

PDFlib, PDFlib+PDI, PDFlib Personalization Server (PPS) 9.1

What is PDFlib?

PDFlib is the leading developer toolbox for generating and manipulating files in the Portable Document Format (PDF). PDFlib's main targets are dynamic PDF creation on a Web server or any other server system, and to implement »Save as PDF« in existing applications. You can use PDFlib to dynamically create PDF documents from database contents, similar to dynamic Web pages. PDFlib has proven itself in a wide range of other use cases as well. Application programmers need only decent graphics or print output experience to be able to use PDFlib quickly. Since PDFlib frees you from the technicalities of the PDF file format, you can focus on acquiring the data and arranging text, graphics, and images on the page.

The PDFlib product family is available in three different flavors: PDFlib, PDFlib+PDI (PDF Import), and the PDFlib Personalization Server (PPS) with the PDFlib Block Plugin for Adobe Acrobat.

PDFlib

PDFlib offers all functions required to generate PDF documents with text, graphics, images, and interactive elements such as annotations or bookmarks. Use PDFlib for the following and many other tasks:

- ▶ add »Save as PDF« capability to your application
- ▶ create PDF documents on a Web server in real time
- ▶ create database reports in PDF
- ▶ take advantage of advanced typography and full Unicode and encoding support for text output
- ▶ advanced color management functionality
- ▶ convert TIFF, JPEG, or other image formats as well as SVG graphics to PDF
- ▶ automatically format tables with all kinds of cell contents
- ▶ create PDF/X-1/3/4/5 documents for commercial printing
- ▶ create PDF/A-1/2/3 for archiving
- ▶ Create PDF/VT for transactional printing
- ▶ create Tagged PDF and PDF/UA for accessibility

PDFlib+PDI (PDF Import)

PDFlib+PDI includes all PDFlib functions, plus the PDF Import Library (PDI). With PDI you can open existing PDF documents and incorporate some pages into the PDFlib output. Use PDFlib+PDI for all PDFlib tasks plus the following:

- ▶ impose multiple PDF pages on a single sheet for printing
- ▶ add text, such as headers, footers, stamps, or page numbers to existing PDF pages
- ▶ place images, e.g. company logo, on existing pages
- ▶ add barcodes to existing PDF pages
- ▶ assemble existing PDF pages
- ▶ add content to PDF/A, PDF/X or PDF/UA documents

PDFlib Personalization Server (PPS) and PDFlib Block Plugin

The PDFlib Personalization Server (PPS) includes PDFlib+PDI plus additional functions for variable data processing using PDFlib Blocks. PPS makes applications independent from layout changes. The designer creates the page layout and converts it to PDF. She takes into account areas as placeholders for variable text and images. In Acrobat she drags a rectangular Block for each area using the PDFlib Block Plugin. Each Block contains a variety of Block properties, such as font size, color, image scaling. The PDFlib Block Plugin offers a Preview feature which shows the results of filling Blocks according to the specified properties.

The programmer writes code to fill PDFlib Blocks with text, images, vector graphics or PDF pages. He doesn't need to know the formatting or position of a Block. Use PPS for all PDFlib+PDI tasks plus the following:

- ▶ customize direct mailings with text and images
- ▶ fill templates for transactional and statement processing
- ▶ personalize promotional material with address data
- ▶ generate individual parts catalogs from a database
- ▶ produce customized documentation for multiple products

What's new in PDFlib 9?

PDF/A-2 and PDF/A-3

PDFlib supports two additional parts of the PDF/A standard for archiving. PDF/A-2 is based on PDF 1.7 and supports transparency, JPEG 2000 compression, layers, and many other features. While PDF/A-2 allows embedding of PDF/A-1 and PDF/A-2 documents, PDF/A-3 allows embedding of arbitrary file types.

Tagged PDF and PDF/UA

Creating Tagged PDF is much easier through various convenience features, such as abbreviated tagging and automatic tagging of Artifacts. PDFlib's table formatter automatically tags tables. Tagged PDF documents including structure elements can be imported with PDI.

Accessible documents can be created according to the PDF/UA standard (Universal Accessibility). PDF/UA is based on PDF 1.7 and improves Tagged PDF for accessibility similar to WCAG 2.0 (Web Content Accessibility Guidelines) in the Web world.

