

ISO 19005-1 (PDF/A-1) Application Notes

Description

This document serves as the official reference for the Application Notes for ISO-19005-1 (PDF/A-1). It is organized by section, consistent with ISO 19005-1 itself. In addition, there are sections about features of PDF that are not referenced in the specification that are of importance to application developers when working with PDF/A-1.

Unless otherwise noted, the material in these notes applies to all conformance levels addressed by these notes, which are simply referred to as PDF/A-1, unless specifically notes for a particular conformance level.

If there is a conflict between these Application Notes and any part of ISO 19005:2005, the standard will always take precedence.

Table of Contents

Introductory Material

-

Introduction

- Design goals for PDF/A
- Relationship between PDF/A and other standards
- References to PDF/A

General

- Identification and conformance
- Selecting the appropriate PDF/A conformance level
- File translation
- Validation of PDF/A workflows
- Compression of PDF objects
- Compression of entire PDF/A files
- Security and encryption of entire PDF/A files

Specification Notes

- 6.1 File Structure
- 6.2 Graphics
- 6.3 Fonts
- 6.4 Transparency
- 6.5 Annotations
- 6.6 Actions
- 6.7 Metadata
- 6.8 Logical Structure
- 6.9 Interactive Forms

Additional Notes

- Digital Signatures
- Scanning and OCR
- Development of receiver requirements
- Document identification and metadata
- File types and extensions
- Assembling PDF/A files into a single output
- PDF/A Viewer Requirements
- Hidden & Invisible Content

Introduction

The PDF/A family of standards was, and is continuing to be, developed by ISO/TC 171/SC 2/WG 5. The goal is to provide a reliable storage format for PDF data across multiple generations of technology. It is a multi-part standard with several levels of compliance. Businesses, governments, libraries, archives and other institutions and individuals around the world use PDF to represent considerable bodies of important information. Much of this information must be kept for substantial lengths of time; some must be kept permanently. The future use of, and access to, these objects depend upon maintaining their visual appearance as well as their higher-order properties.

It is important to note that while these standards ensure the reliable storage and visual reproduction of the data, they make no statement about the quality of the data contained in the files. Because such issues as raster data resolution, printing conditions, annotation requirements (crop marks, register marks, etc.), pages per file, allowed font types, presence of OCR/hidden text, etc. are largely market specific they are not specified in these standards. Rather, they are felt to be the province of trade associations and/or organizations who have already begun to publish such specifications for these parameters in support of the use of the PDF/A-1 standard.

In many situations specifying and/or validating a file as having a specific level of PDF/A-1 compliance will be a key element in assuring an efficient workflow involving the long term preservation of digital documents.

The family of PDF/A standards currently consists of one standard with two conformance levels:

- ISO 19005-1A:2005, Document Management – Electronic document file format for long-term preservation – Part 1: Use of PDF 1.4 (PDF/A-1); specifies the use of the Portable Document Format (PDF) for the not only the visual representation but also semantic information and comprehensive metadata in compliance with all parts of the specification.
- ISO 19005-1B:2005, Document Management – Electronic document file format for long-term preservation – Part 1: Use of PDF 1.4 (PDF/A-1); specifies the use of the Portable Document Format (PDF) for primarily visual representation and does not require the presence of semantic information or inclusion of all specified types of metadata.

A file conforming to ISO 19005-1 must contain all the content information necessary to process and render the document, as intended by the author. It is platform and transport independent.

These application notes currently only include information applicable to:

- ISO 19005-1:2005

All application developers and users constructing workflows are strongly encouraged to use the latest version of the relevant PDF/A standards wherever possible.

It is anticipated that a variety of products will be developed around the PDF/A family of standards, such as readers (including viewers) and writers of PDF/A files, and products that offer combinations of these features. Different products will incorporate various capabilities to prepare, interpret and process conforming files based on the application needs as perceived by the suppliers of the products. However, it is important to note that a conforming reader must be able to read and appropriately process all files conforming to a specified conformance level. It is also important to note that a reader of other PDF documents, does not necessary qualify it as a conforming PDF/A reader.

This document is not a part of the ISO 19005 family but is intended to supplement these standards by providing additional information and guidance. None of the information contained herein should be construed as modifying the standard. This document is intended for implementers of writers and readers of PDF/A files. It also provides information that will help workflow designers use PDF/A to advantage.

This document was developed as a cooperative effort between the technical experts of ISO/TC 171/SC 2/WG 5. To anyone who acknowledges that these Application Notes are provided "AS IS", WITH NO EXPRESS OR IMPLIED WARRANTY: permission to use, copy and distribute them for any purpose is hereby granted without fee, provided that the contents of these notes are not altered, including the NPES copyright notice tag. Neither NPES nor AIIM makes any claims or representations about the completeness of these Application Notes.

These Application Notes are subject to revision and enhancement. The most current version of this document can be found in the NPES standards workroom at <http://www.npes.org/standards/tools.html>. Some earlier revisions may be maintained at the same site.

Comments and suggestions should be sent to the AIIM Secretariat at standards@npes.org.

Design goals for PDF/A

The primary purpose of this part of ISO 19005 is to define a file format based on PDF, known as PDF/A-1, which provides a mechanism for representing electronic documents in a manner that preserves their visual appearance over time, independent of the tools and systems used for creating, storing or rendering the files.

PDF/A-1 is a subset of the full PDF 1.4 Reference (3rd ed.; Boston: Addison-Wesley, 2001) that restricts the use of any visual or interactive element that would change the rendering of the document from the way in which the author created it. It also makes normative a variety of recommendations in the PDF Reference to avoid any ambiguity for future implementers.

