, because it does not create PDF/X compliant files.

`If neither TrimBox nor ArtBox are Specified` – The specification of at least one of these boxes is a requirement of PDF/X (even if the offset values are set to *zero*.) It is always best to check with the printer to get this setting correct, otherwise, leave the offset values set to zero. This will cause the trim box to be set to the page size.

`If BleedBox is not Specified` – The Bleed Box specification is another requirement of PDF/X. If it is not specified or Distiller cannot detect the bleed box from the incoming file (Distiller contains smarts to detect this information, even if the user does not specify it), you can set the bleed box value in relation to the trim box value. Because the size of the bleed box is always greater than the size of the trim box, the offset indicated here is the distance outside the trim box.

`Default values if not specified in the document` – The document output intent is specified in this document. This describes the intended output condition for the file. The PDF/X specification allows you to identify these output conditions in two ways. You can choose either an ICC profile that describes the device or condition for which the file is prepared or a characterized printing condition can be used. Distiller has a preference for specifying an ICC profile. Based on the profile chosen, Distiller will select the proper characterized printing condition for you.

If you do specify a characterized printing condition, it must be registered in order for it to comply with the PDF/X standard.

In addition, the `Trapped` parameter is specified in this section. This parameter is meant to indicate whether or not the file contains trapping information. In almost all cases, the value of this flag will be `Insert False`. Unless you are manually trapping a file, using a trapping plug-in or do not want the file to be trapped, leave this as `Insert False`.

16 PDF Transparency

The PDF standard supports transparency since PDF Reference 1.4 and Acrobat 5 and is continuously enhanced by Adobe. Recent versions of page layout applications such as Adobe InDesign, Adobe Illustrator, QuarkXPress[1] or Esko ArtPro generate PDF documents with transparencies for overlapping transparent objects and drop shadows.

When PDF documents with transparencies are printed on PostScript printers, the printing application must flatten the transparencies, since PostScript does not support transparency. Acrobat Distiller[2] can process PostScript input and supports transparency using the “pdfmark” constructs extension of the PostScript language defined in “pdfmark Reference, Adobe Acrobat SDK Version 8.0” from 2006.

PDF HandShake can convert PDF documents with transparencies to PostScript files with “pdfmark” constructs, and Acrobat Distiller can convert these files back to PDF documents with transparencies. Furthermore, both ImageServer and PrintPreview can convert PostScript with “pdfmark” constructs to raster images with correct visualization of transparencies.

If you want to use PostScript printers or PostScript processing software other than Acrobat Distiller 8, PrintPreview and ImageServer, you probably cannot use PostScript with “pdfmark” constructs as input.

[1] QuarkXPress 7 also creates its native documents using transparency, but exports them as PDF documents using flattening, i.e. transparent objects in QuarkXPress native documents are converted into non-transparent objects in PDF. This was improved in QuarkXPress 8.

[2] version 8 or later

16.1 Control transparency processing in PDF HandShake

16.1.1 PDF transparency in printing

The `PDF Transparency` checkbox in the HELIOS Admin printer settings determines whether printing PDF documents with transparencies (via the HELIOS Acrobat plug-in, the “pdfprint” command, or from an OPI process with layouts replaced by PDF high-resolution images) results in an error or whether PostScript with “pdfmark” constructs is generated. The `PDF Transparency` option of printer queues is disabled by default. Please note that there is a very limited number of printers and PostScript processing applications which can process PostScript with “pdfmark” constructs correctly. Nearly all printers and PostScript processing software will ignore the “pdfmark” transparency constructs and silently replace transparent objects with non-transparent objects, resulting in substantially wrong visual output.

16.1.2 PDF to EPSF conversion and EPSF layout generation from PDF documents

PDF documents can be converted to both vector-based and raster-based EPSF files using the HELIOS ImageServer “layout” program in convert mode, i.e. using the `-l` option. Conversion of PDF documents with transparencies to raster-based EPSF files or other raster image formats is supported by PDF HandShake with ImageServer. Conversion of PDF documents with transparencies to vector-based EPSF files will produce EPSF files using “pdfmark” constructs by default. Since the number of printers or PostScript processing applications that can properly process these EPSF files is limited, you can prevent the generation of “pdfmark” constructs in vector-based EPSF output by option `-o PDFTransparency=FALSE`. If this option is used, “layout” aborts conversion with an error if a source PDF document contains transparencies.

The same rules apply to EPSF layout generation from PDF documents with transparencies.

16.1.3 High-resolution PDF documents with transparencies in OPI workflows

PDF documents with transparencies can be used as high-resolution image files in OPI workflows if the intended output device or processing software supports “pdfmark” constructs, e.g. with a “Create PDF” queue. But you should configure the OPI server to generate raster layouts for PDF high-resolution image files since page layout applications such as InDesign and QuarkXPress cannot display vector-based EPSF layouts with “pdfmark” constructs correctly. If you want to exclude PDF documents with transparencies from OPI workflows, you can set the preference `Global/Opi/PDFTransparency` to `FALSE`.