PDF/X

PDFlib 9 supports PDF/X-1a and PDF/X-3 as well as PDF/X-4/4p and PDF/X-5g/pg. PDFlib 9.1 adds support for PDF/X-5n for exchange of n-colorant production files, e.g. in the packaging industry.

PDF/VT

PDF/VT is a standard for optimized PDF for variable and transactional printing. PDFlib can create output which conforms to PDF/VT-1, PDF/VT-2 or PDF/VT-2s according to ISO 16612-2 for Variable Document Publishing (VDP). Document Part Metadata (DPM) can be attached according to the PDF/VT standard.

Scalable Vector Graphics (SVG)

PDFlib imports vector graphics in the SVG format. SVG is the standard format for vector graphics on the Web, and is supported by all mainstream browsers. PDFlib 9.1 introduces support for ICC profiles, CMYK and spot color in SVG.

Font handling and text output

PDFlib's font engine and text processing have been enhanced in several ways:

- ▶ ideographic variation sequences (IVS) for CJK variant glyphs;
- ▶ WOFF fonts (Web Open Font Format), a new container format for TrueType and OpenType fonts specified by the W3C
- ▶ SVG fonts, i.e. vector fonts specified in SVG format
- ▶ CEF fonts (Compact Embedded Font), a variant of OpenType used for embedding fonts in SVG graphics
- ▶ support for all Unicode normalization forms (NFC, NFKC etc.)
- ▶ automatically create UPR font configuration files with all fonts found in an arbitrary number of directories

Import PDF documents with PDFlib+PDI

The following features are new in the PDF Import library PDI:

- ▶ Tagged PDF documents including structure elements can be imported.
- ▶ Layer definitions can be imported.

PDFlib Personalization Server (PPS) and Block Plugin

The following features are new in PPS:

- ▶ The new Block type »Graphics« can be used to fill PDFlib Blocks with SVG graphics.
- ▶ PDFlib Blocks can be imported with PPS in addition to the Block Plugin.

Create PDFlib Blocks programmatically

In addition to creating PDFlib Blocks interactively with the PDFlib Block Plugin, PDFlib Blocks can be created programmatically with PPS. Existing PDFlib Blocks in imported documents can be copied to the generated PDF output. These features enable advanced document composition workflows where templates for PPS can themselves be built programmatically.

PDF Object Creation API (POCA)

POCA provides a set of methods for creating low-level PDF objects which are included in the generated PDF output. POCA can be used for the following purposes:

- ▶ create Document Part Metadata (DPM) for PDF/VT
- ▶ programmatically create PDFlib Blocks for use with PPS

Multimedia contents

PDFlib can create rich media annotations with Sound, Movie or 3D content. The multimedia content can be controlled with JavaScript and PDF actions. The following new multimedia features are available:

- ▶ rich media annotations
- ▶ rich media execute actions

Enhanced encryption algorithm

PDFlib supports PDF encryption according to Acrobat X/XI/DC. This encryption scheme is based on AES-256 and is specified in PDF 1.7 Adobe extension level 8 and PDF 2.0 according to ISO 32000-2.

What's new in PDFlib 9.1?

PDFlib/PDFlib+PDI/PPS 9.1 introduces new features related to color handling:

- ▶ Support for n-colorant color spaces (DeviceN and NChannel)
- ▶ PDF/X-5n for exchange of n-colorant production files, e.g. in the packaging industry
- ▶ SVG color extension for ICC profiles, spot and DeviceN color as well as Gray/RGB/CMYK device color
- ▶ PANTONE Extended Gamut Coated (XGC) spot colors and PANTONE Plus 2016 update
- ▶ Color gradients with an arbitrary number of stop colors for enhanced color blends
- ▶ Color gradients between different spot colors, e.g. a blend of PANTONE colors