Relationship between PDF/A and other standards

PDF/X

PDF/A-1 adopted the same imaging and color requirements specified in PDF/X-3 (ISO 15930-3:2003). However, three major areas of difference exist that developers should be aware of.

Strict adherence to details of PDF File Structure

Because PDF/X concerns itself primarily with the production of PDF for short-term formalization for print production, it assumes that whatever tool(s) were used to create the PDF (eg. Adobe Acrobat) will also be available for the printing of the document. PDF/A-1, however, is unable to make the same assumptions due to the long-term preservation nature of the format. As such, a lot of work was put into the specification (Sections 6.1.1 to 6.1.9) to insure that PDF/A-1 documents were more consistent and reliable and that implementers of PDF/A-1 in the future would have an easier time.

Presence of non-content elements in the visible area

While PDF/X places restrictions on annotations (and related non-content elements), such that they may not be present inside of the TrimBox (thus keeping them outside of the visible/printable area), PDF/A-1 has no such restrictions. It considers annotations (including hyperlinks, form fields, and digital signature blocks) to be part of the visible document. As such, PDF/X consumers will need to be updated to respect this difference.

Use of XMP Metadata

PDF 1.4 was chosen as the basis for PDF/A-1 for a number of reasons including the fact that it introduced a new XML-based metadata format called XMP. As XML-based metadata schemas were already in use in many parts of the archival community, having a way to integrate those sources with PDF/A-1 was an important goal for the committee. In addition, the committee extended XMP to introduce a number of new extension schemas for specific purposes within the PDF/A-1 framework (Sections 6.7.8 to 6.7.11). PDF/X, however is not concerned with metadata and thus continues to use the old style "Info Dictionary" method.

However, because of the features that they do share, it is possible to create a single PDF that is compliant with both PDF/A-1 and PDF/X-3. Currently, both Acrobat 8 and PDF Appraiser support the production a document with this feature.

PDF/E

PDF/E builds on some of the ideas and implementations from ISO 19005-1, especially in the areas of fonts and metadata in order to ensure a reliable rendering for static information in the PDF. However, in order to address the needs of the engineering community for “live” documents with the latest features of PDF available, they diverged in some specific areas.

Use of PDF 1.6

PDF/E is based on the complete PDF 1.6 standard, as well as features and functionality from 1.4 and 1.5 that is not present in ISO 19005-1. Two specific areas of note are support for transparency and optional content groups – both items up for discussion in ISO 19005-2.

Interactive Media & 3D

As many diagrams in the engineering world require the inclusion of 3D data or other interactive media formats (QuickTime VR, Flash animations, etc.), such elements that are “out of scope” for ISO 19005-1 are allowed in PDF/E.

Security/Encryption

As PDF/E documents are more “here and now” than about the future, the application of encryption and digital rights to a document is appropriate – but such documents are quite incompatible with the philosophy of ISO 19005-1.

We do however, need to address the transition from an "active" PDF/E document and a "static" PDF/A document that is a part of the regulatory lifecycle of engineering documents.

References to PDF/A

Because this initial version of PDF/A-1 is based on PDF 1.4, the standard is being published in parts so that new parts can be added without obsolescing previous parts. For example, PDF/A-1 refers to the format defined by part 1 (ISO 19005-1) of the standard (or PDF 1.4) while part 2 (ISO 19005-2) and later parts may be based on a later version of PDF and/or may define archiving requirements for more complex content types.

References to the standard in product literature, packaging, etc., describing PDF/A compliant applications should include the ISO Standard Number and the Part Number ISO 19005-1.

Identification and conformance

A PDF file is identified by a header, the first line of which is a PDF comment that begins with “%PDF-” and is followed by a version number. For conformance with PDF/A-1:2005, the version number should be 1.4. However, since any file conforming to an earlier version of PDF also conforms to version 1.4, an application that processes PDF/A files must also accept files with any of the following headers:

```
%PDF-1.1  
%PDF-1.2  
%PDF-1.3
```

A version 1.0 PDF (%PDF-1.0) is not, however, conformant to all parts of Section 6 of the PDF/A specification.

A conforming PDF/A-1a file is identified by:

1. being a PDF file; and
2. having a Metadata key in the Catalog dictionary, whose value is a well-formed XMP packet
 - having an rdf:Description element containing a pdfaid:part attribute (or child element), whose value is “1” and a pdfaid:conformance attribute (or child element) whose value is “A”.

A conforming PDF/A-1b file is identified by:

3. being a PDF file; and
4. having a Metadata key in the Catalog dictionary, whose value is a well-formed XMP packet
 - having an rdf:Description element containing a pdfaid:part attribute (or child element), whose value is “1” and a pdfaid:conformance attribute (or child element) whose value is “B”.

An example might look like:

```
<rdf:Description pdfaid:amd="2005" pdfaid:conformance="B"  
pdfaid:part="1"  
xmlns:pdfaid="http://www.aiim.org/pdfa/ns/id/" />
```

The same information can also be represented using the expanded XMP format via child elements instead of attributes:

```
<rdf:Description rdf:about=""  
xmlns:pdfaid="http://www.aiim.org/pdfa/ns/id/">  
  <pdfaid:conformance>B</pdfaid:conformance>  
  <pdfaid:part>1</pdfaid:part>  
  <pdfaid:amd>2005</pdfaid:amd>  
</rdf:Description>
```

This piece of XMP is required to be present in PDF/A-1 document (Section 6.7.11); any file that does not contain this is not compliant with PDF/A-1.

In all cases “being a PDF file” means conforming to the PDF Reference: Adobe Portable Document Format, Version 1.4 , as extended by Errata for PDF Reference, third edition (as published in the first printing, November 2001; last modified 18 June 2003). Throughout the rest of this document that combination will be referred to as the PDF Reference.