16.1.4 Previews of PDF documents with transparencies in PrintPreview

Printing PDF documents with transparencies via “pdfprint” or the Acrobat PDF HandShake plug-in to a PrintPreview queue is fully supported.

16.2 Configure Distiller 8 for processing PDF HandShake output with transparencies

Acrobat Distiller versions before version 8 are not suitable for PDF HandShake output with transparencies. By default, in Distiller 8 (and later), support for “pdfmark” constructs for transparencies is not enabled. The transparency feature of Distiller 8 cannot be changed in its `Edit Adobe PDF Settings...` dialog. The transparency feature of Distiller is determined by the Adobe PDF

Settings file selected in the Distiller user interface. The Adobe PDF Settings files for Mac are usually located in "/Library/Application Support/Adobe/ Adobe PDF/Settings/". The shipped Adobe PDF Settings files all disable Distiller's transparency feature. To enable it, you need to edit these text files and change the default setting of `/AllowTransparency` from `FALSE` to `TRUE`. When Distiller processes a PostScript file containing "pdfmark" transparency features and `/AllowTransparency` is `FALSE`, Distiller generates an error.

16.3 Limitations of PDF Transparency support in ImageServer and PrintPreview

1. Printing PDF documents with transparencies to printer queues with output color space CIELab is not allowed and will result in an error.

2. Separated printing of PDF documents with transparencies may result in wrong visual output when non-separable blend modes are used. Non-separable blend modes are part of the PDF transparency imaging model.

3. PDF HandShake has a sophisticated built-in color management. PDF HandShake converts the color of objects in PDF documents to the color space of the selected output device before they are used for transparency computations. This combination of color management and transparency computation may result in unwanted variations with regards to visual output color.

4. Combinations of PDF transparencies with spot colors or overprints are critical. You cannot expect PDF documents with such combinations to have consistent visual appearance using different PDF viewing or processing applications, or using different options in just one PDF viewing or processing application.

16.4 Troubleshooting with PDF transparencies

Different workflows, different PDF and raster image viewing applications or different settings used in one viewing application tend to produce substantially different results for these critical features. OS X Preview, Adobe Acrobat and Photoshop often produce substantially different output for these critical features. Here is one important advice if you experience missing transparent objects for a particular PDF document with PDF HandShake.

The `Overprint Preview` flag in Adobe Acrobat can have significant influence on how the document is displayed. If you have a critical PDF document with transparencies, please view it in Adobe Acrobat and toggle the `Overprint Preview` flag. If some elements appear with one setting and disappear with the other, you will probably need to set the `DeviceN/PostScript 3` (HELIOS Admin) or `PreserveDeviceN` (Acrobat PDF HandShake plug-in) options, and if that does not help, you will probably have to re-create the PDF document in its page layout application with different settings.

17 PDF layers

PDF layers (or "optional content") are sections of content that can be selectively viewed or hidden. Layers are used for items such as CAD drawings, layered artwork, maps, and multi-language documents. The layers of a document are shown in the `Layers` navigation panel of Adobe Acrobat (`View > Navigation Panels > Layers`). Each layer has a name and a default state (*ON* or *OFF*).

17.1 pdfinfo

"pdfinfo" shows all layers and the default states.

Example:

```
$ pdfinfo -o layer itto.pdf
# pdfinfo 4.0.0
Layer: Name=English, ObjNo=59, DefaultState=Off
Layer: Name=Spanish, ObjNo=63, DefaultState=Off
Layer: Name=French, ObjNo=55, DefaultState=Off
Layer: Name=German, ObjNo=57, DefaultState=On
Layer: Name=Italian, ObjNo=61, DefaultState=Off
Layer: Name=Diagram, ObjNo=53, DefaultState=On
```

Some resources (such as images) are not shown by "pdfinfo" if they belong to a layer that is hidden by default. The option `-l <layerSelection>` can be used to modify the layer states, see 17.3 "Layer selection" below.

17.2 PDF HandShake Acrobat plug-in

The PDF HandShake Acrobat plug-in always uses the currently selected layer states and therefore prints the same layers that Acrobat would print.

17.3 Layer selection

A layer selection can be used with

```
   pdfinfo -l <LayerSelection>
   pdfprint -o layer=<LayerSelection>
```

to override the default states of the layers in a PDF document. `<LayerSelection>` is a string consisting of one ore more directives separated by a colon (":"). Each directive is one of the following:

`None:`	Set all layers to *OFF*.
`All:`	Set all layers to *ON*.
`name or +name:`	Set the layer named "name" to *ON*.
`-name:`	Set the layer named "name" to *OFF*.
`number or +number:`	Set the layer with object number "number" to *ON*.
`-number:`	Set the layer with object number "number" to *OFF*.

Layer names are *not unique*. If multiple layers have the same name, they are all modified by the `name` directive.

