**ISO TC 171/SC 2 N**

Date: 2003-10-31

**ISO/CD 19005-1**

ISO TC 171/SC 2/WG 5

Secretariat: ANSI

# Document management — Electronic document file format for long-term preservation — Part 1: Use of PDF (PDF/A)

**Warning**

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: International Standard
Document subtype:
Document stage: (30) Committee
Document language: E

C:\Documents and Settings\immdca10\Desktop\COMPLETE\I031031D.doc STD Version 2.1

# Contents

Page

 

# Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 19005-1 was prepared by Technical Committee ISO/TC 171, *Document management applications*, Subcommittee SC 2, *Application issues*.

This second/third/... edition cancels and replaces the first/second/... edition (), [clause(s) / subclause(s) / table(s) / figure(s) / annex(es)] of which [has / have] been technically revised.

ISO 19005 consists of the following parts, under the general title *Document management — Electronic document file format for long-term preservation*:

— *Part 1: Use of PDF (PDF/A)*

# Introduction

PDF is a digital format for representing documents, whether they are created natively in PDF, converted from other electronic formats, or digitised from paper or microform. 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. These PDF documents must remain useable and accessible across multiple generations of technology. The future use of, and access to, these objects depends upon maintaining their visual appearance as well as their higher-order properties, such as the logical organization of pages, sections, and paragraphs, machine recoverable text stream in natural reading order, and a variety of administrative, preservation, and descriptive metadata.

Adobe Systems Incorporated makes the PDF specification publicly available. However, the inclusive, feature-rich nature of the format requires that additional constraints be placed on its use to make it suitable for the long-term preservation of electronic documents. This International Standard specifies how to represent unambiguously:

— The visual appearance of PDF documents

— The associated structural and semantic information that maps PDF components into more meaningful concepts

These goals are accomplished by identifying the set of PDF components that may be used and restrictions on the form of their use.

This Standard should lead to the development of various applications, such as products that read, render, write, and validate conforming PDF objects. Different products will incorporate various capabilities to prepare, interpret, and process conforming objects based on the application needs as perceived by the suppliers of those products. However, it is important to note that a conforming application must be able to read and process appropriately all files complying with a specified conformance level.

[TBD] maintains an ongoing series of application notes for guiding developers and users of this ISO standard. These application notes are available at [URL].

# Document management — Electronic document file format for long-term preservation — Part 1: Use of PDF (PDF/A)

## 1 Scope

This International Standard specifies how to use the Portable Document Format (PDF) for long-term preservation of electronic documents. It is applicable to documents containing combinations of character, raster, and vector data. This International Standard does not address the following:

— Methods of capture or conversion

— Rendering issues related to physical devices

— Specific physical methods of storing these documents such as the media and storage conditions, required computer hardware, and/or operating systems

## 2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

*Adobe CMap and CIDFont Files Specification*, Technical Specification #5014, Version 1.0, 8 October 1996, Adobe Systems Incorporated. Available from Internet <http://partners.adobe.com/asn/developer/pdfs/tn/5014.CMap_CIDFont_Spec.pdf>

*Adobe Type 1 Font Format*, 1990, Adobe Systems Incorporated (ISBN 0-201-57044-0). Available from Internet <http://partners.adobe.com/asn/developer/pdfs/tn/T1_SPEC.PDF>

*The Compact Font Format Specification*, Technical Note #5176, 16 March 2000, Adobe Systems Incorporated. Available from Internet <http://partners.adobe.com/asn/developer/pdfs/tn/5176.CFF.pdf>

*Errata for PDF Reference*, third edition, 18 June 2003. Available from Internet <http://partners.adobe.com/asn/acrobat/docs/PDF14errata.txt>

ICC.1:1998-09, *File Format for Color Profiles*, International Color Consortium. Available from Internet <http://www.color.org/ICC-1_1998-09.PDF>

ICC.1A:1999-04, *Addendum 2 to Spec. ICC.1:1998-09*, International Color Consortium. Available from Internet <http://www.color.org/ICC-1A_1999-04.PDF>

*PDF Reference: Adobe Portable Document Format*, Version 1.4, Adobe Systems Incorporated – 3rd ed. (ISBN 0-201-75839-3). Available from Internet <http://partners.adobe.com/asn/acrobat/docs/File_Format_Specifications/PDFReference.pdf>

*Tags for the Identification of Languages*, RFC 1766, March 1995. Available from Internet <http://www.ietf.org/rfc/rfc1766.txt>

*TrueType Reference Manual*, 17 September 1999, Apple Computer, Inc. Available from Internet <http://developer.apple.com/fonts/TTRefMan/>

*XMP: Extensible Metadata Platform*, Version 1.5, 14 September 2001, Adobe Systems Incorporated. Available from Internet <http://partners.adobe.com/asn/developer/xmp/pdf/MetadataFramework.pdf>

## 3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

**3.1**
**conformance level**
identified set of restrictions and requirements to which files and readers must comply [ISO 15930-1]

**3.2**
**dictionary**
associative table containing key-value pairs, specifying the name and value of an attribute for objects, which is generally used to collect and tie together the attributes of a complex object [ISO 15930-1]

**3.3**
**electronic document**
electronic representation of a page-oriented aggregation of text and graphics that can be reproduced on paper or optical microform without significant loss of its information content

**3.4**
**end-of-file marker**
the five character sequence **%%EOF** marking the end of a PDF file

**3.5**
**end-of-line marker**
**EOL marker**
one or two character sequence marking the end of a line of text, consisting of a required **CARRIAGE RETURN** character (U+000D) and an optional **LINE FEED** character (U+000A)

NOTE The parenthetic hexadecimal values indicate the encoding-neutral Unicode values for the EOL marker characters [5].