A conforming PDF/A file has three kinds of content:

- content that affects the final visual reproduction of the composite entity;
- other visual content such as annotations, form fields, etc.
- non-printing content such as bookmarks, metadata, etc.

The PDF/A-1 standards state that a conforming file may include valid PDF features beyond those described in the standard provided they do not affect final visual reproduction of the composite entity and are included as part of PDF Reference 1.4. Implementers of PDF/A writers should be aware that features added in versions of PDF later than that defined in the PDF Reference 1.4 might affect the final visual reproduction. This includes issues such as JPEG2000 compression and 16 bit images from the PDF 1.5 specification, 3D annotations and page scaling from PDF 1.6 and more. These new features should not be used when creating PDF/A-1 files.

In addition, although newer features of PDF, such as object properties would not affect the visual rendering of the PDF, they should not be included in a PDF/A document as there may be other considerations for archiving that have not been fully considered.

It is the responsibility of the producer/writer of PDF/A files to ensure compliance with the appropriate PDF/A file format.

Selecting the appropriate PDF/A conformance level

The use of PDF/A-1, instead of standard PDF, is one step toward ensuring the reliability of the document for the future – but only as far as the visual representation. It does not, necessarily, provide enough information for the extraction of semantic data as would be necessary for high fidelity conversion to other formats. It also does not, necessarily, provide rich metadata that an archivist might need in their studies of the document's history. For users requiring such additional richness in reliability, they should use the PDF/A-1a level of conformance which requires the presence of such semantic and metadata information. PDF/A-1a authoring tools should not use automated or generic tagging methods, as they won't be fully representative of real semantics.

Most importantly, for long-term preservation of electronic documents, PDF/A-1 does not stand alone. Because PDF/A-1 does not ensure the exact replication of source data that implementers will need to implement quality assurance, records management and other controls to ensure the long-term quality and integrity of PDF/A-1 files as records. Refer to the Introduction of ISO 19005-1 and Appendix B for more information.

File translation

PDF/A is a long term file format. As such, without appropriate controls in place to ensure proper migration, it would be inappropriate and ill-advised for the receiver to translate PDF/A files into other file formats as there would be no guarantee that the intended rendering and/or the original semantics are preserved during the conversion process. This includes raster image formats (eg. TIFF) as well as other page description languages, such as Postscript.

However, as noted in Selecting Conformance, it might well be warranted in many workflows to extract content and/or metadata for use elsewhere **in addition to**, but not instead of, the PDF/A document.

Also, for the transmission of documents for short-term usage in a non-PDF/A workflow, conversion of PDF/A to a format that would be acceptable for that purpose may be warranted - but should be viewed as a last resort.

Validation of PDF/A workflows

Most of these application notes concentrate on the creation of conforming PDF/A-1 files. However, once a conforming file has been created and validated, there is a chance that the file may not be interpreted correctly in subsequent processes.

Some workflows may include a conversion from PDF to TIFF, especially if they are using older (non-PDF/A-aware) tools to aggregate or merge documents into sets or integrate into Document Management Systems. As noted in File Translation, such conversion will need to be done with appropriate controls in place. Also, we understand that creation of document thumbnails (as TIFF or JPEG) as a secondary reference for the document will be a common practise.

Viewing or Printing of PDF/A-1 documents should be done, if at all possible, with a viewer that is conformant to the PDF/A specification. Non-conformant viewers may change (without knowledge) the visual representation of the document, thus invalidating the use of PDF/A-1. In addition, since PDF/A-1 permits the inclusion of interactive elements (eg. annotations, form fields and hyperlinks) and describes how they should be handled by a conformant viewer. For example: external links can either be treated as active or inactive. Consideration of these needs should also be taken into account when choosing a conformant viewing tool.

Of special consideration in this area are:

- the handling of PDF forms (aka AcroForms) where a PDF/A conformant viewer should not enable editing or modification of fields.
- proper color management based on the OutputIntent, Default colorspaces and the other requirements of ISO 19001.

Compression of PDF objects

The PDF/A-1 standard does not specify how compression should be performed on the individual elements that are contained either directly in or as “embedded” parts of the PDF file. Any PDF stream may be compressed with any PDF 1.4-supported compression technique, other than LZW, that is appropriate to the data.

For contone images, a variety of compression choices are available including JPEG, which is a lossy compression technique. The use of lossy compression may result in degradation of image quality. In addition, JPEG2000 compression may not be used.

For monochrome images, the use of JBIG2 is fully supported by PDF/A and is recommended due to its excellent compression ratios. Users of JBIG2 should be aware that it supports both a lossless and a lossy mode, though unlike JPEG the data that is “lost” in JBIG2 is of the “image noise” variety and so may not be problematic in certain workflows.

LZW compression is prohibited in PDF/A-1. Note that some older versions of Adobe Acrobat Distiller use LZW compression in the creation of thumbnails. If creation of thumbnails from Acrobat Distiller is desired, then the LZW compressed thumbnails must be recompressed using a compression method approved for PDF/A-1. Other applications may produce PDF files that require similar changes when converting to PDF/A-1.

Although all readers are required to implement all specified compression schemes, writers are not required to use compression.

Compression of entire PDF/A files

External compression can be useful to assist in identification of file corruption during transmission, or can be used to provide a mechanism for password protection of sensitive files. If external compression is to be applied to a PDF/A-1 file, either for storage or transmission, a lossless compression method must be used. The use of lossless compression will ensure that the integrity of such a file is preserved, since lossy compression methods on PDF documents will damage the document, usually beyond repair.

However, placing the PDF/A-1 document into another container format (eg. Zip, Stuffit) – one that is not guaranteed to survive long term archival storage – should take place only as part of a short term process such as emailing, network exchange or even SneakerNet.