17.3.1 pdfprint

By default, optional content is printed or omitted according to the default layer states. The states can be overridden with the option `-o layer=<layerSelection>`.

Example:

```
$ pdfprint -P ppv -p pdf itto.pdf
$ pdfprint -P ppv -p pdf -olayer=French:-German itto.pdf
$ pdfprint -P ppv -p pdf -olayer=None:French:Diagram itto.pdf
```

The first command prints the German version of the document (layer “German” and “Diagram” are *ON* by default, see 17.1 “pdfinfo” example above). The second command prints the French version of the document by activating the layer “French” and deactivating the layer “German”. The third command also prints the French version, now by deactivating all layers and then activating “French” and “Diagram”.

See 17.3 “Layer selection” for the syntax of `<LayerSelection>`.

17.3.2 layout

By default, “layout” shows or hides optional content according to the default layer states. The states can be overridden with the “PDFLayer” attribute.

Example:

```
$ layout -l -oPrintColor=RGB -T PNGf itto.pdf itto-german.png
$ layout -l -aPDFLayer=None:French:Diagram -oPrintColor=RGB -T PNGf
            itto.pdf itto-french.png
```

The first command creates an image of the German version of the document (layer “German” and “Diagram” are *ON* by default, see the 17.1 “pdfinfo” example above). The second command creates an image of the French version of the document.

A About fonts

With PDF HandShake you can use complete PDF documents in the printing business or in an OPI workflow. PDF documents – in contrast to images such as TIFF or Photoshop files – may already contain text elements using included or referenced fonts. So, with PDF HandShake, we have to find a way to read the font information from a given PDF file and to use this information for printing (and for layout generation if you use HELIOS ImageServer).

Quick reference

Section A.1 "PDF font processing" gives an overview of how the fonts in a PDF document are handled by PDF HandShake during printing and during layout generation by ImageServer. The paragraph is a quick reference.

Full-scale reference

Section A.2 "Font types in PDF" gives some more background information about fonts, explains the problems that could arise, and illustrates the workflows when handling fonts. This chapter is meant for those who "want to learn more".

Appendix B "The fonts we deliver" lists all the fonts we deliver with our product.

A.1 PDF font processing

PDF HandShake can only use the fonts that are embedded in a given PDF document or those that are available on the server or printer. For OPI layout generation, the fonts must be either embedded or accessible on the server.

Embedded/non-embedded fonts

Whenever a font is embedded in a PDF file, it will be used by PDF HandShake. This is why we recommend to always embed all fonts when creating PDF files – at least if you are working on different systems.

The fonts we deliver together with our software package are meant to ensure high-quality printing for all PDF files, even for those that do only contain font references and thus are dependent on the available system fonts.

PostScript/TrueType fonts

There are different categories of fonts, namely PostScript fonts and TrueType fonts. Even though PDF HandShake requires PostScript printers (which usually require PostScript fonts), the program can also handle TrueType fonts.

PostScript fonts are also divided into different categories. The most frequently used PostScript fonts are “Type 1” and “Type 3” fonts.

The handling of fonts during printing

PDF files with PostScript Type 1 or Type 3 fonts do not cause any problems. Printing can only fail if a font is not embedded *and* is not available on the server or printer either. In that case, the print job will be aborted or the printer will replace the missing font with *Courier*.

PDF files with TrueType fonts are handled as follows:

If the TrueType font is embedded, PDF HandShake transforms the TrueType font into a PostScript Type 1 font and prints the PDF file correctly. The output quality can be enhanced if the printer has a built-in TrueType rasterizer (see **TrueType fonts** below).

If the TrueType font is not embedded, PDF HandShake tries to find a corresponding PostScript font – one with the same name – on the system, and uses the PostScript font for printing.

For ImageServer users only

With ImageServer, fonts are not only required for printing, but also for layout generation.

The generation of layouts should never fail for PDF files that contain PostScript Type 1, Type 3 or TrueType fonts. Non-embedded missing fonts will be replaced with *Courier* for the screen preview part of the EPSF layout. The printable part of the layout only becomes relevant if you want to print the layouts instead of the high-resolution PDF files. In that case, fonts are handled the same way as if printing high-resolution files. This means that they might be replaced with `Courier` (see **The handling of fonts during printing** above).

To avoid font replacement during printing, you should always activate the `Check Fonts` option on your printer queue (see the respective chapter in the HELIOS ImageServer manual).