**3.6**
**font**
identified collection of graphics that may be glyphs or other graphic elements [ISO 15930-1]

**3.7**
**glyph**
recognizable abstract graphic symbol that is independent of any specific design [ISO/IEC 9541-1]

**3.8**
**long-term**
period of time long enough for there to be concern about the impacts of changing technologies, including support for new media and data formats, and of a changing user community, on the information being held in a repository. This period extends into the indefinite future [ISO 14721]

**3.9**
**PDF**
**Portable Document Format**
file format defined in the *PDF Reference* and its *Errata* [ISO 15930-1]

**3.10**
**reader**
software application that is able to read and process files appropriately [ISO 15930-1]

**3.11**
**writer**
software application that is able to write files [ISO 15930-1]

EDITOR'S NOTE It may be helpful to follow the example of PDF/X and add a Notation clause at this point:

"**4 Symbols and Notation**

PDF operators, PDF keywords, the names of keys in PDF dictionaries, and other predefined names are written in a bold sans serif type font; operands of PDF operators or values of dictionary keys are written in an italic sans serif font. For example: the *Default* value for the **TR2** key.

Individual characters may be identified by their Unicode character name written in uppercase in a bold sans serif type font and a parenthetic hexadecimal Unicode value: **CARRIAGE RETURN** (U+000D)."

# 4 Conformance

## 4.1 General

For the purposes of this part of ISO 19005, references to the "*PDF Reference*" are to the normative reference *PDF Reference: Adobe Portable Document Forma*t as amended by *Errata for PDF Reference* as identified in Clause 2.

The base-line criterion for PDF/A conformance is adherence to Version 1.4 of the *PDF Reference*. A conforming PDF/A document may include any valid PDF 1.4 feature that is not explicitly forbidden by this part of ISO 19005. Features described in PDF specifications prior to Version 1.4 that are not explicitly documented in the *PDF Reference* should not be used.

In recognition of the varying preservation needs of the diverse user communities making use of PDF documents, this part of ISO 19005 defines two PDF/A conformance levels: minimally conforming and fully conforming.

## 4.2 Minimal conformance level

A minimally conforming PDF/A document is a PDF document that meets all of the requirements specified in this part of ISO 19005 to ensure that its rendered visual appearance is preservable over the long-term. All minimally conforming PDF/A documents shall be conforming PDF 1.4 documents that meet the requirements of this part of ISO 19005 except for those defined in Clauses 7.9 and 12. These requirements may not necessarily provide a PDF/A document with sufficiently rich internal information to allow for the automatic search or retrieval of the textual content of that document.

A PDF document meeting the minimal conformance requirements outlined in this sub-clause, and described fully in subsequent clauses, is said to be a "minimally conforming PDF/A document" or a PDF document that meets the "PDF/A minimal conformance level."

## 4.3 Full conformance level

A fully conforming PDF/A document is a PDF document that meets all of the requirements specified in this part of ISO 19005 to ensure that the document logical structure and content text stream, in natural reading order, are preservable over the long-term. All fully conforming PDF/A documents shall be minimally conforming PDF/A documents that meet additional requirements that define specific constraints on, and required uses of, PDF 1.4 features with regard to character Unicode mapping and logical document structure

based on the Tagged PDF framework. Clauses 7.9 and 12 of this part of ISO 19005 clearly indicate the requirements for full conformance above those needed for minimal conformance.

The requirements for full conformance may place greater burdens on PDF/A writers but these requirements should allow for a higher level of document preservation service and confidence over time. Additionally, full conformance may facilitate the accessibility of PDF/A documents for physically impaired users.

A PDF document meeting the full conformance requirements outlined in this sub-clause, and described fully in subsequent clauses, is said to be a "fully conforming PDF/A document" or a PDF document that meets the "PDF/A full conformance level."

### 4.4 Conforming PDF/A readers

A conforming PDF/A reader shall follow all requirements regarding reader functional behaviour specified in this part of ISO 19005.

NOTE The requirements of this part of ISO 19005 with respect to reader behaviour are stated in terms of general functional requirements applicable to all readers. This part of ISO 19005 does not prescribe any specific technical design, user interface, or implementation details of conforming readers.

The rendering of conforming PDF/A files shall be performed as defined in the *PDF Reference* subject to the further requirements specified by this part of ISO 19005. Features described in PDF specifications prior to Version 1.4 that are not explicitly documented in the *PDF Reference* may be ignored by conforming readers.

## 5 File format

### 5.1 General

This clause addresses overall file format issues and the base elements that form the general structure of a PDF/A file.

### 5.2 File header

No data shall precede the file header.

The file header line shall be immediately followed by a comment containing four characters, each of whose encoded values shall be greater than 127.

### 5.3 File trailer

The file trailer dictionary shall contain all of the items listed in Table 1. The keywords **Encrypt** and **Info** shall not be used in the trailer dictionary. No data shall follow the end-of-file marker except a single optional end-of-line marker.

**Table 1 — Trailer dictionary entries**

| Key | Type | Value |
|---|---|---|
| **Size** | integer | Total number of entries in the cross reference table |
| **ID** | array | Array of two hex strings specifying file identifiers |

### 5.4 Cross reference table

The cross reference table shall correctly specify the location of each indirect object in the PDF/A file.

A cross reference subsection header, starting object number, and range shall be separated by a single white-space character.

The **xref** keyword and the cross reference subsection header shall be separated by a single EOL marker.

## 5.5 Document information dictionary

The document information dictionary shall not be present. The proper method for supplying descriptive metadata within a conforming PDF/A file is presented in Clause 11.