Common Features in PDFlib, PDFlib+PDI, and the PDFlib Personalization Server

| | |
|---|---|
| PDF flavors | PDF 1.4 – PDF 1.7 extension level 8 and PDF 2.0 (Acrobat 5–DC) |
| | Linearized (web-optimized) PDF for byteserving over the Web |
| | High-volume output and arbitrary PDF file size (beyond 10 GB) |
| ISO standards for PDF | ISO 32 000-1: standardized version of PDF 1.7 |
| | ISO 32 000-2 (draft): PDF 2.0 |
| | ISO 15 930: PDF/X-1/3/4/5 for the graphic arts industry |
| | ISO 19 005-1/2/3: PDF/A-1/2/3 for archiving |
| | ISO 16612-2: PDF/VT-1/2 for variable and transactional printing |
| | ISO 14289-1: PDF/UA-1 for universal accessibility |
| Fonts | TrueType (TTF and TTC) and PostScript Type 1 fonts |
| | OpenType fonts with PostScript or TrueType outlines (TTF, OTF) |
| | WOFF fonts (Web Open Font Format), a W3C-specified container format for fonts on the Web |
| | CEF fonts (Compact Embedded Font), a variant of OpenType used for embedding fonts in SVG |
| | SVG fonts, i.e. fonts which use the SVG format for describing glyph outlines |
| | Support for dozens of OpenType layout features for Western and CJK text output, e.g. ligatures, small caps, old-style numerals, swash characters, simplified/traditional forms, vertical alternates |
| | Directly use fonts which are installed on Windows or OS X/macOS (»host fonts«) |
| | Font embedding for all font types; subsetting for TrueType, OpenType, and Type 3 fonts |
| | User-defined (Type 3) fonts for bitmap fonts or custom logos |
| | EUDC and SING fonts (glyphlets) for CJK Gaiji characters |
| Text output | Fallback fonts (pull missing glyphs from an auxiliary font) |
| | Text output in different fonts; underlined, overlined, and strikeout text |
| | Glyphs in a font can be addressed by numerical value, Unicode value, or glyph name |
| | Kerning for improved character spacing |
| | Artificial bold, italic, and shadow text |
| | Create text on a path |
| Accessibility | Configurable replacement of missing glyphs |
| | Create Tagged PDF for accessibility, page reflow, and improved content repurposing |
| | Simplified tagging by directly supplying tagging information to all content placement functions |
| | Automatic table and artifact tagging |
| | PDF/UA for universal accessibility, WCAG 2.0 (Web Content Accessibility Guidelines) |
| Internationalization | Additional structure element types and attributes |
| | Unicode strings for page content, interactive elements, and file names; UTF-8, UTF-16, and UTF-32 formats; support for all Unicode normalization forms |
| | Support for a variety of 8-bit and legacy multi-byte CJK encodings (e.g. Shift-JIS; Big5) |
| | CJK fonts and CMaps for Chinese, Japanese, and Korean text |
| | Ideographic variation sequences (IVS) for CJK variant glyphs |
| | Vertical writing mode for Chinese, Japanese, and Korean text |
| | Character shaping for complex scripts, e.g. Arabic, Thai, Devanagari |
| Bidirectional text formatting for right-to-left scripts, e.g. Arabic and Hebrew | |
| SVG vector graphics | Import vector graphics in SVG format; ICC profiles, CMYK and spot colors in SVG* |
| Images | Embed BMP, GIF, PNG, TIFF, JBIG2, JPEG, JPEG 2000, and CCITT raster images |
| | Query image information (pixel size, resolution, ICC profile, clipping path, etc.) |
| | Interpret clipping paths in TIFF and JPEG images |
| | Interpret alpha channel (transparency) in TIFF and PNG images |
| | Image masks (transparent images with a color applied), colorize images with a spot or DeviceN* color |

