A Technical Introduction to PDF/A

PDFlib Whitepaper

The PDF/A Family of Archiving Standards

PDF/A is targeted at reliable long-time preservation of digital documents with text, raster images and vector graphics as well as associated METADATA. The PDF/A format specified in the ISO 19005 standard strives to provide a consistent and robust subset of PDF which can faithfully be reproduced even after a long archiving period, or used for reliable data exchange in enterprise and government environments. This whitepaper discusses important technical aspects of PDF/A-1, PDF/A-2 and PDF/A-3.

PDF/A-1

PDF/A-1, the first standard within a series of multiple parts, has been published in 2005 as ISO 19005-1. It is based on PDF 1.4 (the file format of Acrobat 5) and imposes some restrictions regarding the use of color, fonts, annotations, and other elements. There are two flavors of PDF/A-1:

- Level B conformance (PDF/A-1b; »b« as in »basic«) ensures that the visual appearance of a document is preservable in the long term. PDF/A-1b ensures that the document will look the same when it is viewed or printed some time in the future.
- Level A conformance (PDF/A-1a; »a« as in »accessible«) is based on level B, but adds crucial properties of Tagged PDF: it requires structure information and reliable Unicode text semantics in order to preserve the document's logical structure and natural reading order. Simply put, PDF/A-1a not only ensures that the document will look the same when it is used in the future, but also that its contents (semantics) can be reliably interpreted and will be accessible to physically impaired users. As an important example, screenreader programs can read Tagged PDF documents to blind users.

PDF/A-2

The PDF world advanced a lot since the publication of PDF/A-1. Among many other milestones, PDF 1.7 (the file format of Acrobat 8) has been standardized as ISO 32000-1 in 2008. In order to make numerous new PDF features available in PDF/A workflows, a new part of the standard called PDF/A-2 has been published in 2011 as ISO 19005-2.

PDF/A-2 is based on PDF 1.7 and includes many useful additions which are not available in PDF/A-1. These include important file format aspects such as JPEG 2000 compression, optional content (layers), PDF packages and others. PDF/A-2 documents may contain file attachments provided the attached documents themselves conform to PDF/A-1 or PDF/A-2.

Similar to PDF/A-1, PDF/A-2 offers level B and level A conformance. It adds a new flavor called level U conformance. Level U sits in between PDF/A-2a and PDF/A-2b in that it requires reliable Unicode semantics, but not structure information. PDF/A-2u guarantees that the pages can faithfully be reproduced visually and that the text can be extracted and searched.

PDF/A-2 does not make PDF/A-1 obsolete or force users to migrate to the newer version – after all, this would be absurd for a standard which is targeted at long-time preservation!

PDF/A-3

Another part of the standard called PDF/A-3 has been published in 2012 as ISO 19005-3. PDF/A-3 also supports conformance levels A, B, and U. It differs from PDF/A-2 in the following aspects:

- While PDF/A-2 allows only file attachments which conform to PDF/A, PDF/A-3 allows arbitrary file types as attachments to meet the requirements of various user groups.
- File attachments are associated with the whole document, a page, or some other part of the document. The relationship between the attached file and the corresponding part of the document must be specified explicitly, e.g. source, alternative, or supplemental data.

Typical PDF/A-3 scenarios include embedding of word processor or spreadsheet source files in a final-form PDF/A document, or the inclusion of machine-readable XML data in a PDF intended for human consumption, e.g. an invoice.

PDF/A viewers are not required to do anything specific with attached non-PDF/A files except for extracting them. The PDF/A standard does not guarantee that attachments can be viewed or otherwise used in the future – it simply allows their presence in an archivable document.

In the same spirit as PDF/A-2 which does not replace PDF/A-2, PDF/A-3 does not replace PDF/A-2. Any part of the PDF/A standard can be used for long term archival as appropriate.

Technical Concepts in PDF/A

Fundamental PDF/A requirements

PDF/A requires certain PDF features and prohibits others:

- To guarantee the exact visual reproduction of text all fonts used in a document must be embedded.
- To guarantee exact color reproduction all colors must be specified in a device-independent way.
- METADATA must be embedded using the XMP format. The PDF/A conformance level must be recorded with specific XMP properties.
- Encryption must not be used to make sure that the documents contents can always be accessed without any restriction.
- Certain requirements for annotations and form fields ensure that the visualization is fixed and that screen and print representation are identical.