NOTE: Compressing a file that contains compressed elements may cause the file to increase in size.

Security and encryption of entire PDF/A files

None of the PDF/A standards covered by these application notes permit the use of PDF-based encryption.

If there is a need to apply encryption to protect confidential data during exchange, an external file encryption application must be used. The use of encryption in file exchange requires prior understanding between sender and receiver. Once an entire PDF/A-1 file has been encrypted, the resulting encrypted file is no longer a conforming PDF/A-1 file, and it should be assumed that a compliant PDF/A-1 reader cannot read an encrypted file without prior decryption.

Although PDF 1.5 supports a feature called *Certified Documents* that enables the application of digital rights to a PDF, without the use of encryption, it should not be used with PDF/A-1 due to it being in a newer version of the PDF Specification than that referenced by PDF/A-1. In addition, all aspects of its conformance with archival standards has not been fully evaluated.

File Structure

The purpose of this section of the PDF/A-1 standard is to ensure that there is little to no ambiguity in the actual file format - enabling easier development of PDF/A-1 consuming applications.

Although most of the sections in 6.1 simply make normative recommendations in the PDF Reference, there are a few sections that prohibit specific PDF features that are in opposition to the goals of PDF/A-1. The best examples of this are 6.1.3's prohibition of the /Encrypt key, which is necessary when a PDF file is encrypted and 6.1.7's prohibition of the keys that would be used with external streams.

It should also be noted that although Optional Content (aka PDF Layers) weren't introduced into PDF 1.5, section 6.1.13 prohibits their presence in a PDF/A-1 document since they could significantly effect reliable visual representation.

Graphics

This section of ISO 19005-1 addresses issues concerning the rendering of content that does not involve fonts or interactive elements. It is derived from the PDF/X-3 specification.

Output Intent

The PDF/A standards require that all data in a file be prepared for a single target printing/viewing condition, thus insuring consistently renderable color. This condition must be identified using an OutputIntent, as described in PDF Reference, which includes an ICC profile, through which all device-dependant colors will be mapped.

ISO 19005-1:2005 recommends that the value of the Info key in an output intent dictionary be used to carry a description of the characterized printing condition in a form that will be meaningful to a human operator at the site receiving the exchanged file. In addition, it is recommended that you also use the OutputCondition key for this information as well.

Note that the PDF Reference states that the Info key is required if “OutputConditionIdentifier does not specify a standard production condition.” While it is not clear exactly what is meant in that document by a standard production condition, the requirement that a PDF/A-1 file also be a valid PDF file means that it is safest to use the Info key as well as the OutputCondition key in output intent dictionaries.

PDF/A-1 validation tools or applications that wish to produce a single PDF that is compliant with both PDF/A-1 and PDF/X-3 should take special note of section 6.2.2 and the handling of multiple DestOutputProfile values.

Colour

As described in ISO 19005-1, all colors must be either Device Independent or must match the colour space of the Output Intent profile. In addition, Separation (Spot) and DeviceN colors may be used, provided they meet the requirements set forth in the spec.

If color management is to be applied to a PDF/A-1 file during rendering, care should be taken to apply the same processing to images and vector objects unless different rendering intents have been explicitly selected for them. This will ensure that vector objects intended to match parts of an image will do so.

When a color is specified as ICCBased, the embedded profile is required to be compliant with the PDF 1.4 specification (6.2.3.2). This would only allow for ICC version 3 profiles, and therefore producers and consumers should be sure to validate the version of the profile.

Images

In addition to the stipulations in the spec, writers of PDF/A-1 consuming applications should be aware that the rendering intent for an image can come not only from the Intent key in the Image XObject dictionary, but also from the current (extended) graphics state.

Although not explicitly prohibited by the PDF/A-1 specification (since it is part of PDF 1.5), it is recommended that PDF/A-1 producers not use 16-bit images and that any PDF/A-1 validator ensure that they are not present.

Embedded Postscript

Postscript can be embedded into a PDF file, in hopes that a consumer will include such code in the actual printing process. However, since most printers today are not Postscript-based, since no interactive Reader can consume Postscript and since the processing of embedded Postscript would therefore change the final rendering - the inclusion of Postscript in a PDF/A-1 file is forbidden.

There are three ways in which Postscript can be embedded - none of which may be present in a PDF/A-1 document.

- Form XObject with Subtype2 key and value of PS
- Form XObject with PS key
- ps operator in a content stream

Content Streams

Authors of PDF/A-1 creation and consuming tools should be aware that although not explicitly stated in section 6.2.10 of the PDF/A-1 spec, all content stream rules apply not only the content streams of a page, but to all five possible uses of content streams in a PDF:

- Content stream of a page
- stream of a Form XObject
- appearance stream of an annotation, including form fields
- content stream of a Type 3 font glyph
- stream of a tiling pattern

Fonts

Introduction

Historically, there have been a number of font-related problems in PDF files. Many of these problems have been addressed in the PDF/A-1 standard by placing restrictions on font usage and embedding.

The PDF/A-1 standard reduces the probability of font-related problems by requiring the creator to ensure that all fonts referenced in the file are embedded, and that only embedded fonts and metrics are used when reading, displaying, and printing a PDF/A-1 file.

A PDF/A-1 conforming writer must always embed fonts (either completely or as only a subset), and a conforming reader must always use the embedded fonts.