A.2 Font types in PDF

Fonts in PDF files

The PDF file format allows the following font types:

❍ PostScript

- Type 1
- Type 3
- CID-keyed fonts with Type 1 glyphs (*CIDFontType0*)

  - Multiple Master font (Type 1 font extension)

❍ TrueType

  - Non-CID-keyed TrueType fonts (standard)
  - CID-keyed TrueType fonts (*CIDFontType2*)

❍ OpenType

  - OpenType based on TrueType
  - OpenType based on CFF

Font subsets can be generated from all of the font types listed above. Font subsets contain less glyphs than the original font it is derived from and have a different name, e.g.:

- Original font name: *Times-Roman*
- Subset font name: *LCIFOK+Times-Roman*

Different font types and font handling

Before discussing the different types of fonts – at least those that are allowed in PDF documents – we want to define some specific expressions that will be used later:

A.2.1 Bitmap fonts vs. outline fonts

Bitmap fonts

In bitmap fonts, the characters are represented by a pattern of pixels. Angled or curved character elements have a serrated shape that becomes more and more obvious when scaling a bitmap font. Therefore, to avoid heavy serrations, the character patterns have to be different for every point size. This would consume a lot of memory. So, bitmap fonts are not very well suited for digital printing.

Outline fonts

In outline fonts, the shape of each character is described geometrically by lines and curves. Outline-format characters are infinitely scalable, and are therefore not limited to a particular point size. Nearly all font types we discuss below, are outline fonts. Type 3 fonts are the only exception; they can be either in bitmap or in outline format.

PostScript fonts

PostScript fonts are very successful in the printing business because each character of a font is handled like a graphical object. PostScript interpreters apply complex algorithms to the fonts and thus are able to transform them into a pixel pattern for a specific output device.

Type 1 fonts

Type 1 is a font format for single-byte Latin fonts. It can be used with PostScript printers. Type 1 fonts use a specialized subset of the PostScript language that is optimized for performance and compact representation. The Type 1 operator set includes so-called "hint information" to generate accurate bitmaps for small sizes and low resolutions. In general, Type 1 fonts guarantee the most accurate results on printouts. All PostScript fonts we deliver with our software package are Type 1 fonts.

Type 3 fonts

Type 3 fonts can use the full PostScript language to draw their glyphs. Thus, Type 3 fonts can use features such as shadings, multiple colors, and fill patterns, which are not supported in Type 1. One drawback is that Type 3 fonts are not optimized for size or performance like Type 1 fonts are, and there is no built-in method for adding "hint information". Type 3 fonts look slightly bolder than they would if expressed as a Type 1 font. Type 3 fonts can be useful for special-purpose or very complex fonts (such as complex logos). The format also provides a way to represent bitmap characters.

Type 3 fonts are extremely rare in modern prepress environments.

TrueType fonts

TrueType fonts are rather wide-spread. Nevertheless, they can cause problems in high-quality printing. When converting PDF files to PostScript for a printer, TrueType fonts within the PDF file can both be transformed to PostScript Type 1 fonts and be left unchanged for printers with a built-in TrueType rasterizer. A TrueType rasterizer is an optional feature of a PostScript RIP, which enables direct processing of TrueType fonts. A PostScript printer without TrueType rasterizer cannot process TrueType fonts.
The transformation of TrueType fonts to PostScript Type 1 fonts is only an approximation, which results in a slight loss in quality. PDF HandShake generates both a Type 1 font approximation and an unchanged TrueType font in PostScript for each TrueType font in a PDF document. The unchanged TrueType font is embedded in a PostScript Type 42 font envelope. A printer with TrueType rasterizer can be recognized by the following PPD entry:

```
*TTRasterizer:Type42
```

Multiple master fonts

Multiple master font formats are considered extensions to the Type 1 format. Usually, one font file only contains one representation of a specific font as far as weight and width are concerned. For example: *Font1-Light*, *Font1-Regular*, *Font1-Bold*, *Font1-CondensedLight*, *Font1-ExpandedLight*. Multiple master fonts include two or more “master” fonts within a single font file. This allows users to interpolate many intermediate “instances” of the typeface. PDF HandShake does recognize multiple master fonts but cannot use them or transform them into a usable format. The only thing PDF HandShake can do is to try and find a PostScript Type 1 font of the same name on the server or to replace the font with *Courier*.

CID fonts

CID-keying is a method of defining multiple-byte encoded fonts. CID-keying is available as an extension for Type 1 fonts (*CIDFontType0*) and for TrueType fonts (*CIDFontType2*). Glyphs are accessed by their character ID instead of by their glyph name. CID fonts are mainly used by some applications, e.g. Adobe InDesign, and for fonts with large character sets such as Chinese, Japanese, and Korean (CJK) language fonts. PDF HandShake supports CID fonts that are embedded in the document. They are converted to PostScript Type 0 fonts for output.

OpenType fonts

OpenType is an extension of the TrueType format that allows inclusion of CFF fonts (*Compact Font Format*, a compressed representation of a Type 1 font). OpenType fonts based on CFF are treated the same way as Type 1 or CID-keyed Type 1 fonts. OpenType fonts based on TrueType are treated the same way as TrueType or CID-keyed TrueType fonts.

