A Technical Introduction to PDF/A-1/2/3/4

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 series defines 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 the major technical aspects of PDF/A-1, PDF/A-2, PDF/A-3 and PDF/A-4.

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 restrictions regarding the use of color, fonts, annotations and other elements. There are two flavors of PDF/A-1 (called conformance levels):

- 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 in the near or far 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 can be interpreted reliably 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

PDF 1.7, the file format of Acrobat 8, has been standardized as ISO 32000-1 in 2008. In order to make new PDF features available in PDF/A, 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 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 another 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 visual appearance of pages can be reproduced faithfully 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 part of the standard – after all, this would be absurd for a standard which is targeted at long-term 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 is quite similar to PDF/A-2 and 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 use cases.
- File attachments are associated with the whole document, a page, or some other part of the document. The kind of relationship between an attached file and the corresponding part of the document must be specified explicitly, e.g. source, alternative, or supplemental data. For each file attachment its relationship to some part of the document must be specified with the *AFRelationship* key.

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. In fact, the ZUGFeRD and Factur-X invoice standards are an important application of PDF/A-3.

PDF/A-4

PDF/A-4 has been published in 2020 as ISO 19005-4. Since it is based on PDF 2.0 (published as ISO 32000-2 in 2017 and updated in 2020) it can take advantage of new PDF features. While PDF/A-2 and PDF/A-3 each comprise three different conformance levels which tended to confuse users, PDF/A-4 simplifies things since PDF/A-4 documents may or may not contain tags. Unlike previous parts of the standard no dedicated conformance level is required for tagged PDF/A-4 documents, thus eliminating the previous A/B/U conformance levels. Similarly, PDF/A-4 documents may or may not contain file attachments. The attached files must conform to PDF/A-1, PDF/A-2 or PDF/A-4.

While abandoning the A/B/U conformance levels, PDF/A-4 introduces two new conformance levels:

- PDF/A-4f allows non-PDF/A file attachments similar to how PDF/A-3 extends PDF/A-2.
- PDF/A-4e is targeted at the engineering community. It is slated as successor of the PDF/E-1 standard ISO 24517-1 which is based on PDF 1.6. The initial plan to define a new flavor PDF/E-2 has been cancelled. Instead, PDF/A-4e adds RichMedia annotations for 3D content in U3D or PRC format to the base PDF/A-4 format.

Regarding structure information and accessibility PDF/A-1a/2a/3a require only the mere presence of tags, but don't go into detail regarding the nature and use of PDF tags. PDF/A-4 goes one step backwards and one step forthwards at the same time: while PDF/A is agnostic regarding the presence of tags, it points out the advantages of Tagged PDF regarding content repurposing and accessibility. Regarding the specifics the standard references the PDF/UA standard (ISO 14289) which discusses many details of Tagging. Also, PDF/A-4 inherits the rigid regime of PDF tags which is part of the underlying PDF 2.0 specification.

Which part to use?

In the same sense as PDF/A-2 does not replace PDF/A-1, PDF/A-3 does not replace PDF/A-2 and PDF/A-4 does not replace PDF/A-3. Any part of the PDF/A standard can be used for long term archival. You simply have to relinquish certain PDF features as long as you work with an older part of the PDF/A standard. For example, simple office documents without transparent graphics can still be implemented with PDF/A-1. If you need arbitrary file attachments use PDF/A-3 or PDF/A-4f. If you need RichMedia/3D contents use PDF/A-4e.

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. The only exception are fonts used for invisible text; these don't have to 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. While PDF/A-1/2/3 impose strict requirements on custom METADATA properties, this has been relaxed in PDF/A-4.
- Encryption is not allowed to make sure that that the document 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 without guaranteed rendering results. 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«.

Specific restrictions in PDF/A-1

PDF/A-1 reflects 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:

- 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 of transparency support available. Since identical behavior in all PDF viewers could not be guaranteed transparency was completely banned 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 allow the use of transparency.
- File attachments were banned from PDF/A-1 to make sure that all document contents are fully archivable.

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 color profiles or CIE Lab color specifications. The optional output intent describes the color characteristics of the document with an ICC profile. 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. Often raster image data 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 an associated ICC profile.

XMP METADATA and extension schemas in PDF/A-1/2/3

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

METADATA in PDF/A-4

The convoluted concept of XMP extension schemas introduced with PDF/A-1 didn't really catch on with developers and users. The industry had to struggle for several years to work out those details about extension schema processing which were missing from the standard text. This led to frustration, since on the one hand it was hard to correctly add custom METADATA properties to PDF/A, and on the other hand applications which didn't use custom properties nevertheless triggered XMP-related errors in PDF/A validators. PDF/A-4 eliminates these problems in a radical way by completely getting rid of XMP extension schema descriptions. They are replaced with a machine-readable schema description according to the Relax NG standard, published in 2014 as ISO 16684-2. However, unlike the required extension schemas in PDF/A-1/2/3, schema descriptions are optional in PDF/A-4.

Another source of problems was the requirement to synchronize XMP METADATA with entries in the document information dictionary. This so-called crosswalk was underspecified and even got some

details wrong in the first published version of PDF/A-1. Since PDF 2.0, the basis of PDF/A-4, almost completely deprecates document info entries, PDF/A-4 no longer requires METADATA synchronization.

PDF/A-1/2/3 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 contents 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) 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-1/2/3 with conformance level A we recommend to adhere to the PDF/UA requirements in order to improve accessibility. For more information refer to the PDFlib Whitepaper on PDF/UA.