Problems not Addressed by PDF/A-1

There are a number of potential font problems that cannot be addressed by the PDF/A-1 standard, as they are rooted in the PDF/A creator's environment. These potential problems are as follows:

- Fonts may be substituted without warning. Unintended font substitution may occur if the fonts are not available to the creation application when it exports a PDF/A-1 file, generates a PostScript file, or converts a PostScript file to PDF/A-1.
- Some combinations of Microsoft Windows printer drivers and TrueType fonts may produce undesirable results if the file is to be used for purposes other than rendering for printed output. PostScript files generated from these drivers that contain TrueType fonts may lead to un-searchable and un-editable text in PDF files, depending on how that PostScript is converted to PDF.
- Microsoft Windows printer drivers, in combination with the selected PPD, allow the user to specify how TrueType fonts are to be embedded in the resulting PostScript file. Typically, TrueType fonts can be included as TrueType outlines as FontType 42, CIDFontType 2 (with FontType 42 wrappers), or they may be converted to FontType 1 outlines, or bitmaps in either FontType 3 or FontType 32 format. Regardless of the conversion performed, the fonts can be embedded in the PDF/A. The highest quality output will be obtained by retaining the fonts as TrueType (Type 42), preserving their hinting information and this approach is recommended. It is unlikely that text produced by bitmap fonts would print at high quality on a high-resolution device.
- When using a workflow that starts by generating a PostScript file and then converts it into PDF it is recommended that all fonts used in the document are embedded in the PostScript stream if possible, and that the conversion tool be set to use those embedded fonts. If the conversion tool embeds fonts in the PDF file that were not present in the PostScript there is a chance that the wrong font could be selected if more than one version of the font is available on the computer. This is especially important when using a server-based conversion tool where the document is printed on one machine and the PostScript converted to PDF on another. For the same reason, EPS graphics containing text should have all fonts used embedded in them where possible.

Embedding Issues

Creators are warned that not all fonts can be legally embedded and that legal restrictions on embedding should be determined and respected as part of the creation process. Creators need to be aware that there are two types of embedding restrictions – technical and legal.

- **Technical:** Fonts in TrueType or OpenType format contain a flag word (in the OS/2 table) that specifies whether embedding is allowed and if so what kind (installable, view & print, view, print & edit). PDF/A creators **must** respect this flag.
- **Legal:** Fonts, like software programs, will include an End User License Agreement (EULA) that stipulates the uses that the creator permits the licensee/user. These may include the ability to embed, and if so, for what purposes. There is no way for the creating software to know about this EULA, thus it is incumbent on the author of the document being converted to PDF/A to respect such licenses.

Vendors creating PDF/A-1 tools (preflight or validation utilities, PDF/A-1 creators, etc.) are encouraged to include checks for these issues where possible, because without it, the file may not be a valid PDF/A-1 file.

Metadata

A PDF/A-1a creator is recommended to include a /Metadata entry in the FontFile Stream Attributes Dictionary (Section 6.7.10). This block of XMP is should contain five entries in the xmpRights schema.

5. xmpRights:Title – the font’s name, which will be a copy of the string value from the /FontName key in the /FontDescriptor dictionary of the embed font.
6. xmpRights:Copyright – the copyright information for the embedded font. This information will be extracted from the ‘name’ table (ID #0) of a TrueType or OpenType font or from a Postscript comment in a Type 1 font.
7. xmpRights:Marked – always a Boolean “True”.
8. xmpRights:Owner - if the owner can not otherwise be determined, then the value of this attribute shall be the same as that as xmpRights:Copyright.
9. xmpRights:UsageTerms – the terms under which the font may be used. This information will be extracted from the ‘name’ table (ID #13 or 14) of a TrueType or OpenType font or from a Postscript comment in a Type 1 font

Documents that conform to PDF/A-1A are required to contain **/ToUnicode** entries in each Font Dictionary (Section 6.3.8), unless they meet the requirements of that section. As described in Adobe TechNote #5411 (ToUnicode Mapping File Tutorial), **/ToUnicode** entries represent a specially formatted stream of data that maps each glyph ID in the font to a specific Unicode codepoint.

OpenType Embedding

Although PDF 1.6 supports the complete embedding of OpenType (.otf) fonts, PDF 1.4 (on which PDF/A-1 is based) requires the OpenType fonts be “broken apart” into their TrueType

or Type 1 origins and that information placed into the PDF as if they were TrueType or Type 1 fonts originally.

Render Mode 3

Section 6.3.4 states “the font programs for all fonts used within a conforming file shall be embedded within that file”, thus basically saying that “all fonts have to be embedded”. However, it then goes on to say “except when fonts are used exclusively with text rendering mode 3”.

As stated in section 5.2.5 of the PDF Reference, text elements in a PDF can be displayed in one of seven different rendering modes. These modes define whether to stroke, fill, stroke and fill, or clip the text. One unique mode is #3 - that the text is neither stroked nor filled, thus making it “invisible” or “hidden”. Such text is used primarily (though not only) for placement of text that has been generated from a scanned images via Optical Character Recognition (OCR) technology.

This enables programs that create “Image + Hidden Text” PDFs to not enlarge the document via the embedding of the font(s) used for the hidden text. However, it makes checking/verification of PDF/A documents more difficult as the checker must now analyze the actual usage of each text operation to determine the mode in use.

Font Subsets

The ISO 19005-1 neither requires nor prohibits the use of font subsets in a PDF/A file. Embedding font subsets is recommended to avoid potential font name conflicts because font subsetting forces a renaming of the font within the PDF file.

When subsetting a font, section 6.3.5 requires that “all font glyphs referenced for rendering” be provided. This means that the embedded font data must include at least the same glyphs that are used, but may contain more. For example, if the only glyphs used in a font are for the characters ‘H’, ‘e’, ‘l’ and ‘o’ - then the font need only contain those 4 glyphs. However, the PDF/A-1 producer may choose to include additional characters beyond just those 4 and it would still be in compliance. PDF/A-1 validation tools should be aware of this distinction.