Font naming conventions

PDF HandShake recognizes fonts in PDF documents by name. The naming conventions for fonts are not standardized for all font types and systems, meaning that even though a font is identical on two systems, the font names can be different. A PDF file, for example, can contain a reference to a font called “Times,Italic”, whereas the corresponding font on the server is called “Times-Italic”. PDF HandShake provides a mechanism that allows handling the different naming conventions. There are two searching strategies:

❍ The original font name from the document is used

❍ Otherwise, if the font name used in the document contains a comma, the comma is replaced by a hyphen and the new font name is used. The comma–hyphen replacement is sensible for most of the documents coming from a Windows PC because in Windows (TrueType) font names the font attributes are often separated by a comma. In PostScript font names, there is no comma; font attributes are usually separated by a hyphen

Summary

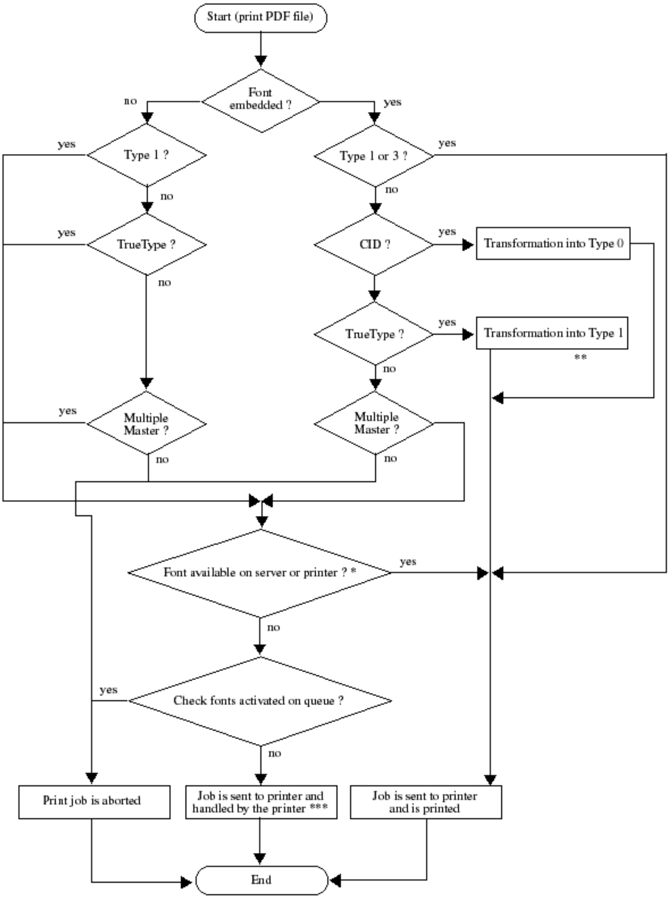
The tables below summarize how PDF HandShake deals with the different types of fonts. The first table describes the handling of fonts during printing, the second one describes the handling of fonts during OPI layout generation. The tables are followed by a flowchart (Fig. A.1) which illustrates what exactly happens on the server and printer when a PDF file is being printed. Please note that the `Check Fonts` option – which is mentioned in the tables – is available to ImageServer users only.

About printer defaults

Printers can handle missing fonts very differently. In most cases, printers use their default font if they receive a print job with missing fonts. For most printers the default font is *Courier*. Most printers deliver a warning message when they substitute a missing font. Some devices abort incomplete jobs completely or halt them and deliver a message so that you can install the missing font.

Font type (Status)	**Printing / Layout generation – Print preview**
Type 1 (embedded)	Embedded font is used.
Type 1 (by reference)	Corresponding Type 1 font from the server or printer is used – if available. Otherwise, if the font is missing, there are two options: If `Check Fonts` is active the job is aborted completely. If `Check Fonts` is not active the handling of the job depends on the printer's default settings; the printer might use *Courier*, abort the job or hold the job and deliver a warning.
Type 3 (embedded)	Embedded font is used.
Type 3 (by reference)	Not relevant (Type 3 fonts are always embedded).
TrueType (embedded)	Font is transformed into Type 1. Both the Type 1 *and* the Type 42 font are sent to the printer. The Type 42 font is used if the printer has a built-in TrueType rasterizer.
TrueType (by reference)	PostScript font of the same name is used – if available on the server. Otherwise, … (see "**Type 1** (included by reference)").
Multiple Master (embedded)	PostScript font of the same name is used – if available on the server (very unlikely). Otherwise, … (see "**Type 1** (included by reference)").
Multiple Master (by reference)	PostScript font of the same name is used – if available on the server (very unlikely). Otherwise, … (see "**Type 1** (included by reference)").
CID (embedded)	Font is transformed into Type 0.
CID (by reference)	Not supported, the job is aborted.

Font type (Status)	**Layout generation – Screen preview**
Type 1 (embedded)	Embedded font is used.
Type 1 (by reference)	Corresponding Type 1 font from the server is included – if available. Otherwise, *Courier* is used.
Type 3 (embedded)	Embedded font is used.
Type 3 (by reference)	Not relevant (Type 3 fonts are always embedded).
TrueType (embedded)	Font is transformed into Type 42. Type 42 font is used.
TrueType (by reference)	PostScript font of the same name is used – if available on the server. Otherwise, *Courier* is used.
Multiple Master (embedded)	PostScript font of the same name is used – if available on the server (very unlikely). Otherwise, *Courier* is used.
Multiple Master (by reference)	PostScript font of the same name is used – if available on the server (very unlikely). Otherwise, *Courier* is used.
CID (embedded)	Font is transformed into Type 0.
CID (by reference)	Not supported, the job is aborted.


