datasheet

PDFlib, PDFlib+PDI, PDFlib Personalization Server (PPS) 10

What is PDFlib?

PDFlib is a well-established developer component for generating and manipulating PDF documents. PDFlib's main targets are dynamic PDF creation on a Web server and PDF creation in any kind of server, desktop or mobile application. You can use PDFlib to dynamically create PDF documents from database contents, similar to dynamic Web pages. Since 1997 PDFlib has proven itself in a wide range of use cases and development environments. Application programmers use PDFlib to create PDF output with text, images, graphics and interactive elements.

PDFlib is in active use by tens of thousands of users worldwide. The large-scale deployment helps us to improve and fine-tune the API, the internal implementation and the generated PDF output. Conformance to the relevant ISO standards for PDF, PDF/A, PDF/X, PDF/UA and PDF/VT as well as compatibility with common PDF viewers have high priority. PDFlib supports all features provided by the PDF format.

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 methods for generating PDF documents with text, graphics, images, and interactive elements such as annotations or bookmarks. The PDFlib API supports PDF creation on different levels which can freely be mixed.

Low-level APIs allow you to place individual items on the page, using formatting information provided by the application. An application which already determined the layout (e.g. for screen display) can directly place text, images and other objects on the PDF page.

High-level APIs support powerful formatting features:

- ▶ Single-line text output with various formatting controls.
- ▶ Textflow formatter for placing arbitrary amounts of text in one or more columns or pages. The text can be formatted in non-rectangular shapes, flow around images and contain interactive elements and structure information for Tagged PDF.

- ▶ The flexible table formatter supports all types of contents in table cells like text, images, graphics and annotations or form fields. Large tables can be placed on multiple pages. Details of table formatting are controlled by a variety of options.
- ▶ Image and SVG placement according to various fitting and formatting methods.

PDFlib+PDI (PDF Import)

PDFlib+PDI includes all PDFlib functions plus the PDF Import Library PDI which incorporates existing PDF pages and interactive elements in the generated PDF output. Use PDFlib+PDI for all PDFlib tasks plus the following:

- ▶ impose multiple PDF pages on a single sheet
- ▶ add text, such as headers, footers, stamps, or page numbers to existing PDF pages
- ▶ place images, e.g. company logo, on existing pages
- ▶ rearrange, assemble or amend PDF pages while maintaining PDF/A, PDF/X or PDF/UA conformance

PDFlib Personalization Server (PPS) and PDFlib Block Plugin

The PDFlib Personalization Server (PPS) includes PDFlib+PDI plus additional methods for variable data processing using PDFlib Blocks. PPS makes applications independent from layout changes. The designer creates a layout PDF and marks areas as placeholders for variable text, images and graphics using the PDFlib Block Plugin for Acrobat. Each Block contains a variety of Block properties such as font size, color, image scaling options etc. The PDFlib Block Plugin offers a Preview feature which shows the results of filling Blocks according to the specified properties.

The developer 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 mail merge with text and images, template filling for transactional and statement processing, personalizing promotional material, generating individual parts catalogs from a database or producing customized documentation for multiple products..

What's new in PDFlib 10?

Color Fonts and Emoji

OpenType color fonts allow glyph descriptions with color and transparency. They are especially popular for emoji fonts. PDFlib supports color fonts in the formats SVG, Microsoft COLR and Apple sbix along with OpenType features, variation selectors and emoji variation sequences, e.g. for changing skin color or gender of a particular emoji.

Import Form Fields and Annotations with PDFlib+PDI

PDFlib+PDI traditionally imported only the page contents, but ignored interactive elements. Links and other annotation types, form fields, actions and JavaScript are imported along with the page contents to ensure that interactive documents retain their functionality. Interactive elements remain functional even if the imported pages are placed in different order or if multiple imported pages are placed on the same output page. Annotation rectangles are transformed appropriately if the page containing the annotation has been scaled or rotated. Annotation and form field import are compatible with PDF/A, PDF/UA and PDF/X.

Annotations

Creation of all types of annotations (comments) has been overhauled, extended and simplified. Annotation appearance streams are created automatically; annotations are supported in PDF/A mode as far as allowed by the standard.

Form filling

Imported form fields can be filled, e.g. to populate text fields. The contents and visual appearance (color, border etc.) of imported form fields can be modified or kept in the original state.

Multimedia

Video and sound files can be used in PDF with Screen annotations and Rendition actions. These are more powerful than Sound and Movie annotations and don't require the deprecated Flash technology which is no longer supported in PDFlib 10 nor Acrobat DC.