Font Metrics

Section 6.3.6 makes normative the statement in the PDF Reference that “the glyph width information in the Widths entry of the font dictionary and the embedded font program be consistent”. Although not explicitly mentioned in the PDF/A-1 specification, this applies to all embedded font types, including Type 3 fonts.

Developers should be aware that although the PDF/A-1 specification implies that all glyphs in the embedded data must have a matching entry in the Widths table, only those glyphs actually used need to have Width entries.

Transparency

Section 6.4 of the PDF/A-1 specification lists a series of keys that are prohibited from being present in a PDF as well as other keys that must have specific values. As noted in that section, “These prohibitions prohibit the use of transparency within a conforming file”

This list, however, is not comprehensive in terms of all keys that can be present in a PDF to represent transparency and developers should be sure to not use (and check) all such keys - for example the **/Group** key on a **/Page** dictionary.

Annotations

PDF/A provides for the presence of those types of annotations that are used for markup and commenting, but prohibits annotations that rely on resources outside of the file - specifically FileAttachment, Sound and Movie. Although not explicitly stated in 6.5.2, annotations types that are defined in PDF 1.5 and 1.6 are not permitted and therefore PDF/A-1 producers shall not include them and validators shall fail on them.

In addition, all annotations **must** be available for viewing and printing, as required by the **F** key discussion in 6.5.3.

Annotations are required to be rendered by a compliant Reader according to the (required) "Appearance Streams" embedded into the PDF itself rather than generated "on the fly". In addition, a compliant Reader may choose whether to be an "interactive" Reader or not - the difference being whether annotations are simply for presentation only or the user may actually interact with them. But in both cases, a compliant Reader **must** be sure to adhere to the requirement in 6.5.1 "to display the values of the Contents key".

Actions

The PDF Reference supports a concept whereby something will happen when the user performs an explicit or implicit action in a PDF viewer - these “things” are called Actions. PDF/A-1 permits a limited set of these Actions, which are detailed in section 6.6.1. Specifically, any action that could change the visual representation of the document or is not documented in the PDF Reference is not permitted. This includes the /Hide action which isn't specifically prohibited by PDF/A-1, but should have been.

Developers of PDF/A-1 compliancy checkers should be sure to check ALL places that an Action can be present in a PDF.

- Document Catalog/Root - **/Catalog**
- Page Dictionary - **/Page**
- Bookmarks - **/Outline**
- Annotations and AcroForms - **/Annots (/Widgets)**

A compliant Reader can choose to support Actions or not. However, whether a Reader is interactive or not, it must (as discussed in Section 6.6.3) provide access to certain values in the Action dictionaries.

Metadata

A corrigendum was created to amend 19005-1 predominately in the area of metadata due to implementation errors and updates to the XMP standard on which PDF (and by reference, PDF/A) metadata is based. Please reference the corrigendum (SC2 N397-19005) for details.

Logical Structure

The PDF/A-1a standard **does not** (unfortunately) actually make mandatory that any content be tagged (or that any substantial amount of content be tagged).

Putting this to the extreme a PDF could be a valid PDF/A-1a file if:

- Option A
 - it has a **StructTreeRoot** with a **Kids** entry with exactly one child element
 - has **Marked** value of true in the **MarkInfo** dict in the **Catalog** dict
 - none of the contents (page contents, annots, form fields) is 'tagged'
- Option B
 - it has a **StructTreeRoot** with a Kids entry with exactly one child element
 - has **Marked** value of true in the **MarkInfo** dict in the **Catalog** dict
 - one arbitrary piece of contents (some page contents, annots or form fields), but by no means all of the contents of the PDF, is 'tagged'

This is certainly **not** the intent/goal of PDF/A-1a, and developers should be sure that if they choose to create a PDF/A-1a compliant document, they should do proper and valid tagging.

Interactive Forms

A PDF 1.4 document may contain a type of PDF form called an “AcroForm”, because it is represented by an **/AcroForm** entry in the document’s **/Catalog** dictionary. AcroForms are fully supported by PDF/A-1, with restrictions as noted in Section 6.9 (as well 6.2 through 6.6).

XFA

PDF 1.5 introduced a new type of form called an XFA (extensible Forms Architecture) form, because it is represented by an **/XFA** entry in the **/AcroForm** dictionary.

XFA (eXtensible Forms Architecture) is an XML grammar for describing both static and dynamic documents including interactive forms (which is what it is used for in PDF). PDFs with XFA incorporated are created by the Adobe LiveCycle Designer forms authoring tool as well as other members of Adobe's LiveCycle Server product line.

Because XFA is laid out "on the fly" by Adobe Acrobat, it goes against the goals of PDF/A-1. As such, it is prohibited in PDF/A-1. However, since XFA can also represent the static data set of a form - work is being done by the committee to evaluate how to allow just the dataset portion of XFA in a PDF/A-1 document.

Appearances

Section 6.9 states that “Every form field shall have an appearance dictionary...”

Section 6.5.3 states “If an annotation dictionary contains the AP key, the appearance dictionary that it defines as its value shall contain only the N key.”

As such, form fields can only have an N key in their AP dicts - and not D, R, etc.

Digital Signatures

A Digital Signature in a PDF is represented by the **/Signature** type of AcroForm field (which is actually an annotation type of type **/Widget**), as is therefore bound by all the requirements of Sections 6.2 through 6.6 as well as 6.9.

Invisible Signatures

When signing a PDF using Adobe Acrobat or 3rd party solutions, users can choose to create a Visible or Invisible signature. Since an invisible signature does not show anything on the screen, some PDF creation tools may not include the PDF/A-1 required Appearance for the field. Developers of PDF signature applications should be sure to provide an appearance (with an empty content stream) for invisible signatures, in order to maintain compliance with PDF/A-1.