## 5.6 Extraneous binary data

Except for an optional EOL marker, no data shall exist between the **endobj** marking the end of one indirect object and the object number marking the start of the next indirect object in the file.

## 5.7 String objects

A literal string is a sequence of characters enclosed in parentheses or a sequence of hexadecimal data enclosed in angle brackets **< >**.

Literal strings that are broken across lines shall contain a **BACKSLASH** character (U+005C) immediately before any EOL markers.

Octal representations of single characters shall contain exactly four characters: the **BACKSLASH** character followed by three digits, each in the range of **0** to **7**.

Hexadecimal strings shall contain an even number of characters, each in the range **0** to **9**, **A** to **F**, or **a** to **f**.

White-space shall not occur within a hexadecimal string.

## 5.8 Stream objects

A stream object is a sequence of bytes delimited by the **stream** and **endstream** keywords.

The **stream** keyword shall be followed by a **CARRIAGE RETURN** (U+000D) and **LINE FEED** (U+000A). The **endstream** keyword shall be preceded by a **CARRIAGE RETURN** and **LINE FEED**.

The value of the **Length** key specified in the stream dictionary shall match the number of bytes in the file following the **CARRIAGE RETURN** and **LINE FEED** pair after the **stream** keyword and preceding the **CARRIAGE RETURN** and **LINE FEED** before the **endstream** keyword.

NOTE These requirements remove potential ambiguity regarding the beginning and ending of stream content.

## 5.9 Indirect objects

The object number, generation number, and **obj** keyword shall be located on a single line and shall be separated by a single white-space character.

The object number and **endobj** keyword shall be preceded by an EOL marker. The **obj** and **endobj** keywords shall be followed by an EOL marker.

## 5.10 Linearized PDF

Linearization is permitted but any linearization information supplied within a file should be ignored by a conforming PDF/A reader.

### 5.11 Filters

The **ASCIIHexDecode**, **ASCII85Decode**, and **LZWDecode** filters shall not be permitted.

NOTE The ASCII filters are little used in practice. The use of the LZW decompression algorithm is subject to intellectual property constraints.

### 5.12 Streams

A stream object dictionary shall not contain the **F**, **FFilter**, or **FDecodeParams** keys.

NOTE The use of these keys would permit the existence of document content external to the file.

### 5.13 Implementation limits

A conforming PDF/A file shall not violate any of the limits specified in Table C.1 of the *PDF Reference*.

### 5.14 Optional content

The document catalog dictionary shall not contain the **OCProperties** key.

NOTE This key is defined in PDF 1.5 for the use of optional content that can be used to generate alternative renderings of a document.

## 6 Graphics

### 6.1 General

This clause describes restrictions placed on both conforming PDF files and readers. It is intended to address graphical rendering issues that do not involve fonts and interactive elements.

### 6.2 Colourspaces

All colours shall be specified in a device-independent manner, either directly by the use of a device independent colourspace, or indirectly by the use of an **OutputIntent**. A conforming file may use any colourspace specified in the *PDF Reference*, except as restricted below.

NOTE Device-independence of colour specifications is required to allow predictable rendering.

#### 6.2.1 ICCBased colourspaces

Any **ICCBased** colourspace shall be embedded and shall conform to ICC specification ICC.1:1998-09 and its addendum ICC.1A:1999-04.

A conforming reader shall render **ICCBased** colourspaces as specified by the ICC specification, and shall not use the **Alternate** colourspace specified in an ICC profile stream dictionary.

#### 6.2.2 Uncalibrated colourspaces

A conforming file may use the **DeviceGray** colourspace and at most one of the other two uncalibrated colourspaces defined in PDF: **DeviceRGB** and **DeviceCMYK**. If an uncalibrated colourspace is used in a file then that file's document catalog dictionary shall contain an **OutputIntents** array with exactly one member and that member shall be an **OutputIntent** dictionary with the following characteristics:

— The **OutputCondition** key shall be present with a non-empty string as its value

— The **DestOutputProfile** key shall be present and its value shall be a profile stream that contains an ICC profile defining a colourspace that has the same number of components as the device dependent colourspace that is used in the file and that conforms to the requirements for **ICCBased** colourspaces used as source colour specifications

When rendering a **DeviceGray** colour specification in a document whose **OutputIntent** is an RGB profile a conforming reader shall convert the **DeviceGray** colour specification to RGB by the method described in Section 6.2.1 of the *PDF Reference*.

When rendering a **DeviceGray** colour specification in a document whose **OutputIntent** is a CMYK profile a conforming reader shall convert the **DeviceGray** colour specification to **DeviceCMYK** by the method described in Section 6.2.2 of the *PDF Reference*.

When rendering colours specified in a device-dependent colourspace a conforming reader shall use the profile specified in the **OutputIntents** array as the source colourspace.

#### 6.2.3 Named colorants in Separation and DeviceN colourspaces

A conforming reader shall follow the following rules when rendering colourspaces based on **DeviceN** or **Separation** colourspaces:

— If the named colorants in the colourspace are all from the list **Cyan**, **Magenta**, **Yellow**, **Black**, and the document's **OutputIntent** is a CMYK profile, then the colorants shall be treated as components of the colourspace specified by the **OutputIntent** and the alternate colourspace shall not be used

— If the named colorants in the colourspace are all from the list **Red**, **Green**, **Blue**, and the document's **OutputIntent** is an RGB profile, then the colorants shall be treated as components of the colourspace specified by the **OutputIntent** and the alternate colourspace shall not be used

— If the only named colorant is **Gray** and the document's **OutputIntent** is a Gray profile, then the colorant shall be treated as the component of the colourspace specified by the **OutputIntent** and the alternate colourspace shall not be used

— In all other cases the **Alternate** colourspace shall be used

### 6.3 Images

An Image dictionary shall not contain the **Alternates** key or the **OPI** key.