| | |
|---|---|
| Color | Grayscale, RGB (numerical, hexadecimal strings, HTML color names), CMYK, CIE L*a*b* color |
| | Integrated support for PANTONE® (incl. PANTONE+) and HKS® colors |
| | DeviceN (n-colorant) color space based on process or spot colors* |
| | User-defined spot colors |
| Color management | Color gradients between an arbitrary number of process colors or spot colors* |
| | ICC-based color with ICC profiles; support for ICC 4 profiles |
| | Rendering intent for text, graphics, and raster images |
| Archiving | ICC profiles as output intent for PDF/A and PDF/X; multi-colorant (xCLR) profiles for PDF/X-5n* |
| | PDF/A-1a/1b, PDF/A-2a/b/u and PDF/A-3a/b/u |
| Graphic arts | XMP extension schemas for PDF/A |
| | PDF/X-1a, PDF/X-3, PDF/X-4, PDF/X-4p, PDF/X-5p, PDF/X-5pg, PDF/X-5n* |
| | Embedded or externally referenced output intent ICC profile |
| Variable Document Printing (VDP) | External graphical content (referenced pages) for PDF/X-5p and PDF/X-5pg |
| | Settings for overprint, text knockout, etc. |
| Textflow Formatting | PDF/VT-1, PDF/VT-2, and PDF/VT-2s for variable and transactional printing |
| | Format text into one or more rectangular or arbitrarily shaped areas with hyphenation (user-supplied hyphenation points required), font and color changes, justification methods, tabs, leaders, control commands |
| Table formatting | Advanced line-breaking with language-specific processing |
| | Flexible image placement and formatting |
| | Wrap text around images or image clipping paths |
| | Table formatter places rows and columns, and automatically calculates their sizes according to a variety of user preferences. Tables can be split across multiple pages. |
| Vector graphics | Table cells can hold single- or multi-line text, images, vector graphics, PDF pages, path objects, annotations, and form fields |
| | Table cells can be formatted with ruling and shading options |
| | Flexible stamping function |
| | Matchbox concept for referencing the coordinates of placed images or other objects |
| | Common vector graphics primitives: lines, curves, arcs, ellipses, rectangles, etc. |
| Layers | Smooth shadings (color blends), between multiple process or spot colors*, pattern fills and strokes |
| | Transparency (opacity) and blend modes |
| | External graphical content (Reference XObjects) for variable document printing |
| | Reusable path objects and clipping paths imported from images |
| Security | Optional page content which can selectively be displayed |
| | Annotations and form fields can be placed on layers |
| | Encrypt PDF document or attachments with 128/256-bit AES or RC4 128-bit encryption |
| Interactive elements | Unicode passwords |
| | Document permission settings (e.g. printing or copying not allowed) |
| | Create form fields with all field options and JavaScript |
| | Create barcode form fields |
| | Create actions for bookmarks, annotations, page open/close and other events |
| | Create bookmarks with a variety of options and controls |
| Multimedia | Page transition effects, such as shades and mosaic |
| | Create all PDF annotation types, such as PDF links, launch links (other document types), Web links |
| | Named destinations for links, bookmarks, and document open action |
| | Create page labels (symbolic names for pages) |
| Multimedia | Embed 3D animations in PDF |
| | Embed Sound, Movie and 3D content in PDF and control it with JavaScript |
| | Flash-based navigators for custom presentation of PDF portfolios |

| | |
|---|--|
| Georeferenced PDF | Create PDF with geospatial reference information |
| Metadata | Document information: common fields (Title, Subject, Author, Keywords) and user-defined fields Create XMP metadata from document info fields or from client-supplied XMP streams Process XMP image metadata in TIFF, JPEG, JPEG 2000 images and SVG graphics |
| Programming | Language bindings for C, C++, Cobol, COM, Objective C, Java, .NET, Perl, PHP, Python, REALbasic/Xojo, RPG, Ruby Virtual file system for supplying data in memory, e.g., images from a database Generate PDF documents on disk file or directly in memory (for Web servers) |
| <i>* New in PDFlib/PDFlib+PDI/PPS 9.1</i> | |

Additional Features in PDFlib+PDI and the PDFlib Personalization Server

| | |
|------------------------|--|
| PDF input (PDI) | Import pages from existing PDF documents Import all PDF versions up to PDF 1.7 extension level 8 (Acrobat X/XI/DC) and PDF 2.0 Import documents which are encrypted with any of PDF's standard encryption algorithms Query information about imported pages Clone page geometry of imported pages (e.g. BleedBox, TrimBox, CropBox) Delete redundant objects (e.g. identical fonts) across multiple imported PDF documents Repair malformed input PDF documents Copy PDF/A or PDF/X output intent from imported PDF documents Import Tagged PDF documents including structure hierarchy Import layer definitions (optional content) |
| pCOS interface | pCOS interface for querying details about imported PDF documents |

Additional Features in the PDFlib Personalization Server (PPS)