Color and Font Requirements

As noted above, since Signatures must have Appearances that conform to the relevant sections of the PDF/A-1 specification, the colors used must be compliant with Section 6.2.3 and all text must be associated with a font that meets the requirements of Section 6.3.

Timestamping

The use of an RFC 3161-complaint timestamp as part of the PKCS#7-compatible signature is permitted by the PDF 1.5 specification. Since this feature does not effect the visual display of the PDF, it is permitted by PDF/A-1.

Revocation

As with Timestamping, since the presence of certificate revocation information does not effect the visual display of the PDF, it is permitted by PDF/A-1, even though it is a PDF 1.5 feature.

Certified Documents/Permissions/Transforms

PDF 1.5 introduced the ability to use a Digital Signature to add a limited form of digital rights to a PDF without encryption. This feature, although it does not use encryption, is not within the stated goals of PDF/A-1 and should not be used.

Scanning and OCR

PDF/A-1 fully supports PDF documents that are created either directly from scanning or are the product of previously scanned documents (eg. from TIFF or other image formats).

PDF/A-1b compliancy allows for scans that have not gone through the Optical Character Recognition (OCR) process, since PDF/A-1b does not require searchability or tagging/structure. You can choose to use OCR, if you wish, with PDF/A-1b, but it is not required.

PDF/A-1a documents, however, will require an OCR step in order to provide the text. PDF/A-1a does not require that the image be replaced by text because most archivists recommends **against** doing this for archival documents. Instead, PDF/A-1 simply requires that the text is present, so that "Image + HiddenText" style documents are fully acceptable. Be aware, however, that such text **MUST ALSO** be tagged/structured so as to provide the semantic information as well and that the tagging/structure should be done (or at least reviewed) by hand rather than done only by automated means.

Developers of OCR applications should take advantage of the relaxation of the font embedding requirement when text is in "render mode 3" (aka Hidden Text) in order to keep their file sizes down. Also, developers should, if possible, take advantage of PDF/A-1's support for JBIG2 when creating PDFs from monochrome scans.

Development of receiver requirements

As specified in the standard, “a conforming reader is a software application that shall be able to read and appropriately process all conforming PDF/A-1 files.” The PDF/A standard has been constructed to be useful to everyone. The committee acknowledges that some receivers may wish to place additional restrictions on submitted PDF/A-1 files beyond those in the specification.

Any additional restrictions should always act in a way that further limits the options that may be used by a supplier of PDF/A-1 files. For example, a receiver may choose to prohibit JPEG-compressed images or require a minimum resolution for scanned documents.

Companies producing PDF/A-1 Readers and validation software must write to the complete PDF/A-1 standard. However, companies are encouraged to include options that enable (but do not require) receiver-identified restrictions to be applied under user control. An example of such an option would be the ability to produce a warning if a JPEG-compressed image is encountered in a file.

Archives and entities who receive and maintain PDF/A files are encouraged to review existing restrictions in the creator’s preparation specifications as they start to accept PDF/A-1 files; many of their existing limitations may be based on historical problems in writing or reading applications that may not be applicable to the PDF/A-1 compliant tools.

The following are examples of possible receiver-imposed restrictions:

- No JPEG compression is allowed.
- No MultipleMaster fonts can be used.
- All scans must be produced at a specified or minimum resolution.

The receiver’s specifications should also identify the standard characterized printing condition for which files are to be prepared, or, if non-standard conditions are to be used, provide appropriate characterization data. This identification should include a text description, but prepared ICC profiles for the specified printing condition, suitable for file preparation and inclusion in PDF/A-1 files directly, may also be provided by the receiver of the PDF/A-1 file.

Some industry groups are establishing preferred workflows that dictate the use of specific characterization data or specific profiles. For example, it is recommended that for Office documents, that the sRGB or AdobeRGB profiles be used for the OutputIntent of the document.

Document identification and metadata

The document ID in the Trailer of a PDF/A-1 file may be regarded as unique. Adding this document ID to a file reference in an electronic data interchange (EDI) environment is more likely to lead to correct identification of the intended file than using only the file name, which may not be unique. It will also distinguish between revisions of the same file since the **ID** consists of an array of two numbers - one for the document itself and the second for the specific revision of the document. Inclusion of additional file identification fields, such as the document title and modification date may also be helpful.

Document creators are strongly encouraged to follow the recommendations of the PDF Reference to ensure that the **ID** in the **Trailer** is likely to be unique and to correctly update just the second value (Revision) when appropriate.

Additionally, the use of the PDF version 1.4 **Metadata** key is required by PDF/A-1. Information placed using this mechanism may be beneficial to production processes and should be used as appropriate.

Tools used to edit PDF/A-1 files must preserve embedded metadata, and must update the xmp:VersionID field as appropriate. This is necessary in order to ensure that PDF/A-compliant documents maintain their compliancy and that any appropriate history of document processing be included.

File types and extensions

Many operating systems use internal information associated with the file to ensure that the correct application is opened when a file is selected, for example, by double-clicking on its icon.

Operating systems that use a “file extension” as part of the naming convention can build a connection between the extension and the processing application.

PDF/A-1 files are PDF files. Therefore, it is recommended that PDF/A files be named with the extension used by normal PDF files (.pdf) and that their Macintosh file type should be set to ‘PDF ’ (note the space.)

Assembling PDF/A files into a single output

OutputIntent Concerns

Multiple PDF/A-1 files can be assembled easily into a single compound entity for output if they have all been prepared for the same output condition.