If an Image dictionary contains the **Interpolate** key, its value shall be *false*.

Use of the **Intent** key shall conform to the rules in Clause 6.8.

### 6.4 Form XObjects

A form XObject dictionary shall not contain the **OPI** key.

### 6.5 Reference XObjects

A conforming file shall not contain any reference XObjects.

### 6.6 PostScript XObjects

A conforming file shall not contain any PostScript XObjects.

### 6.7 Extended graphics state

An **ExtGState** dictionary shall not contain the **TR** key. An **ExtGState** dictionary shall not contain the **TR2** key with a value other than *Default*. A conforming reader may ignore any instance of the **HT** key in an **ExtGState** dictionary.

Use of the **RI** key shall be governed by Clause 6.8.

### 6.8 Rendering intents

Rendering intents are permitted in both **ExtGState** and **Image** dictionaries. Where a rendering intent is specified its value shall be one of the four values defined in the *PDF Reference*: *RelativeColorimetric*, *AbsoluteColorimetric*, *Perceptual*, or *Saturation*.

### 6.9 Content streams

A content stream shall not contain any operators not documented in the *PDF Reference*, even if such operators are bracketed by the **BX**/**EX** compatibility operators.

## 7 Fonts

### 7.1 General

The intent of the requirements stated in this clause is to ensure that future rendering of the textual content of a PDF file matches the static appearance of the file as originally created, on a glyph by glyph basis. Additionally, these requirements allow the recovery of semantic properties for each character of the textual content.

### 7.2 Font types

Only fully conformant Type 0, Type 1, Type 3, and TrueType fonts shall be referenced within a PDF/A file. Type 0 font conformance is defined by Section 5.6 of the *PDF Reference*. Type 1 font conformance is defined by adherence to the *Adobe Type 1 Font Format* document or the *Compact Font Format Specification*; Type 3 font conformance is defined by Section 5.5.4 of the *PDF Reference*; TrueType font conformance is defined by the *TrueType Reference Manual*.

For the purposes of the requirements stated by this part of ISO 19005, multiple master fonts are considered a special case of Type 1 fonts; any requirement explicitly stated with regard to Type 1 fonts also shall be implicitly required with regard to multiple master fonts.

NOTE 1 The allowable valid font types are constrained to those whose definition is unambiguous and publicly available.

NOTE 2 It is the responsibility of the file writer to ensure the conformance of all fonts. This part of ISO 19005 does not prescribe the manner in which conformance is determined.

### 7.3 Composite fonts

For all composite (Type 0) fonts referenced within a PDF/A file, the **CIDSystemInfo** entry of the **CIDFont** and **CMap** dictionaries shall be compatible; in other words, the **Registry** and **Ordering** strings of each of the **CIDSystemInfo** dictionaries shall be identical, as described in Section 5.6.2 of the *PDF Reference*.

#### 7.3.1 CIDFonts

For all Type 2 **CIDFonts**, the **CIDFont** dictionary shall contain a **CIDToGIDMap** entry that shall be a stream mapping from CIDs to glyph indices or the name **Identity**, as described in Table 5.13 of the *PDF Reference*.

### 7.3.2 Cmaps

The integer value of the **WMode** entry in a **CMap** dictionary shall be identical to the **WMode** value in the embedded **CMap** stream.

## 7.4 Embedded font programs

All Type 0, Type 1, and TrueType fonts referenced for rendering within a PDF/A file shall be embedded within that file, including the 14 standard Type1 fonts if they are used.

NOTE 1 An example of a font referenced but not rendered is text mode 3 (invisible).

All Type 0 **CIDFont** programs shall be in the compact font format. Type 1 font programs shall be embedded either in the original (non-compact) Type 1 font format or in the compact font format. All TrueType font programs, including those for Type 2 **CIDFonts**, shall be in the TrueType format. All **CMap** streams shall follow the syntax defined in *Adobe CMap and CIDFont Files Specification*.

Only fonts that are legally embeddable in a file for unlimited, universal rendering shall be used.

All PDF/A conforming readers shall use the embedded fonts, rather than other locally resident, substituted, or simulated fonts, for the visual reproduction of all text.

The requirements for font program metadata are described in Clause 11.9.

NOTE 2 Only fonts whose characters are referenced within a file need to be embedded in that file. Furthermore, as stated in Clause 7.5 font programs can be for font subsets, as long as the embedded programs provide glyph definitions for all characters referenced within the file. Embedding the font programs allows any PDF/A conforming reader to reproduce correctly all glyphs in the manner in which they were originally published without reference to possibly ephemeral external resources. By definition, Type 3 fonts always include an embedded font program in the form of per-glyph streams of PDF graphics operators that paint the glyphs.

NOTE 3 This part of ISO 19005 does not allow the embedding of fonts whose legality depends upon special agreement with the font copyright holder. Such an allowance would place unacceptable burdens on an archive to verify the existence, validity, and longevity of such claims.

## 7.5 Font resources

For all Type 3 fonts, the font dictionary shall include a Resources dictionary, listing all named resources required by the glyph descriptions, as described in Table 5.9 of the *PDF Reference*.

NOTE This requirement may help to identity external resources that should properly be embedded within the PDF/A file.

## 7.6 Font subsets

Type 0 **CIDFont** and Type 1 and TrueType font subsets, as described by Section 5.5.3 of the *PDF Reference*, may be used as long as the embedded font programs define all of the font glyphs used within the file.