Start (print PDF file)
Font embedded ?
no
yes
Type 1 ?
yes
no
Type 1 or 3 ?
yes
no
TrueType ?
yes
no
CID ?
yes
Transformation into Type 0
TrueType ?
yes
no
Transformation into Type 1
**
Multiple Master ?
yes
no
Multiple Master ?
no
Font available on server or printer ? *
yes
no
Check fonts activated on queue ?
yes
no
Print job is aborted
Job is sent to printer and handled by the printer ***
Job is sent to printer and is printed
End

* Common replacement is considered
** Type 42 font is used if printer has a built-in TrueType rasterizer
*** The printer handles the job accounting to its default settings:
- font is substituted by default font (e.g. Courier)
- job is aborted

Fig. A.1: Font handling when printing a PDF file with OPI

B The fonts we deliver

All the fonts we include in our software package are PostScript Type 1 fonts. They guarantee the most accurate printing results on PostScript printers.

After installation, the delivered fonts will automatically be available on the server and can be used for printing (and for layout generation if you happen to use ImageServer).

If you run PDF HandShake without an activation key (in demo mode), the fonts will be available for screen previews only – they will not be accessible during printing.

PDF HandShake contains 131 PostScript Type 1 fonts (out of the 136 Adobe PostScript 3 standard fonts). Altogether, the fonts need about 10 MB disk space on the server.

PostScript 3 fonts

Here are the PostScript 3 fonts we deliver:

- AlbertusMT-Italic
- AlbertusMT-Light
- AlbertusMT
- AntiqueOlive-Bold
- AntiqueOlive-Compact
- AntiqueOlive-Italic
- AntiqueOlive-Roman
- Apple-Chancery
- Arial-BoldItalicMT
- Arial-BoldMT

- Arial-ItalicMT
- ArialMT
- AvantGarde-Book
- AvantGarde-BookOblique
- AvantGarde-Demi
- AvantGarde-DemiOblique
- Bodoni-Bold
- Bodoni-BoldItalic
- Bodoni-Italic
- Bodoni-Poster
- Bodoni-PosterCompressed
- Bodoni
- Bookman-Demi
- Bookman-DemiItalic
- Bookman-Light
- Bookman-LightItalic
- Chicago
- Clarendon-Bold
- Clarendon-Light
- Clarendon
- CooperBlack-Italic
- CooperBlack
- Copperplate-ThirtyThreeBC
- Copperplate-ThirtyTwoBC
- Coronet-Regular
- Courier
- Courier-Bold
- Courier-BoldOblique
- Courier-Oblique
- Eurostile-Bold
- Eurostile-BoldExtendedTwo
- Eurostile-ExtendedTwo
- Eurostile
- Geneva
- GillSans-Bold

- GillSans-BoldCondensed
- GillSans-BoldItalic
- GillSans-Condensed
- GillSans-ExtraBold
- GillSans-Italic
- GillSans-Light
- GillSans-LightItalic
- GillSans
- Goudy-Bold
- Goudy-BoldItalic
- Goudy-ExtraBold
- Goudy-Italic
- Goudy
- Helvetica-Bold
- Helvetica-BoldOblique
- Helvetica-Condensed-Bold
- Helvetica-Condensed-BoldObl
- Helvetica-Condensed-Oblique
- Helvetica-Condensed
- Helvetica-Narrow-Bold
- Helvetica-Narrow-BoldOblique
- Helvetica-Narrow-Oblique
- Helvetica-Narrow
- Helvetica-Oblique
- Helvetica
- HoeflerText-Ornaments
- HoeflerText-Regular
- JoannaMT-Bold
- JoannaMT-BoldItalic
- JoannaMT-Italic
- JoannaMT
- LetterGothic-Bold
- LetterGothic-BoldSlanted
- LetterGothic-Slanted
- LetterGothic

- LubalinGraph-Book
- LubalinGraph-BookOblique
- LubalinGraph-Demi
- LubalinGraph-DemiOblique
- MonaLisa-Recut
- Monaco
- NewCenturySchlbk-Bold
- NewCenturySchlbk-BoldItalic
- NewCenturySchlbk-Italic
- NewCenturySchlbk-Roman
- NewYork
- Optima-Bold
- Optima-BoldItalic
- Optima-BoldOblique
- Optima-Italic
- Optima-Oblique
- Optima
- Oxford
- Palatino-Bold
- Palatino-BoldItalic
- Palatino-Italic
- Palatino-Roman
- StempelGaramond-Bold
- StempelGaramond-BoldItalic
- StempelGaramond-Italic
- StempelGaramond-Roman
- Symbol
- Times-Bold
- Times-BoldItalic
- Times-Italic
- Times-Roman
- TimesNewRomanPS-BoldItalicMT
- TimesNewRomanPS-BoldMT
- TimesNewRomanPS-ItalicMT
- TimesNewRomanPSMT