PDF 2.0 Support

PDFlib supports new features in PDF 2.0 according to ISO 32000-2:2017 and the dated revision ISO 32000-2:2020. This includes new structure element types and nesting rules for Tagged PDF, graphical features, encryption, interactive elements, document and page-level output intents and many other areas. PDFlib+PDI and PPS process PDF 2.0 documents as well. Features marked as deprecated in PDF 2.0 are also deprecated or removed in PDFlib.

Color Management

The industry-standard sRGB ICC profile is automatically applied to RGB images which don't include an embedded ICC profile. This facilitates faithful color reproduction and use of RGB images in PDF/A or PDF/X documents.

Faithful color reproduction of imported PDF pages and SVG graphics with transparency has been simplified by automatic creation of transparency groups.

Optimized PDF Output

ICC profiles and other objects in PDF are cached and duplicates are detected. In combination with optimized content streams with fewer operators this results in significant PDF file size reductions. Large PDF documents beyond 10 GB can be created in combination with optimization and linearization.

SVG Import

Several details of SVG processing have been improved including certain CSS directives. Remote resources, e.g. fonts and images, can be fetched over the network.

Font Support and Text Handling

Enhancements in addition to color font support:
► Fonts in WOFF2 format can be used.
► Improved support for OpenType layout features.
► Improved Type 3 font creation.

Convenience Features

Many existing API methods have been enhanced with convenience options and simplified procedures for common tasks. For example, graphics state properties can be supplied directly via options to many API methods without the need for creating and applying a gstate object. Path objects can be drawn inside a matchbox which has been created earlier, e.g. via Textflow or Table formatting.

The default PDF output compatibility is PDF 1.7 Extension Level 8, the file format of Acrobat X/XI/DC.

Updated Pantone® Color Libraries

The integrated Pantone spot color database has been extended to cover the latest additions to the Solid Coated and Solid Uncoated color libraries. Spot colors which are no longer supported by Pantone, Inc. have been removed, e.g. the hexachrome color library.

pCOS Interface

The pCOS interface (which is included in PDFlib+PDI and PPS) supports additional pseudo objects for querying PDF details, e.g. new PDF standards, form fields, signatures and ICC profiles.

Tagged PDF and PDF/UA

Automatic Table tagging tags tables which span multiple pages correctly with a single Table element. Links spanning multiple lines are created as a single Link annotation with multiple rectangles. Structure element tags can be supplied inline in Textflow along with formatting options.

PDF 2.0 introduces a strict model for the relationship of structure elements. PDFlib supports new PDF 2.0 structure types and attributes and enforces the new structure element nesting rules. Tag namespaces (tagsets) and structure destinations are supported.

Untagged documents can be imported into PDF/UA as Artifact which simplifies handling of existing decorative content which is not available as PDF/UA.

PDFlib Personalization Server (PPS) and Block Plugin

The PDFlib Block Plugin for Acrobat received various usability improvements including a filter for temporarily reducing the number of visible Blocks for screen and Preview. Blocks can easily be moved by small or large amounts. Additional Block properties support all

graphics state options including overprint settings. Structure elements (tags) are available as Block property and Preview supports PDF/UA cloning.

Networking

Remote resources can be fetched from the network with a new PDFlib API method, i.e. independently of the means provided by the environment. SVG processing automatically retrieves remote resources such as fonts or images.

Language Bindings

All language bindings have been updated to the latest versions. The Perl, PHP and Ruby bindings are UTF-8-enabled by default. The C++ binding supports the Unicode string types u8string/u16string/u32string introduced with C++11 and C++20.

Miscellaneous Improvements

New features in various areas:

▶ Type 3 fonts may contain a mixture or colorized and non-colorized glyphs.
▶ Additional appearance options for path objects.

Modernized Code Base and Code Security

The code has been reviewed and improved for robustness; unnecessary code has been removed, e.g. workarounds for outdated versions of Acrobat and RIPs. Removal of deprecated features also resulted in smaller and cleaner code. Third-party libraries used in PDFlib have been updated to the latest versions to take advantage of security improvements and vulnerability protection.

Deprecated and Removed Methods

While a number of PDFlib API methods have been declared as deprecated in earlier versions, API methods so far have never been removed, but were still offered as migration aid. PDFlib 10 removes all API methods which have been declared as deprecated in PDFlib 9 or earlier versions. In most cases simple one-to-one replacements are available.