For all Type 1 font subsets referenced within a PDF/A file, the font descriptor dictionary shall include a **CharSet** string listing the character names defined in the font subset, as described in Table 5.18 of the *PDF Reference*.

For all **CIDFont** subsets referenced within a PDF/A file, the font descriptor dictionary shall include a **CIDSet** stream identifying which CIDs are present in the embedded **CIDFont** file, as described in Table 5.20 of the *PDF Reference*.

NOTE The use of font subsets allows a potentially substantial reduction in the size of PDF/A files.

### 7.7 Font metrics

For all embedded fonts, a conforming PDF/A reader shall use the font metrics specified inside the embedded font program and shall ignore the metrics given in the required **Widths** entry of the font dictionary.

### 7.8 Character encodings

All non-symbolic TrueType fonts shall specify **MacRomanEncoding** or **WinAnsiEncoding** as the value of the **Encoding** entry in the font dictionary. All symbolic TrueType fonts shall not specify an **Encoding** entry in the font dictionary, and their font programs' cmap tables shall contain exactly one encoding.

NOTE This requirement makes normative the suggested guidelines described in Section 5.5.5 of the *PDF Reference*.

### 7.9 Unicode character maps

This sub-clause is applicable only for files meeting the full conformance level of this part of ISO 19005. For minimal conformance the requirements of this sub-clause can be ignored.

The font dictionary shall include a **ToUnicode** entry whose value is a CMap stream object that maps character codes to Unicode values [5], as described in Section 5.9 of the *PDF Reference*.

Fonts meeting the any of following three conditions are exempted from this requirement:

1) Fonts that use the predefined encodings **MacRomanEncoding**, **MacExpertEncoding**, or **WinAnsiEncoding**, or that use the predefined **Identify-H** or **Identity-V** Cmaps

2) Type 1 fonts whose character names are taken from the Adobe standard Latin character set or the set of named characters in the Symbol font, as defined in Appendix D of the *PDF Reference*

3) Type 0 fonts whose descendent **CIDFont** uses the **Adobe-GB1**, **Adobe-CNS1**, **Adobe-Japan1**, or **Adobe-Korea1** character collections

NOTE The Unicode mapping allows the retrieval of semantic properties about every character referenced in the file.

## 8 Transparency

The **SMask** key shall not be used in an **ExtGState** object or in an Image **XObject** with any value other than *None*.

A **Group** object shall not be included in a form **XObject** if it includes an **S** key with a value of *Transparency*.

The following keys, if present in an **ExtGState** object, shall have the values shown:

**BM** *Normal* or *Compatible*

**CA** *1.0*

**ca** *1.0*

NOTE These provisions prohibit the use of transparency within a conforming PDF/A file. The visual effect of partially transparent graphics can be achieved using techniques other than the use of the PDF 1.4 transparency keys, including pre-rendered data or flattened vector objects. The use of such techniques does not prevent a file from being PDF/A conformant.

# 9 Annotations

## 9.1 General

Conforming PDF/A readers shall provide a mechanism to display the actual contents of all annotations. The actual content is construed to be the value of the **Contents** key of the annotation dictionary, not the visual presentation specified by the annotation's appearance stream.

NOTE This part of ISO 19005 does not prescribe the specific behaviour or technical implementation details that readers may use to implement this functional requirement.

## 9.2 Annotation types

Annotation types not defined in the PDF Reference shall not be permitted. Additionally, the **FileAttachment**, **Sound**, and **Movie** types shall not be permitted.

NOTE Support for multimedia is outside the scope of this part of ISO 19005.

## 9.3 Annotation dictionaries

For annotation types that do not display text, the **Contents** key of the annotation dictionary should be specified with an alternative description of the annotation's contents in human-readable form.

An annotation dictionary shall not contain the **CA** key with a value other than *1.0*.

# 10 Actions

## 10.1 General

The **Launch**, **Sound**, **Movie**, **ResetForm**, **ImportData**, and **JavaScript** actions shall not be permitted. Additionally, the deprecated set-state and no-op actions shall not be permitted. Named actions other than **NextPage**, **PrevPage**, **FirstPage**, and **LastPage** shall not be permitted. In response to each of the four allowed named actions, conforming PDF/A readers shall perform the appropriate action described in Table 8.45 of the *PDF Reference*.

## 10.2 Trigger events

An interactive form field shall not include an **AA** entry for an additional-actions dictionary. The document catalog shall not include an **AA** entry for an additional-actions dictionary.

NOTE These additional-actions dictionaries are prohibited to exclude the use of arbitrary JavaScript within PDF/A files.

## 10.3 Hypertext links

Conforming PDF/A readers may choose to make hyperlinks non-actionable but they shall provide a mechanism to display the **F** and **D** keys of a **GoToR** action dictionary, the **URI** key of a **URI** action dictionary, and the **F** key of a **SubmitForm** action dictionary.

NOTE Since hyperlinks transfer the thread of execution outside the control of a reader, a reader may choose to make them not actionable. However, for purposes of archival disclosure of the complete information content of PDF/A documents, readers must provide some mechanism to expose the destination of all hyperlinks. However, this part of ISO 19005 does not prescribe the specific behaviour or the technical implementation details that readers may use to meet this functional requirement

# 11 Metadata/XML

## 11.1 General

This clause specifies requirements for metadata within PDF/A files. Metadata is essential for effective management of a file throughout its life cycle. A file depends on metadata for identification and description, as well as for documenting appropriate technical and administrative matters. As a result, PDF/A file producers likely will have to comply with various domain-specific metadata requirements. This specification outlines a structured, consistent process that supports a broad variety of metadata requirements.