- Univers-Bold
- Univers-BoldExt
- Univers-BoldExtObl
- Univers-BoldOblique
- Univers-Condensed
- Univers-CondensedBold
- Univers-CondensedBoldOblique
- Univers-CondensedOblique
- Univers-Extended
- Univers-ExtendedObl
- Univers-Light
- Univers-LightOblique
- Univers-Oblique
- Univers
- ZapfChancery-MediumItalic
- ZapfDingbats

C Glossary

Bilevel

Bilevel images contain only two colors: *black* and *white*. “Lineart” is another term for bilevel images.

CCITT

Comité Consultatif International Téléphonique et Télégraphique, renamed to *ITU-T* in 1993. This group specified communication protocols for a particular class of devices (e.g. facsimile devices) and represents specific modes of compression (CCITT Group 3, CCITT Group 4).

CIE

Commission Internationale de l’Eclairage. The CIE is responsible for the definition of color models and the standardization of color descriptions. The Lab color space has been defined by the CIE.

CIELab

Lab colors are defined by the L-value for lightness and the co-ordinates defining the quantitative distance of a color from a reference white point. The Lab color space includes all visible colors and is device independent.

CMYK

Color printing is based on the CMYK color space. *Cyan*, *Magenta*, *Yellow*, and *Black* inks are mixed on paper to produce a given color. The definitions of the basic colors cyan, magenta, yellow, and black are slightly different in Europe (**Euro**), the US (**SWOP**), and Japan.

Color matching

The process of adjusting colors to achieve maximum similarity from the gamut of one color space to another. In practice, the color data delivered by a given device – e.g. a scanner – has to be transformed so that the colors can exactly be reproduced by a second device – e.g. a printer.

Color mode

See: **Bilevel**, **Grayscale**, **Indexed colors**, **RGB**, **CMYK**, **CIELab**, **Spot color**, **Multichannel**, **Multitones**.

ColorSync

ColorSync offers a programming interface to a fast computing engine which uses ICC profiles as parameters to perform color transformations between different devices. ColorSync has been co-developed by Apple Computer and Linotype-Hell.

Compression

See: **LZW**, **CCITT**, **JPEG**, **JPEG 2000**, **JBIG2**, **PackBits RLE**.

DeviceLink profile

The *International Color Consortium* (ICC) has defined ICC profiles which describe the color characteristics of graphics devices (such as scanners, monitors, and printers) and of abstract color spaces (such as CIELab D50, CIELab D65, and CIELuv). Additionally, the ICC has defined DeviceLink ICC profiles, which describe mappings from one particular color space to another particular color space, e.g. from an offset press CMYK color space to a gravure press CMYK color space. Many DeviceLink ICC profiles maintain special characteristics of the black channel and fulfill special conditions for color sums according to the drying capabilities of a particular printing process.

Downsampling

The process of transforming a high-resolution image into a low-resolution image.

dpi

See: **Resolution**

EPSF

Encapsulated PostScript File (EPSF) is an image format. EPSF files contain a PostScript program. When their program is being interpreted, the image is re-created. EPSF files may additionally contain an image preview of the image. Many illustration and DTP applications can create EPSF files. Please note that Mac-EPSF files and PC-EPSF files (for Windows computers) are not identical. PC-EPSF files contain LZW- compressed TIFF previews (instead of PICT previews) and, therefore, behave different in specific situations.

Note that there are two fundamentally different types of EPSF files:
Vector (object) based EPSF, created by drawing programs such as Illustrator, and *raster* (bitmap) EPSF files, created by programs such as Photoshop, or scanners.

EtherShare

Extremely fast, reliable, feature-rich Apple compatible file and print server. Includes TCP printer drivers supporting mDNS ("Bonjour") for easy printer selection on OS X clients.

Euro

Euro, formerly *Euroscale*, defines the European ink set for the process colors cyan, magenta, yellow, and black.

File format

See: **EPSF**, **JPEG**, **PICT**, **TIFF JPEG 2000**, **JBIG2**, **PDF**

Gamut, gamut mapping

The range of color that a given device can produce. Gamut mapping means re-defining the colors of a given device so that its gamut becomes (approximately) identical to that of a second device.

Grayscale

Grayscale images are a generalization of bilevel images. They contain black and white and different shades of gray.

Home directory

Private directory provided for each UNIX user. The home directory is the current directory as soon as you login to a server via Terminal.

