

ISO TC 171/SC 2 N

Date: 2003-07-3

ISO/WD 19005

ISO TC 171/SC 2/WG

Secretariat: ANSI

Document management — Long-term electronic preservation — 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 document 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: (20) Preparatory
Document language: E

H:\My Documents\PDF-A\ISO_19005_(E).doc STD Version 2.0

Copyright notice

This ISO document is a working draft or committee draft and is copyright-protected by ISO. While the reproduction of working drafts or committee drafts in any form for use by participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Requests for permission to reproduce this document for the purpose of selling it should be addressed as shown below or to ISO's member body in the country of the requester:

[Indicate the full address, telephone number, fax number, telex number, and electronic mail address, as appropriate, of the Copyright Manger of the ISO member body responsible for the secretariat of the TC or SC within the framework of which the working document has been prepared.]

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

Contents

Page

Foreword.....	v
Introduction.....	vi
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Conformance.....	2
4.1 General.....	2
4.2 Minimal conformance.....	2
4.3 Full conformance	3
4.4 Conforming PDF/A readers.....	3
5 File Format	3
5.1 General.....	3
5.2 Document catalog.....	3
5.3 Line Endings	3
5.4 White-Space	3
5.5 File Header	4
5.6 Trailer	4
5.7 Cross reference table	5
5.8 Document information dictionary	5
5.9 Extraneous binary data	5
5.10 String object.....	5
5.11 Name Objects	5
5.12 Stream Objects	5
5.13 Indirect objects	6
5.14 Linearized PDF	6
5.15 Filters	6
5.16 Streams.....	6
6 Graphics	6
6.1 General.....	6
6.2 Colorspaces	7
6.2.1 ICCBased colorspaces.....	7
6.2.2 Uncalibrated colorspaces	7
6.2.3 Named colorants in Separation and DeviceN colorspaces	7
6.3 Images.....	8
6.4 Form XObjects	8
6.5 Reference XObjects	8
6.6 PostScript XObjects	8
6.7 Extended Graphics State	8
6.8 Thumbnails.....	8
6.9 Rendering Intents	8
6.10 Content Streams	9
7 Fonts	9
7.1 General.....	9
7.2 Font types.....	9
7.3 Composite fonts	9
7.3.1 CIDFonts	9
7.3.2 Cmaps	9
7.4 Embedded font programs	9

7.4.1	Metadata	10
7.5	Font resources	10
7.6	Font subsets.....	10
7.7	Font metrics	10
7.8	Character encodings	10
7.9	Unicode character maps	11
8	Transparency	11
9	Hyperlinks.....	12
9.1	Retention of internal and simple links.....	12
9.2	Destinations and actions	12
9.3	Hyperlinks and metadata	12
9.4	Limits	12
10	Annotations	13
11	Metadata/XML.....	14
11.1	General.....	14
11.2	Properties	14
11.3	Normalization	15
11.4	XMP Header	15
11.5	File Identifiers.....	15
11.6	File Provenance Information	15
11.7	Extension Schemas	16
11.8	Validation.....	16
11.9	Font Metadata.....	16
12	Logical structure.....	17
12.1	Tagged PDF	17
12.1.1	Mark information dictionary	17
12.1.2	Artifacts	17
12.1.3	Hyphenation	17
12.1.4	Word breaks	17
12.1.5	Structure hierarchy.....	18
12.1.6	Structure types.....	18
12.2	Natural language specification.....	18
12.2.1	Text strings.....	18
12.3	Alternate descriptions.....	19
12.4	Replacement text	19
12.5	Expansions of abbreviations and acronyms	19
13	Forms	19
Annex A (informative) Security Issues.....		20
Annex B (informative) Ways to make PDF Files more efficient.....		21
13.1	Balanced pages trees.....	21
13.2	Use of Non-XMP Metadata	21
Bibliography.....		22

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 3.

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 International Standard may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 19005 was prepared by Technical Committee ISO/TC 171, *Document imaging 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.

Introduction

In the traditional documentation world, long term document storage and preservation has been accomplished by a combination of careful storage of paper records under controlled conditions and the use of optical reproduction of these materials in a variety of reduced size photographic formats such as microfilm, microfiche, etc. However, these methods do not address the growing number of documents that are created and used as electronic records in a wide variety of data formats.

There was an urgent need to archive electronic documents in a way that would ensure preservation of their contents over an extended period of time, and to further ensure that those documents would be able to be retrieved and rendered with a consistent and predictable result in the future.

Even more important was the need to define a both a file format and the behavior of retrieval devices (readers) that is compatible with both electronic documents and scanned images of traditional documents. This need has existed, and continues to exist, in a growing number of international government and industry segments, including legal systems, libraries, newspapers, regulated industries, and others.

The initial activity to define the business and technical requirements, and study possible solutions, was sponsored by a joint committee formed under AIIM International (the Association for Information and Image Management, International) and NPES (NPES The Association for Suppliers of Printing, Publishing and Converting Technologies).