## 11.2 Properties

The Catalog dictionary of a conforming PDF/A file shall contain the **Metadata** key. The metadata stream that forms the value of that key shall conform to Version 1.5 of *XMP – Extensible Metadata Platform*. All metadata properties pertaining to a file shall be embedded in the file as XML packets. Metadata properties shall also be either defined in Adobe XML schemas or defined in one or more extension schemas that comply with XMP requirements. The metadata stream shall be visible as plain text to non-PDF/A aware tools and so shall be unfiltered.

EDITOR'S NOTE We discussed adding a crosswalk from the document info dictionary metadata properties to XMP metadata properties.

## 11.3 Normalization

Metadata shall be entered, saved, and retained in a normalized fashion to facilitate interchange and support consistent depiction of metadata by conforming PDF/A readers. All normalization shall be defined by schemas. The following normalizations are mandatory:

— When a property is represented by start and end tags, e.g. “<prop>value</prop>”, whitespace at the start and end of the value shall be removed. If the value consists of nothing but whitespace, it shall be reduced to a single SPACE (U+0020) character.

EDITOR'S NOTE We should add the XML definition of whitespace, which is different from PDF whitespace.

— When a property is represented as an attribute, the value is the entire quoted attribute value including all whitespace.

— Properties defined as sequences or bags may be input as repeated simple properties and normalized to a sequence or bag according to the schema. The degenerate case of a single simple property where a bag or sequence is expected shall be accepted and normalized.

— Repeated properties in the input shall be normalized to a sequence container if there is no schema.

— Bags and sequences with just one element may be output as a single simple property if the schema does not specify otherwise.

— Localizable properties with only one localization (value) shall be accepted as a simple property. This shall be normalized to an alternative container with one item having the 'x-default' language.

— Localizable properties with just an x-default value may be output as a simple property if the schema does not say otherwise.

NOTE These normalizations are based on recommendations in Section 3.4.8 of *XMP – Extensible Metadata Platform*.

EDITOR'S NOTE Adobe states that some of these normalizations need to be modified.

## 11.4 XMP header

The **bytes** attribute shall not be used in XMP headers.

If the XML encoding for a packet is other than UTF-8, the encoding attribute shall be used. The packet body shall conform to the encoding indicated in the header.

## 11.5 File identifiers

A PDF/A file should have one or more metadata properties to characterize, categorize, and otherwise identify the file. This specification does not mandate any specific identification scheme. Identifiers may be externally based, such as an International Standard Book Number (ISBN) or a Digital Object Identifier (DOI), or internally based, such as a Globally Unique Identifier (GUID)/Universally Unique Identifier (UUID) or another designation assigned during workflow operations. Identifiers may be included through use of the **xap:Identifier** property, use of the **xapMM:DocumentID**, **xapMM:VersionID**, and **xapMM:RenditionClass** properties, or use of properties from an extension schema. Any identification system may be used so long as the properties comply with XMP requirements and this part of ISO 19005.

EDITOR'S NOTE Additional discussion is needed with Adobe on how best to provide for an **xap:Identifier** property in either the PDF/A standard or the XMP specification.

## 11.6 File provenance information

A metadata audit trail in the form of chronological entries in the **xapMM:History** property should indicate all steps taken to create, transform, or otherwise instantiate the file. In cases where original files are transformed into PDF/A format, entries should document processing (e.g., transformed from PDF 1.4 to PDF/A); altering file content or functionality (e.g., embedded JavaScript and audio objects were not retained); handling of preexisting metadata (e.g., all InfoDictionary values converted to XMP); and any other processes that have an impact on file content. In cases where PDF/A is the original format, the **xapMM:History** property should include documentation of workflow processes (e.g., descriptions of activities and handoffs), citations to policies governing file handling (e.g., titles of official directives under which files are collected, processed, and used), names and versions of software tools, as well as other matters that are needed to indicate the context of the document's creation and use. Each action should include a timestamp.

A second audit trail should consist of retained versions of all XMP metadata values that have been edited, cancelled, or otherwise changed as a file moves through its life cycle. A timestamp for each value shall provide a chronology of changes to metadata associated with file receipt, review/approval, indexing, filing, transfer between custodians, and other activities.

## 11.7 Extension schemas

All extension metadata used in conjunction with a PDF/A file shall be based on extension schemas. All extension schemas shall have a unique name in the form of a Universal Resource Identifier (URI) and shall consist of: 1) a table in XML format that conforms with the format outlined in Table 4; or 2) a machine readable format that conforms with the W3C RDF Schema Specification. All extension schemas shall be embedded within the file as separate XML packet streams in a manner that does not alter the visual appearance of the document. A conforming PDF/A reader shall parse and display all properly formed extension metadata and extension schemas.

**Table 2 — Extension schema template**

| Property | Valid type | Description | Category |
|---|---|---|---|
| [namespace prefix: property name:] | [Text, Integer, URI, etc.] | [Description of property] | [Internal, External or Relational] |

EDITOR'S NOTE Option (2) would be acceptable only if it is supported by Adobe. Additional discussion is needed with Adobe concerning proposals to modify the XMP specification to include support for machine readable schemas.

### 11.8 Validation

All XMP metadata shall be validated for conformance with XML/RDF syntax, as well as for proper values and data types whenever a file is saved or resaved.

EDITOR'S NOTE Additional discussion is needed with Adobe concerning how to permit XMP data typing.