If they have not been prepared for the same output condition the receiver must ensure that the individual files can achieve acceptable quality when transformed to the actual output condition to be used. Developers of assembly tools are encouraged to highlight such situations to the user, and/or to provide suitable color transformation options.

If the containing PDF/A-1 document and/or one or more target PDF/A-1 documents contain objects in device independent colour spaces, and if the profiles embedded in the OutputIntents in those files are not identical, then the colours in those files need to be transformed into a common data space as part of any assembly process to ensure that the correct gamut & tone compression is performed for each entity. Most commonly, this will be the color space of the intended output device. Where all profiles are identical, the files may be assembled directly, retaining device independent colors.

This is important because the profile is used to convert data from device independent color spaces to the color space of the intended output device. In addition to converting from one color space to another, this transformation also normally includes gamut and tone scale compression as well as color separation and black printer generation. Two profiles can be based on the same characterization data (same characterized printing condition) but contain significantly different tone compression (high key, low key, etc.), black generation algorithms, etc.

Those attributes were used to gain customer approval of the rendered data and therefore need to be preserved by using the profile associated with the data by the data provider. If profiles associated with different files are identical, then only one needs to be used for data transformation.

In many cases it may be determined whether two profiles are the same by performing a simple byte-for-byte comparison, but this may show them as different because of changes that have no technical significance (e.g. a change of copyright data), or because tags not used by a specific job have been omitted to reduce file size. The two key elements that allow a user to determine if profiles are identical are the Profile ID field, Bytes 84 to 99 of the profile header, and the profileDescriptionTag.

The Profile ID field is optional. If used, it records a checksum value generated using the MD5 fingerprinting method as defined in RFC 1321. The profileDescriptionTag is a required Tag in all profiles and containing invariant and localizable versions of the profile description for display.

If the Profile ID field is present, and the same, in multiple profiles, it provides a high degree of assurance that the profiles are the same. While not as reliable, an exact match of the

contents of the profileDescriptionTag between profiles is a good assurance that they are the same.

Note that all of these approaches will often show two profiles as different, even though they contain identical color transforms.

Metadata

When assembling multiple documents into a single one, great care must be taken by the assembling application to merge unique XMP blocks into the final XMP packet, as well as resolving any conflicts between common blocks.

In addition, especially when dealing with PDF/A-1a compliant documents, developers must include file provenance information as per Section 6.7.7. Doing this for PDF/A-1b is not required, though would be worthwhile.

Tagging/Structure

Since tagging and structural information in a PDF is both Page-relative as well as “global” to the document, an assembly application must be sure to correctly merge the global information from the Catalog in order to get a new document that continues to be valid.

AcroForms

Merging Acrobat forms from multiple PDFs into a single PDF, even without the restrictions placed by PDF/A-1, can be complex task as it involves the merging of the /AcroForm dictionaries, especially the /Fields array. Conflicts in default values and attributes can prove complex to resolve.

However, the biggest problem when merging forms is the “feature” of AcroForms where all fields of the same name MUST share the same data (though they can have different appearances). Thus merging two copies of the same form into a single document will produce a document with INCORRECT values, since there will no longer be two different values for “Name” (for example).

PDF/A Viewer Requirements

A viewer of PDF documents does not necessarily provide those requirements stipulated in the PDF/A-1 specification (section 5.4, etc.) that are incumbent on a viewer of PDF/A-1 documents. Such requirements need not be applied to all PDFs - only those that identify themselves as PDF/A compliant.

Some (but possibly not all) areas that need to be handled by a PDF/A-1 compliant viewer are:

6.2.2 *Output Intent*

Viewers need to be sure to use the Output Intent as the source profile through which Device colors are rendered.

6.2.3.2 *ICCBased colour spaces*

Viewers shall not use the Alternate color space.

6.2.3.3 *Uncalibrated colour spaces*

Viewers shall process DeviceGray through DeviceRGB or DeviceCMYK as specified by the Output Intent.

Viewers need to be sure to use the Output Intent as the source profile through which Device colors are rendered.

6.3.4 *Embedded font programs*

Viewers shall use the embedded fonts, rather than other locally resident, substituted or simulated fonts.

6.5.1 *Annotations/General*

Viewers shall provide a mechanism to display the values of the Contents key of annotation dictionaries.

6.6.3 *Hypertext links*

Viewers may choose to make hyperlinks non-actionable.

Viewers shall provide a mechanism to display the F and D keys of a GoToR action, the URI key of a URI action and the F key of a SubmitForm dictionary.

6.9 *Interactive Forms*

Viewers shall not use the form fields to change the rendered representation of the page or the content of the file at any time.

Viewers shall render the field according to the appearance dictionary without regard to the form data.

File identification

The actual version number present in the first line of the PDF document (%PDF-x.x) should not be used by a viewer to determine if the PDF meets requirements for PDF/A compliance.

Hidden & Invisible Content

PDF 1.4 spec states:

“Hidden page elements. For a variety of reasons, elements of a document’s logical content may be invisible on the page: they may be clipped, their color may match the background, or they may be obscured by other, overlapping objects. Consumer applications must still be able to recognize and process such hidden elements; for example, formerly invisible elements may become visible when a page is reflowed, or a text-to-speech engine may wish to speak text that is not visible to a sighted reader. For the purposes of Tagged PDF, page content is considered to include all text and illustrations in their entirety, whether or not they are visible when the document is displayed or printed.”

A simple example could be a page with a white background that has white text on it. This text is not visible, but would extract or be copying - as well as be read by a screen reader. This means that the visible contents of that page differs from the contents of a text extract. This is not technically in opposition to the PDF/A-1 specification, but might be considered not in the spirit. It is recommended that producers of PDF/A-1 should try not to produce documents with “hidden content”.