In addition to these straight-forward requirements, however, PDF/A requires various other PDF features which are more subtle (e.g. certain entries in font data structures), and prohibits some critical structures (e.g. certain combinations of TrueType fonts and encodings). There are many aspects which must be implemented and checked by software developers before they arrive at fully standard-conforming PDF/A products. PDF/A is much more than simply »PDF with embedded fonts and no encryption«!

Additional restrictions in PDF/A-1

PDF/A-1 somehow suffers from the fact that it was the first in the PDF/A family: the standard was created at a time when important PDF concepts were not yet ready for prime time. As a result, the following features are prohibited in PDF/A-1, but are allowed in the newer parts PDF/A-2 and PDF/A-3:

- All features which require PDF 1.5 or above, e.g. JPEG 2000 compression and layers (optional content).
- Transparency: although transparency is possible in PDF 1.4, it was not considered suitable for archiving purposes at the time because there was no consistent description and implementation of transparency support available. Since identical behavior in all PDF viewers could not be guaranteed, it was decided to completely ban transparency from PDF/A-1. After the publication of PDF/A-1 the exact semantics of PDF transparency have been clarified and standardized in ISO 32000-1; later standards therefore very well allow the use of transparency.
- File attachments were banned from PDF/A-1 to make sure that all document contents are fully archivable. While PDF/A-2 allows file attachments, it restricts them to PDF/A-1 or PDF/A-2 files to make sure that attached files can also faithfully be reproduced. PDF/A-3 further relaxes this rule to allow arbitrary file types as attachments.

Device-independent color specification

In order to ensure consistent color reproduction across output devices and time, PDF/A requires the use of device-independent color, usually achieved via ICC profiles or CIE Lab color specifications. The optional output intent describes the color characteristics of the document. While these concepts are widely used in the graphic arts industry, enterprise PDF developers are not necessarily familiar with color management and must familiarize themselves with ICC profiles and related concepts.

Raster images, e.g. TIFF and JPEG, play a vital role in document creation. Scanned paper documents and photographs from digital cameras are common examples of raster image data in document workflows.

In many cases raster image data in modern workflows is already device-independent, usually by means of an embedded ICC color profile or standardized color spaces such as sRGB. Such images are ready for use in PDF/A. However, legacy image data is in many cases device-dependent, such as black-and-white or RGB scans without any associated ICC profile.

XMP METADATA and extension schemas

Extensible METADATA *Platform* (XMP) is an XML-based format modeled after W3C's RDF *(Resource Description Framework)* which forms the foundation of the semantic Web initiative. In 2012 XMP has been standardized as ISO 16684-1. PDF/A mandates the use of XMP METADATA for storing information about a document inside the PDF itself. XMP provides a powerful and flexible framework for storing standard and custom METADATA properties (see our separate Whitepaper on XMP).

The XMP specification includes more than a dozen predefined schemas with hundreds of properties for common document and image characteristics. The most widely used predefined XMP schema is called the Dublin Core. It includes properties such as Title, Creator, Subject, and Description.

XMP is extensible by its very nature, i.e. company- or industry-specific METADATA requirements can be met by constructing custom schemas. PDF/A supports this concept. However, in order to ensure automated retrieval PDF/A mandates that a machine-readable description of the custom METADATA must be embedded in the document. This is achieved with an »XMP extension schema description«: a standardized part of the XMP METADATA describes the structure of custom XMP METADATA properties.

Level A conformance: Tagged PDF

PDF/A-1a, PDF/A-2a and PDF/A-3a require the use of Tagged PDF. While plain PDF only places visible content on a page, Tagged PDF requires that the document's logical structure is recorded within the structure hierarchy. Tagged PDF offers predefined structure element types for common parts of a document such as headings, tables, and lists. So-called marked content items can be considered the equivalent of tagged content in markup languages. They refer to elements in this structure tree. Similar to HTML and XML, Tagged PDF supports attributes for structure elements. For example, table elements can carry attributes regarding the row or column spanning properties of table cells.

Level A conformance also requires that all text in the document has Unicode semantics available (see below) and that logical words are separated by space characters.

PDF/UA-1 (Universal Accessibility) is a new standard which clarifies many aspects of Tagged PDF. It has been published in 2012 as ISO 14289. Although there is no direct relationship between both standards, a PDF/A document can at the same time conform to PDF/UA. In fact, if you want to create PDF/A with conformance level A we recommend to adhere to the PDF/UA requirements in order to improve accessibility. For more information please refer to the PDF/UA Whitepaper on the PDFlib Web site.