### 11.9 Font metadata

For all embedded Type 0, Type 1, or TrueType font programs, the embedded font file stream dictionary should include a **Metadata** entry whose value is an XMP metadata stream. The following XMP metadata elements should be supplied: **xap:Title**, giving the name of the font; **xapRights:Copyright**, giving the copyright statement; **xapRights:Marked**, with the Boolean value **true**; **xapRights:Owner**, giving the legal owner of the font; and **xapRights:UsageTerms**, giving a statement of the licensing terms under which the font is being used. Additional XMP metadata may be included at the discretion of the file writer.

NOTE Font rights information is helpful in order to preserve the identity and scope of the intellectual property rights of the font copyright holder. While many fonts embed statements of copyright and licensing terms within the font itself, this is not a uniform practice. Therefore it is advantageous to require the explicit representation of rights statements in the PDF/A file. Even though this may be redundant, it obviates the necessity for some future system to have the ability to parse through the particular internal structure of font programs.

### 11.10 Version and conformance level identification

EDITOR'S NOTE This is a placeholder for requirements concerning the identification of the purported PDF/A standard version compliance and the conformance level.

## 12 Logical structure

### 12.1 General

This clause is applicable only for files meeting the full conformance level of this part of ISO 19005. For minimal conformance the requirements of this clause can be ignored.

The intent of the requirements in this clause is to ensure the recovery of a PDF file's textual content as a sequence of words defined in the natural reading order of the language in which they are written. Similarly, the individual characters of each word must be recoverable in their natural reading order. Furthermore, these requirements allow the recovery of higher-level semantic information concerning the logical structure of the document.

### 12.2 Tagged PDF

A PDF/A file shall meet of all the requirements set forth for Tagged PDF in Section 9.7 of the *PDF Reference*.

NOTE Tagged PDF defines conventions for explicitly declaring and describing the logical structural aspects of document content.

#### 12.2.1 Mark information dictionary

The document catalog shall include a **MarkInfo** dictionary whose sole entry, **Marked**, shall have a value of *true*.

NOTE This setting indicates that the file conforms to the Tagged PDF conventions.

### 12.2.2 Artifacts

To the fullest extent possible, pagination features such as running heads or page numbers, cosmetic layout features such as footnote rules or background screens, and production aids such as cut marks and color bars should be specified as pagination, layout, and page artifacts, respectively, as described in Section 9.7.2 of the *PDF Reference*.

### 12.2.3 Word breaks

Within show strings, word breaks shall be explicitly indicated by the presence of one or more spacing characters between all of the individual words in the show string. If a word ends at a show string boundary, one or more spacing characters shall be inserted at the end of the show string. Note that a single word may span two or more show strings; word breaks are indicated only by the explicit presence of one or more spacing characters, not by the boundaries of a show string. For the purposes of indicating word breaks, a sequence of two or more consecutive spacing characters is semantically equivalent to a single spacing character.

The spacing characters are: **HORIZONTAL TABULATION** (Unicode U+0009), **LINE FEED** (U+000A), **VERTICAL TABULATION** (U+000B), **FORM FEED** (U+000C), **CARRIAGE RETURN** (U+000D), **SPACE** (U+0020), **NO-BREAK SPACE** (U+00A0), **EN SPACE** (U+2003), **EM SPACE** (U+2003), **ZERO WIDTH SPACE** (U+200B), and **IDEOGRAPHIC SPACE** (U+3000).

NOTE Even for writing systems that do not normally include spacing characters between words in typographical representations, it is important that the spacing characters be included in the PDF/A file to remove ambiguity regarding word boundaries.

### 12.2.4 Structure hierarchy

The logical structure of the PDF document shall be described by a structure hierarchy rooted in the **StructTreeRoot** entry of the document catalog, as described in Section 9.6 of the *PDF Reference*.

Each structure element dictionary in the structure hierarchy shall have a **Type** entry with the name value of **StructElem**. Any process that purports to determine PDF/A conformance shall report an error condition if structure element dictionary without a **Type** entry with the name value **StructElem** is discovered in a PDF file.

NOTE The explicit documentation of a document's logical structure may prove valuable to future efforts to recover the document's full semantic value for the purposes of rendering or migration to other data formats. PDF/A writers should attempt to capture a document's logical structure hierarchy to the finest granularity possible, making use of the standard structure types for grouping elements, block-level structure elements, paragraph-like elements, list elements, table elements, inline-level structure elements, link elements, and illustration elements, as defined in Section 9.7.4 of the *PDF Reference*, to the fullest extent possible.

### 12.2.5 Structure types

To the fullest extent possible, the definition of block-level structuring elements should follow the strongly structured paradigm as described in Section 9.7.4 of the *PDF Reference*.

All non-standard structure types shall be mapped to the nearest functionally equivalent standard type, as defined in Section 9.7.4 of the *PDF Reference*, in the role map dictionary of the structure tree root. This mapping may be indirect; within the role map a non-standard type can map directly to another non-standard type, but eventually, the mapping must arrive at a standard type.

## 12.3 Natural language specification

The default natural language for all text in a document shall be specified by the **Lang** entry in the document catalog.

To the fullest extent possible, all textual content within a document that differs from the default language should be indicated by use of a **Lang** property attached to a marked-content sequence, or by a **Lang** entry in a structure element dictionary, as described in Section 9.8.1 of the *PDF Reference*.

The value of the **Lang** entry in the document catalog, structure element dictionary, or property list shall be a language identifier as defined by *RFC 1766, Tags for the Identification of Languages*, as described in Section 9.8.1 of the *PDF Reference*.

NOTE 1 The distinction between words foreign to a language and foreign words incorporated by common usage into a language is problematic. The intent of these requirements is to allow for future unambiguous semantic interpretation of textual content. PDF/A writers should attempt to comply with this intent to the fullest extent possible.