PDF/A-4 abandons level A conformance and simply mentions the advantages of Tagged PDF for content recovery. The standard references PDF/UA for further guidance, i.e. the recommendation above is now included in the standard.

PDF/A-2/3 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, a format which is deprecated or no longer supported in many current operating systems and applications. This format has 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 don't carry any common interpretation.

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 attribute can be assigned to an individual glyph or a sequence of multiple glyphs.

PDF/A-4 eliminates level U conformance, but recommends level U Unicode properties for all documents. However, this is not a strict requirement.

Annotations and PDF/A-4 Level E conformance

PDF supports a variety of annotation types (also called comments) which enrich documents. Some annotation types are prohibited in PDF/A; allowed annotations must adhere to several rules.

In PDF/A-1 *Sound* and *Movie* annotations are not permitted since »support for multimedia content is outside the scope« of the standard. In the same spirit PDF/A-2 and PDF/A-3 disallow the newer *3D* and *Screen* annotation types. PDF/A-4 prohibits *Sound*, *Screen* and *Movie* annotations.

In addition, PDF/A-4 introduces conformance level E. It can be considered the successor of the PDF/E standard for PDF in engineering which didn't find widespread adoption. PDF/A-4e allows *3D* and *RichMedia* annotations in support of interactive applications. Regarding 3D data the standard recommends *RichMedia* annotations instead of *3D* annotations.

Another new condition in PDF/A-4 which stems from PDF 2.0 is the requirement to have annotation appearances included in the document. These describe the graphical representation of an appearance. While the appearance dictionary contains a description of its visual representation (such as border style, color, font etc.) the task of creating the visual representation from the description is up to the PDF viewer and not standardized. In order to ensure reliable rendering of annotations the PDF creation software must include the visual representation of the appearance of all annotation types except Popup and Link.

File Attachments and PDF/A-4 Level F conformance

Attachments can be embedded in a PDF document on the document level or on a page with the help of *FileAttachment* annotations. Rules for embedded files differ substantially among PDF/A parts:

- PDF/A-1 completely prohibits attachments.

- PDF/A-2 allows attachments, but the embedded documents must conform to PDF/A-1 or PDF/A-2.
- PDF/A-3 allows attachments with arbitrary content types.
- PDF/A-4 allows attachments which conform to PDF/A-1, PDF/A-2 or PDF/A-4. It also introduces a dedicated conformance level F which allows arbitrary content types.

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 uses PDF/A as a carrier document.

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. 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 and PDF/A-3 contain additional requirements regarding technical details of the signature. The standard also recommends to include timestamps and certificate revocation information in the signature.
- PDF/A-4 allows one certification signature, one or more approval signatures and one or more time-stamp signatures. All signatures must conform to an appropriate PAdES profile.

Conforming PDF/A Viewers

While 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 basic rendering of a document on screen or paper is specified in ISO 32000, PDF/A further qualifies several aspects of rendering including the following:

- 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 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.

PDF/A Validation

PDF/A validation is the process of checking whether a document conforms to the requirements of a particular part of the PDF/A standard. Validation has been available for a long time as part of Acrobat's Preflight component as well as from several independent software vendors. In order to provide a useful resource for the community the Open Preservation Foundation (OPF), the PDF Association and the Digital Preservation Coalition (DPC) collaborated in the development of a freely available and reliable PDF/A validator called veraPDF. Its development has been funded by the European Commission's Preforma project and is supported by the PDF software developer community as organized in the PDF Association.

If you are in doubt regarding the standard conformance of a particular PDF/A document we recommend to check the issue with veraPDF.

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. 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 several 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 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-1/2/3.

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

In order to make use of digital signatures in PDF/A workflows the signature software 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. Once the available editing and modification tools in Acrobat are used, PDF/A conformance is no longer guaranteed.

Document assembly and Tagged PDF

Assembling documents from Tagged PDF pages is particularly tricky. On the technical level the structure hierarchies of the involved PDF documents must be combined which involves convoluted operations with the Tagging data structures. Even more difficult are semantic challenges. For example, the document assembly process must take into account the logical entities which are combined. For example, a structure element such as a paragraph or table may span multiple pages. If these pages are separated or combined in different order the structure hierarchy is easily spoiled.

Document assembly with Tagged PDF requires careful planning of all involved semantic entities. For example, the task can be simplified if the workflow ensures that major semantic units like document sections start on a new page.

PDF/A Support in PDFlib GmbH Products

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 PDFlib product family support all flavors of PDF/A-1, PDF/A-2 and PDF/A-3 (PDF/A-4 support in development). 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
- assemble Tagged PDF/A from existing tagged pages
- attach XMP METADATA to the generated documents, including XMP extension schemas
- 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 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.

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 application program must explicitly use certain function calls and options (e.g. for font embedding).

- The PDFlib application program must refrain from using certain other function calls and option settings (e.g. encryption).

If the PDFlib application 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 exception; there is no risk of creating non-conforming output. Details of required and prohibited operations are discussed in the PDFlib documentation.

Processing PDF/A with PDFlib+PDI

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. 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.

PDF/A-conforming signatures with PLOP DS

PDFlib PLOP DS is a toolkit for applying digital signatures to PDF documents according to the PAdES signature standards required for signatures according to European eIDAS regulations. PLOP DS applies signatures to PDF/A documents such that the signed output also conforms to PDF/A.

PDFlib GmbH
Franziska-Bilek-Weg 9
80339 München, Germany
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 PDF/A Competence Center, in 2011 the PDF association broadened its scope to cover all aspects of PDF technology. Today, it provides an industry meeting-place, and a platform for members to exercise thought-leadership in the community.

- 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.