The following caveat applies to combined PDF/A and PDF/UA documents: since the *Scope* table attribute is required in PDF/UA, but not available in PDF 1.4 and therefore PDF/A-1, proper tagging of tables with header cells in not possible. We recommend to avoid PDF/A-1a if tables are involved, and work with the newer PDF/A-2a or PDF/A-3a standards instead.

Level U conformance: Unicode requirements

PDF/A-2 and PDF/A-3 offer level U conformance in addition to levels A and B. Level U requires proper Unicode semantics for all text in the document, but does not mandate Tagged PDF. This requirement is rooted in the fact that PDF supports a variety of font and encoding techniques, not all of which support Unicode. For example, PDF supports PostScript Type 1 fonts which have been introduced in the 1980's, while the Unicode consortium started its work in 1991. PDF/A conformance levels A and U require that supplementary Unicode mapping information must be present for fonts which do not contain it internally. But not all Unicode values are acceptable: values in the Private Use Area (PUA) are not allowed since they do not carry any common interpretation (semantics).

Symbolic fonts are an important area where this PDF/A requirement holds, e.g. fonts containing logos or pictograms. Since standardized Unicode values are not available for custom symbolic glyphs, suitable Unicode semantics must be provided in an »ActualText« marked content attribute for the text. While this attribute is commonly used only in Tagged PDF, it can also be supplied in untagged documents – and this is what level U conformance requires. The ActualText may be assigned to an individual glyph or a sequence of multiple glyphs. It may consist of an arbitrary Unicode string.

As an example, code 0x1A in the common WingDings font contains an image of a computer keyboard with the glyph name *keyboard* and the PUA Unicode value U+F037. For lack of better substitute text the glyph name could be used to construct suitable ActualText, e.g. »symbol for keyboard«. It should be noted that programmatically constructing ActualText must be considered a makeshift solution; human-selected text is always preferable to machine-generated ActualText.

Conforming PDF/A Viewers

While all conforming PDF/A documents are PDF documents, not all PDF viewers are necessarily conforming PDF/A viewers. This is caused by additional requirements imposed on PDF viewers by the PDF/A standard. The concept of a »PDF reader« as defined in the standard includes tools for viewing the contents of a document interactively, but also encompasses non-interactive tools such as a Raster Image Processor (RIP). While the basic PDF process of rendering a document to screen or paper is specified in ISO 32000-1, PDF/A further qualifies some aspects of rendering. Some examples:

- While plain PDF viewers are free to ignore ICC-based color specifications and may use the alternate color space instead, conforming PDF/A readers must always use the device-independent ICC-based color information.
- Conforming PDF/A readers must ignore certain device-specific information in a document, e.g. black generation and undercolor removal (these are device-specific features for the graphic arts industry).
- Conforming PDF/A readers are not allowed to render documents with fonts which may happen to be available locally on the viewing system. Instead, only the fonts embedded in the document are allowed for rendering.
- Starting with PDF/A-2, conforming viewers must ignore old-style document information fields and must fully rely on XMP METADATA.

Processing PDF/A Documents

Special care must be taken when processing PDF/A documents in order to maintain standard conformance. Even simple operations may spoil a document's conformance status. It is therefore crucial to deploy only tools which are PDF/A-aware to guard against the risk that PDF/A documents are modified in a way which violates the standard.

Splitting and Merging

Even simple operations may result in non-conforming documents. For example, inserting a page in a PDF/A document poses some immediate dangers:

- If the inserted page stems from a non-PDF/A document, it may use unembedded fonts.
- Even if the imported page stems from a PDF/A document dangers lurk in multiple areas. For example, the color characteristics (e.g. output intent) of both documents don't necessarily match, which again could result in non-conforming output.
- A small operation such as adding a METADATA field may violate the standard unless the software properly implements the rules for XMP METADATA as mandated by PDF/A.

Any kind of content or METADATA processing applied to PDF/A documents must be applied with PDF/A-aware software to avoid jeopardizing PDF/A conformance.

Digital Signatures

Digital signatures in PDF documents can be used to check the document's integrity, authenticate the person who created the signature, and determine the date and time of signature. Integrated digital signatures are part of PDF 1.4 and are allowed in PDF/A. Multiple document signatures using PDF's incremental update feature are also allowed. However, the signatures must meet certain requirements for PDF/A:

- If the signature has a visual appearance (e.g. an image or a textual representation of the signer's name) this appearance must meet the same PDF/A requirements as other document parts (device-independent color, fonts embedded, etc.).
- PDF/A-2 additionally contains requirements regarding the technical details of the signature. The standard also recommends to include timestamping and certificate revocation information in the signature.
- The signing procedure should be recorded in the document's XMP METADATA.