ICC, ICC profiles

International Color Consortium. A group of vendors who defined the ICC profile format. This format is a cross-platform specification which allows third-party vendors to develop profile tools and applications supporting the ICC profile standard. The founding members of this consortium include: Adobe Systems Inc., Agfa-Gevaert N.V., Apple Computer Inc., Eastman Kodak Company, FOGRA (Honorary), Microsoft Corporation, Silicon Graphics Inc., Sun Microsystems Inc., and Taligent Inc. These companies have committed themselves to fully support this specification in their operating systems, platforms and applications.

Indexed colors

Images with indexed colors use colors from a given RGB color map. Every pixel of such an image contains a reference to a specific color in the map.

JPEG

Joint Photographic Experts Group JPEG is a file format and – at the same time – a mode of compression. Images are compressed by replacing several similar colors by one color only. Thus, some color data gets lost and cannot be recreated when the files are opened again.

JPEG 2000

A potential successor to JPEG with better compression and multiresolution images. JPEG 2000 gives reasonable quality down to 0.1 bits/pixel. *JPEG* quality drops dramatically below about 0.4 bits/pixel. In addition, JPEG 2000 also allows lossless compression.

JBIG2

A compression standard for lossless bilevel image coding. JBIG2 has been developed by the *Joint Bilevel Image Experts Group*.

Layout

ImageServer generates low-resolution representations of high-resolution image files or PDF files. These representations are called “layout” files.

LZW

LZW (*Lempel*, *Ziv*, *Welch* – the names of developers) is a lossless compression that can be used for different color modes and file formats. It collects repetitive patterns in a table and saves references to this table whenever possible.

Multichannel

A multichannel image contains CMYK channels and one or more channels different from CMYK, which are all treated as process colors. The most famous example is Pantone Hexachrome.

Multitones

Colorized **Grayscale** image.

PackBits RLE

Apple Mac OS compression, used e.g. in PICT files. See also: **RLE**.

PCShare

PCShare is a high-end SMB-based file server and print server software for Windows computers which are attached to the host via Ethernet, etc. Since PCShare is compatible with EtherShare, Windows users can share network printers and files with Mac and UNIX users, too.

PDF

Portable Document Format. File format developed by Adobe Systems, Inc. PDF files were originally designed for online reading on all platforms.

PICT

The PICT file format is a native classic Mac OS image format.

PostScript

Industry standard page-description language invented by Adobe and introduced in 1985 for printing documents that integrate text, graphics, images, and color.

PostScript fonts

In PostScript font files, the character information is given in PostScript code. PostScript printers require PostScript fonts or some sort of software that is able to transform non-PostScript fonts. For detailed information about the "font business", please read refer to A "About fonts".

PPD

PostScript Printer Description. A file format developed by Adobe Systems, Inc. PPD files contain information enabling software to produce the best results possible for each type of designated printer.

Rendering intent

The approach taken when a *Color Management Module* (CMM) maps or translates the colors of a source device to the gamut of a destination device. Each profile supports different rendering intents. Changing the rendering intent may lead to a different output result, even though you did not change the profile.

Resolution

The *dots per inch* (dpi) value of an image indicates its resolution. The dpi value of a given device (scanner, printer, etc.) defines its resolution capacity. Very clear and sharp images require input/output devices with a high resolution (about 300 dpi or more). Monitor resolutions usually range from 72 to 100 dpi.

RGB

Screens and monitors produce colors by means of *Red*, *Green* and *Blue* light (RGB). The light intensities make up a given color. Scanners also work with RGB colors. They read the amounts of red, green, and blue light that are reflected from an image (or transmitted if you scan transparent images). RGB images contain three components per pixel, namely a specific amount of red, green, and blue.

RIP

Raster Image Processor. Converts vector data (e.g. PostScript) to raster image pixel data. The RIP may be either an external unit or part of the output device itself. A PostScript laser printer for example contains its RIP.

RLE

Run-Length Encoding. Mode of compression that saves repetitive patterns only once and adds the number of repetitions. RLE is a lossless compression.

Root directory

The top-most directory on a UNIX computer is called the “root” directory. If you are logged in as “root”, you can access all other directories and subdirectories on the system.

Spooler

Set of programs which manage print jobs. A spooler acts as a buffer for the files that have been sent to an output device. In this documentation, “printer queue” may be used as a synonym for “spooler”.

Spot color

Exactly defined full tone color. Spot colors are printed on their own separation plates when separations are specified.

SWOP

Specifications for Web Offset Publications. US standards for color separation and color printing.

TIFF

Tagged Image File Format. Standard graphics file format for files that are to be exchanged among several applications and environments, including Mac, Windows, and UNIX.

TrueType font

TrueType is an outline font standard developed by Apple and Microsoft in the late 1980s as a competitor to Adobe's Type 1 fonts used in PostScript. It has become the most common format for fonts on both the Mac OS and Microsoft Windows operating systems.For detailed information about the "font business", please refer to A "About fonts".

WebShare

WebShare is a high-performance HELIOS file server which enables fast and secure real time file access via any web browser. Since WebShare is compatible with PCShare and EtherShare, web users can share network files with Mac, Windows, and UNIX users, too.

Index