Similarly, deprecated options have been removed. In most cases the underlying functionality is still supported and simple replacement options are available.

Some features are no longer available at all, either because there is no demand in the marketplace (e.g. Reference XObjects) or because they have been declared as deprecated in PDF 2.0, e.g. OPI and Flash.

PDFlib 9 applications which didn't use any deprecated features can be used unmodified with PDFlib 10. If your application uses deprecated features please take a look at the PDFlib Migration Guide. This document contains detailed lists of deprecated and removed API methods, options and features as well as notes regarding the required changes in application code.

The PDFlib Migration Guide also explains how to identify deprecated features with PDFlib 9 before migrating to PDFlib 10.

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[1]
	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: PDF 2.0 including dated revision ISO 32000-2:2020[1] PDF 2.0
	ISO 15 930: PDF/X-3/4/5 for data exchange the graphic arts industry
	ISO 19 005-1/2/3: PDF/A-1/2/3 for archiving
	ISO 16612-2: PDF/VT-1 for variable and transactional printing
	ISO 14289-1: PDF/UA-1 for universal accessibility
Fonts	OpenType fonts with PostScript or TrueType outlines (TTF, TTC, OTF, OTC)
	OpenType color fonts in the formats SVG, COLR (Microsoft) and sbix (Apple)[1]
	WOFF and WOFF2[1] fonts (Web Open Font Format)
	User-defined (Type 3) fonts for bitmap fonts or custom logos
	Support for of OpenType layout features[1] for Western and CJK text output, e.g. ligatures, small caps, old-style numerals, swash characters, simplified/traditional forms, vertical alternates
	Access fonts which are installed on Windows or macOS
	Font embedding and subsetting
	EUDC and SING fonts (glyphlets) for CJK Gaiji characters
	Fallback fonts: substitute missing glyphs with glyphs from another font
Text output	Text output in different fonts; underlined, overlined, and strikeout text
	Glyphs in a font can be addressed by numerical value, Unicode or glyph name
	Kerning for improved character spacing
	Variation selectors and emoji variation sequences[1]
	Artificial bold, italic, and shadow text
	Text on a path
	Flexible stamping method
Textflow Formatting	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, tags[1]
	Advanced line-breaking with language-specific processing, e.g. for Thai
	Flexible image placement and formatting
	Wrap text around images or image clipping paths
Table formatting	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.
	Table cells can hold text, images, SVG graphics, PDF pages, path objects, annotations and form fields
	Table cells can be formatted with ruling and shading options
	Matchbox concept for referencing the coordinates of placed images or other objects
Interactive elements	Form fields[1] with all field options and JavaScript (todo hyperlink for footnotes)
	PDF annotations[1] (comments) such as PDF links, launch links (other document types), Web links
	PDF 2.0 annotation attributes, e.g. opacity[1]
	Non-rectangular annotations and Link annotations with multiple rectangles[1]
	Create actions for bookmarks, annotations, page open/close and other events
	Create bookmarks with a variety of options and controls
	Page transition effects, such as shades and mosaic
	Named destinations for links, bookmarks, and document open action
	Create page labels (symbolic names for pages)
Multimedia	Embed 3D animations in PDF and control it with JavaScript
	Sound and video can be embedded or referenced from an external file or Web server with Screen annotations and Rendition actions[1]