In order to make use of digital signatures in PDF/A workflows the signature tool must be aware of PDF/A, i.e. observe the rules outlined above.

The bottom line is that only PDF/A-aware tools must be used in PDF/A workflows; otherwise PDF/A conformance may be spoiled. In order to avoid PDF/A violations through accidental modification Adobe Acrobat opens PDF/A documents in read-only mode by default.

PDF/A Support in the PDFlib Product Family

PDFlib GmbH introduced PDF/A functionality in its products in 2006. PDFlib products were the first with support for XMP extension schemas. All products in the latest version 9 of the PDFlib product family support all flavors of PDF/A-1, PDF/A-2 and PDF/A-3. It provides application developers with a toolkit which allows the following PDF/A-related operations:

- create PDF/A from scratch, e.g. based on text from a database
- convert raster images (e.g. scans) to PDF/A
- process existing PDF/A documents, e.g. merge or split
- work with ICC profiles and device-independent color to deal with all color management issues
- create PDF/A level A with structure information (Tagged PDF), also in combination with PDF/UA
- attach XMP METADATA to the generated documents, including XMP extension schemas
- determine the glyph names in fonts to determine suitable ActualText for symbolic fonts
- attach PDF/A documents to PDF/A-2 or arbitrary file types to PDF/A-3

All of these operations can be implemented with simple PDFlib function calls. Sample code for a variety of programming languages and development environments is provided with the PDFlib distribution. Additional programming techniques for PDF/A are available in the PDFlib Cookbook. To facilitate font embedding as required by PDF/A, the Japanese Resource Kit for PDFlib includes common Japanese fonts. These fonts come with an embeddable license which is included in the software license.

Creating PDF/A with PDFlib

Creating PDF/A-conforming output with PDFlib is achieved by the following means:

- PDFlib automatically takes care of several formal settings for PDF/A, such as PDF version number and required XMP identification entries.
- The PDFlib client program must explicitly use certain function calls and options (e.g. for font embedding).
- The PDFlib client program must refrain from using certain other function calls and option settings (e.g. encryption).

If the PDFlib client program obeys to these rules, valid PDF/A output is guaranteed. If PDFlib detects a violation of the PDF/A creation rules it throws an exception which must be handled by the application. No PDF output is created in case of an error; there is no risk of creating non-conforming output if an error occurs. Details of required and prohibited operations are discussed in the PDFlib documentation.

Processing PDF/A with PDFlib

Additional rules apply when importing pages from existing PDF/A-conforming documents. When dealing with existing PDF/A documents, PDFlib+PDI carefully examines the PDF/A properties of all input and output documents to make sure that the output still conforms to PDF/A. For additional control the output intent of an imported document can be copied to the output PDF, effectively cloning the PDF/A color properties of an existing document. Similarly, XMP METADATA from imported documents can be cloned or merged.

Creating PDF/A level A with PDFlib

PDF/A conformance level A can be regarded as level B plus Tagged PDF. PDFlib's support for PDF/A level A is based on the features for producing Tagged PDF: each content item can be placed at a particular location in the document's structure tree; content items which are not relevant for the document structure (e.g. headers and footers, pagination) can be tagged as artifacts which means that they will be ignored when the document is read aloud by software or converted to some other format. Alternative text can be attached to images and vector graphics. PDFlib automatically tags tables and artifacts which is a big time-saver for the developer. PDFlib checks the supplied tags to make sure that the structure element nesting and attributes conform to ISO 32000-1. For example, heading or list tags must be properly nested.

Integrated support for PDF/UA makes it easy to create PDF output which is both accessible and archivable. Note that you need detailed knowledge about the document's logical structure in order to create Tagged PDF. PDFlib takes care of the PDF-related details, but it cannot infer the document structure from its contents.

PDFlib GmbH
Franziska-Bilek-Weg 9
80339 München, Germany
phone +49 • 89 • 452 33 84-0
support@pdflib.com
www.pdflib.com/knowledge-base/pdfa

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.

Founded in 2006 as the PDF/A Competence Center, the PDF Association exists to promote the adoption and implementation of International Standards for PDF technology.

- Developers use the PDF Association to share knowledge and experience with PDF technology.
- Decision-makers use the PDF Association to learn about the role and capabilities of PDF and PDF's subset standards in ECM and other electronic document applications.
- End-users benefit from improved reliability, quality and functionality and interoperability in their experience of electronic documents.