All text strings encoded in Unicode whose language is not the default natural language for document or not the natural language defined by the innermost enclosing structure element or marked-content sequence shall indicate their language using the internal escape sequence described in Section 3.8.1 of the *PDF Reference*.

### 12.4 Alternate descriptions

To the fullest extent possible, all structure elements whose content does not have a natural predetermined textual analog, e.g., images, formulas, etc., should supply an alternate text description using the **Alt** entry in the structure element dictionary, as described in Section 9.8.2 of the *PDF Reference*.

NOTE Alternate descriptions provide textual descriptions that may aid in the proper interpretation of otherwise opaque non-textual content.

### 12.5 Replacement text

To the fullest extent possible, all textual structure elements that are represented in a non-standard manner, e.g., custom characters or inline graphics, should supply replacement text using the **ActualText** entry in the structure element dictionary, as described in Section 9.8.3 of the *PDF Reference*.

NOTE Replacement text provides textual equivalents that may aid in the proper interpretation of otherwise opaque, unusual representations of textual components.

### 12.6 Expansions of abbreviations and acronyms

To the fullest extent possible, all instances of abbreviations and acronyms in textual content should be placed in a marked-content sequence with a **Span** tag whose **E** property provides a textual expansion of the abbreviation or acronym, as described in Section 9.8.4 of the *PDF Reference*.

NOTE Abbreviation and acronym expansion provides textual equivalents that may aid in the proper interpretation of otherwise opaque nomenclature.

## 13 Forms

The intent of the requirements of this clause is to ensure that there is no ambiguity about the rendering of form fields.

A conforming PDF/A reader shall not use form fields to change the rendered representation of the page or the content of the document at any time. Form fields shall not perform actions of any type.

The **NeedAppearances** flag of the interactive form dictionary either shall not be present or shall be **false**.

Every form field shall have an appearance dictionary associated with the field's data.

A conforming PDF/A reader shall render the field according to the appearance dictionary without regard to the form data. A conforming PDF/A reader shall not implement any feature that would allow the document appearance to change.

EDITOR'S NOTE Additional text needs to be added here to define appropriate reader behaviour with respect to the display of a check mark or question mark next to a digital signature. This text will be developed through comments on this CD.

# Annex A
(informative)

# Security issues

EDITOR'S NOTE In draft 5 submitted by the US delegation at the New Orleans WG meeting this clause was a placeholder for a discussion of security issues: (lack of) encryption, signatures, URL links, etc. The text will be developed through comments on this CD.

# Annex B
(informative)

# Best practices for PDF/A

## B.1 Balanced page trees

All PDF/A files should contain balanced page trees. No **Pages** node should contain more then 12 entries.

NOTE The search speed for a balanced tree is O(log n). The search speed for a completely unbalanced tree can approach O(n).

## B.2 Use of non-XMP metadata

Use of non-XMP metadata at the file level is strongly discouraged as there is no assurance that such metadata can be preserved in accordance with this specification. In cases where non-XMP metadata is present, the preference is to convert it to XMP, embed it in the file, and document the conversation in the **xapMM:History** property. The **xapMM:History** property should also be used to indicate any non-XMP elements that have not been converted.

Failure to preserve metadata will cause problems in locating, interpreting, managing, and authenticating a file, which will in turn diminish or cancel archival value.

## B.3 Natural language identifiers

PDF/A writers should make the greatest effort possible to identify languages using ISO 639/ISO 3166 or IANA registered identifiers [1, 2, 4]. Private use identifiers should be used only if the language does not have a defined identifier within ISO 639/ISO 3166 or IANA registry. In the event that a language is truly unknown, the identifier **x-unknown** should be used.

## B.4 Recommended Software Requirements for Capturing or Converting Documents to PDF/A

For archival preservation purposes, this Best Practices statement provides recommended requirements for software that captures or converts documents to PDF/A to ensure that resulting PDF/A documents retain their quality and integrity as records.

ISO 15489-1:2001, Clause 7.1, specifies that “to support the continuing conduct of business, comply with the regulatory environment, and provide necessary accountability, organizations should create and maintain authentic, reliable and useable records, and protect the integrity of those records for as long as required” [3].

The regulatory environment for submitting documents to an organization’s archival institution may include requirements, standards and policies for electronic documents that stipulate document quality rules such as minimum image resolution, compression restrictions, or prohibited processes that either alter or dispose of approved data. For archival preservation purposes, the quality and integrity of documents created according to these legal and regulatory requirements, applicable standards, and organizational policy must be retained when they are captured or converted to PDF/A.

To meet this critical archival need, PDF/A capture or conversion processes shall replicate the exact content and quality of the source document within the PDF/A file. Following are examples of specific software requirements that accomplish this:

— PDF/A writers shall not use lossy compression, subsampling, downsampling, or any other process that either alters the content or degrades the quality of source data in the PDF/A document

— Software shall not substitute searchable text, based on optical character recognition, for the original scanned text within the bit-mapped image of documents that are scanned to PDF/A from paper or converted to PDF/A from image formats

# Bibliography

*[1]* ISO 639-1, *Codes for the representation of names of languages – Part 1: Alpha-2 code*

*[2]* ISO 3166-1, *Codes for the representation of names of countries and their subdivisions – Part 1: Country codes*

[3] ISO 15489-1, *Information and documentation – Records management – Part 1: General*

[4] *Language Tags*, IANA. Available from Internet <http://www.iana.org/assignments/language-tags>

[5] *The Unicode Standard*, Unicode Consortium (ISBN 0-201-61633-5). Available from Internet <http://www.unicode.org/versions/Unicode4.0.0/>