Internationalization	Support for Unicode 14 including supplementary characters outside the Basic Multilingual Plane (BMP) and Unicode normalization forms
	CJK fonts 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, CSS[1], network resources[1]
Images	Load BMP, PNG, TIFF, JBIG2, JPEG and JCCITT
	PEG 2000 raster images
	Query image information (pixel size, resolution, ICC profile, clipping path, etc.)
	Use clipping path in TIFF and JPEG images
	Use 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, HTML color names), CMYK, CIE L*a*b* color
	Integrated for PANTONE® and HKS® colors[1]
	DeviceN (n-colorant) color space based on process or spot colors
	User-defined spot color
	Color gradients (smooth shadings) between process colors or spot colors, pattern fills and strokes
Color management	ICC-based color with ICC profiles[1]
	Rendering intent for text, graphics, and raster images
	Embedded or externally referenced ICC profiles as output intent for PDF 2.0, PDF/A and PDF/X
	Multicolor (xCLR) externally referenced output intent ICC profiles for PDF/X-5n
	Output intent ICC profiles on document or page level for PDF 2.0
Archiving	PDF/A-1a/1b, PDF/A-2a/b/u and PDF/A-3a/b/u including XMP extension schemas
Graphic arts	PDF/X-3, PDF/X-4, PDF/X-4p, PDF/X-5n
	Settings for overprint, text knockout, black point compensation[1], etc.
Accessibility	Tagged PDF for accessibility
	Tagging of interactive elements, e.g. annotations and form fields[1]
	Automatic table[1] and artifact tagging
	PDF/UA-1 for universal accessibility
	Tagging in Textflow[1]
	PDF 2.0 tag namespaces (tagsets)[1]
	PDF 2.0 structure types, attributes and tag nesting rules[1]
	Structure destinations, i.e. links point to a specific structure element[1]
Variable Document Printing (VDP)	PDF/VT-1 for variable and transactional printing
Vector graphics	Common vector graphics primitives[1]: lines, curves, arcs, ellipses, rectangles, etc.
	Smooth shadings (color blends) between multiple process or spot colors, pattern fills and strokes
	Transparency (opacity) and blend modes
	Reusable path objects[1] and clipping paths imported from images
Layers	Optional page content which can selectively be displayed
	Annotations and form fields can be placed on layers
Security	Encrypt PDF document or attachments
	Unicode passwords
	Document permission settings, e.g. printing or copying not allowed
Georeferenced PDF	PDF with geospatial reference information
Metadata	Document information: common fields (Title, Subject, Author, Keywords) and user-defined fields
	Synchronize XMP metadata from document info fields or XMP streams
	User-supplied custom XMP metadata

	Process XMP image metadata in TIFF, JPEG, JPEG 2000 and SVG
Programming	Language bindings for C, C++, Java, .NET, Objective C, Perl, PHP, Python, RPG, Ruby
	Helper method for downloading content from the Internet[1]
	Virtual file system for supplying data in memory, e.g., images from a database
	Generate PDF documents on disk file or directly in memory

[1] *New or considerably improved in PDFlib 10*

Additional Features in PDFlib+PDI and the PDFlib Personalization Server (PPS)

PDF input (PDI)	Import pages from existing PDF documents
	Import all PDF versions up to PDF 2.0
	Import annotations and form fields including actions[1]; annotation rectangles are transformed, e.g. scaled or rotated, to match the placed page; link annotations remain functional even if pages are arranged in different order
	Fill imported form fields[1]
	Import JavaScript and other actions on document or page level[1]
	Import encrypted documents
	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
	Tagged PDF content aggregation: import pages from Tagged PDF documents including structure elements
	Import layer definitions (optional content)
	Improved performance for importing content from very large documents[1]
pCOS interface	pCOS interface for querying details about imported PDF documents (see separate pCOS datasheet)
	Additional pCOS pseudo objects for querying PDF details, e.g. PDF standards, form fields, signatures and ICC profiles[1]

[1] *New or considerably improved in PDFlib+PDI 10*

Additional Features in the PDFlib Personalization Server (PPS)

Variable Document Printing (VDP)	PDF personalization with PDFlib Blocks which can be filled with text, image, PDF data, or SVG graphics
	Create PDFlib Blocks programmatically with PPS
	Copy PDFlib Blocks from imported documents
PDFlib Block Plugin	PDFlib Block Plugin for creating PDFlib Blocks interactively with Adobe Acrobat
	Preview PPS Block filling in Acrobat
	Filter Blocks for Display and Preview based on Block name, type, size or position[1]
	Copy Blocks to Preview file
	Snap-to-grid for interactively creating or editing Blocks in Acrobat
	Move Blocks by small or large increments for simple yet precise positioning[1]
	All graphics state properties, e.g. overprint settings, available as Block properties[1]
	Clone PDF/A, PDF/UA or PDF/X status 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
	Tagging for Block contents[1]

[1] *New or considerably improved in PPS 10 and Block Plugin 6*

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, macOS, Linux and Unix, as well as for IBM System i and IBM Z.

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

- ► C and C++
- ► Java
- ► .NET
- ► Objective-C (macOS and iOS) and Swift
- ► Perl
- ► PHP
- ► Python
- ► RPG (IBM System i)
- ► Ruby

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 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 are using our products since 1997. In 2006 we were one of the founding members of the PDF Association (formerly PDF/A Competence Center). 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 Europe, North America and Japan.

Contact

Fully functional evaluation versions 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
www.pdflib.com   sales@pdflib.com