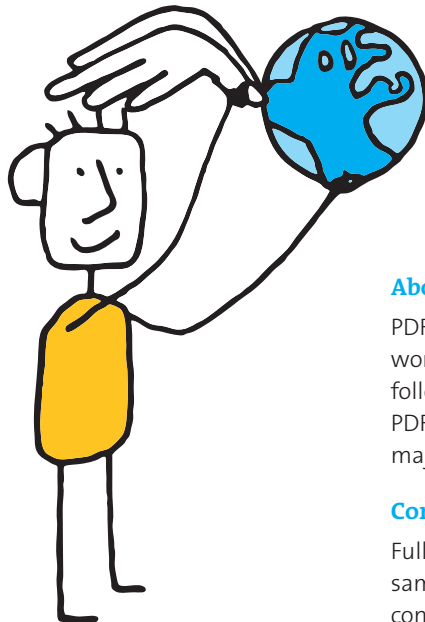
| | |
|---|--|
| Variable Document Printing (VDP) | PDF personalization with PDFlib Blocks for text, image, PDF data, or SVG vector graphics Create PDFlib Blocks programmatically with PPS Copy PDFlib Blocks from imported documents |
| PDFlib Block Plugin | PDFlib Block plugin for creating PDFlib Blocks interactively in Acrobat on Windows and OS X/macOS Preview PPS Block filling in Acrobat Copy Blocks to Preview file Snap-to-grid for interactively creating or editing Blocks in Acrobat Clone PDF/X or PDF/A properties of the Block container Convert PDF form fields to PDFlib Blocks for automated filling Textflow Blocks can be linked so that one Block holds the overflow text of a previous Block PANTONE® and HKS® spot color names integrated in the Block plugin Support for Retina displays on OS X/macOS* |
| <i>*New in Block Plugin 5.10</i> | |

Supported Development Environments

PDFlib is everywhere – it runs on practically all computing platforms. We offer 32- and 64-bit variants for all common flavors of Windows, OS X/macOS, Linux and Unix, as well as for IBM iSeries/i5 and zSeries mainframes.

The PDFlib core is written in highly optimized C code for maximum performance and small overhead. Via a simple API (Application Programming Interface) the PDFlib functionality is accessible from a variety of development environments:

- ▶ COM for use with VB, ASP, etc.
- ▶ C and C++
- ▶ Cobol (IBM zSeries)
- ▶ Java, including servlets and JSP
- ▶ .NET for use with C#, VB.NET, ASP.NET, etc.
- ▶ Objective-C (OS X/macOS, iOS)
- ▶ Perl
- ▶ PHP
- ▶ Python
- ▶ REALbasic/Xojo
- ▶ RPG (IBM iSeries/i5)
- ▶ Ruby, including Ruby on Rails



Benefits of using PDFlib Software

Rock-solid Products

Tens of thousands of programmers worldwide are working with our software. PDFlib products meet all quality and performance requirements for server deployment. All products are suitable for robust 24x7 server deployment and unattended batch processing.

Speed and Simplicity

PDFlib products are incredibly fast – up to thousands of pages per second. The programming interface is straightforward and easy to learn.

PDFlib Products all over the World

Our products support all international languages as well as Unicode. They are used by customers in all parts of the world.

Professional Support

If there's a problem, we will try to help. We offer commercial support to meet the requirements of your business-critical applications. By adding support you will have access to the latest versions, and have guaranteed response times should any problems arise.

Licensing

We offer various licensing programs for server licenses, integration and site licenses, and source code licenses. Support contracts for extended technical support with short response times and free updates are also available.

About PDFlib GmbH

PDFlib GmbH is completely focused on PDF technology. Customers worldwide use PDFlib products since 1997. The company closely follows development and market trends, such as ISO standards for PDF. PDFlib GmbH products are distributed all over the world with major markets in North America, Europe, and Japan.

Contact

Fully functional evaluation versions including documentation and samples are available on our Web site. For more information please contact:



PDFlib GmbH

Franziska-Bilek-Weg 9, 80339 München, Germany
 phone +49 • 89 • 452 33 84-0, fax +49 • 89 • 452 33 84-99
 sales@pdfliib.com
 www.pdfliib.com