The use of a restricted subset of the Adobe Portable Document Format (PDF), a publicly available published specification, similar to the work done in the printing and publish industry and known as PDF/X (ISO 15930), was identified as a solution path and the project became known as PDF/A.

That work has led to this International Standard, which addresses the use of PDF for the long term storage of multi-page documents that may contain a mixture of text, raster images and vector graphics. It addresses the features and requirements that must be supported by reading devices that will be used to retrieve and render the archived documents. A goal of this initial version is to emulate static paper with the added need to include electronic annotations, electronic signatures, marginalia, approvals, etc.

Because there is a significant need to index, inter-relate, and search such archived records, considerable effort was made to insure that the file format itself includes an appropriate level of information about the document as well as enabling the association of appropriate metadata with each file. However, it must be noted that such information requirements vary widely among the various anticipated user communities. Therefore, emphasis was placed on minimizing the required information but ensuring that the metadata capability was versatile enough to accommodate a wide variety of user needs.

This standard does not address the media used to record the electronic data or the associated requirements for its storage and/or maintenance. Such requirements are addressed by other ISO technical committees and International Standards.

It is anticipated that a variety of products will be developed based on PDF/A, 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.

An ongoing series of Application Notes [1] is maintained for the guidance of developers and users of the ISO PDF/A family of International Standards. They are available from TBD at <http://TBD>.

Intellectual Property Notice from Adobe Systems:

Adobe Systems Incorporated does not have a trademark with respect to the term "PDF" and therefore there are no releases that need to be obtained from Adobe for the use of the term "PDF/A". In the future if Adobe elected to

seek to trademark "PDF", permissions would be granted to the International Organization for Standardization to use "PDF/A" in conjunction with the PDF/A standard.

Document management — Long-term electronic preservation — Use of PDF (PDF/A)

1 Scope

This International Standard specifies the use of the Portable Document Format (PDF) for the long term preservation of black and white and color compound documents as electronic data. Compound documents may contain combinations of character, raster, vector and other data. This International Standard also specifies methods for rendering of an exact visual reproduction of the document as it appeared at the time it was submitted for preservation. It also enables the preservation and retrieval of appropriate metadata.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this International Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

Adobe CMap and CIDFont Files Specification, Technical Note #5014, Version 1.0, October 8, 1996, Adobe Systems Incorporated

Adobe Type 1 Font Format, 1990, Adobe Systems Incorporated (ISBN 0-201-57044-0)

The Compact Font Format Specification, Technical Note #5176, March 16, 2000, Adobe Systems Incorporated

PDF Reference: Adobe Portable Document Format, Version 1.4, third edition, 2001, Adobe Systems Incorporated (ISBN 0-201-75839-3)

Tags for the Identification of Languages, RFC 1766, March 1995

TrueType Reference Manual, September 17, 1999, Apple Computer, Inc.

Type 1 Font Format Supplement, Technical Specification #5015, May 15, 1994, Adobe Systems Incorporated

The Unicode Standard, Unicode Consortium

XMP: Extensible Metadata Platform, Version 1.5, September 14, 2001, Adobe Systems Incorporated

3 Terms and definitions

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

3.1

hyperlink

a relationship between two anchors, called the head and the tail of the hyperlink [DEXTER].

NOTE Anchors are identified by an anchor address: an absolute Uniform Resource Identifier (URI), optionally followed by a '#' and a sequence of characters called a fragment identifier.

EXAMPLE <http://www.w3.org/hypertext/WWW/TheProject.html>

<http://www.w3.org/hypertext/WWW/TheProject.html#z31>

3.2

anchor address

in an anchor address, the URI refers to a resource; it may be used in a variety of information retrieval protocols to obtain an entity that represents the resource, such as an HTML document.

NOTE The fragment identifier, if present, refers to some view on, or portion of the resource.

3.3

reader

a software application that is able to read and appropriately process files [ISO 15930-3]

3.4

electronic document

an electronic representation of a page-oriented aggregation of text and images that is susceptible to being reproduced on paper or optical microform without significant loss of its information content

3.5

information content

the shareable knowledge carried by an electronic document

3.6

semantic content

that portion of an electronic document's information content carrying interpretable meaning beyond the purely visual level, including structural constructs such as words, pages, and chapters, as well as descriptive and administrative metadata

3.7

long-term

a period of time extending across one or more generational changes in the technological environment underlying the representation or processing of electronic documents

3.8

preservation

the maintenance of the recoverability and usability of the information content of electronic documents

4 Conformance

4.1 General

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 standard.

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

4.2 Minimal conformance

A minimally conforming PDF/A document is a PDF document that meets all of the requirements specified in this standard to insure that its static rendered visual appearance is preservable over extended time periods. All minimally conforming PDF/A documents shall be conforming PDF 1.4 documents that meet additional specific requirements, which define specific constraints on the use of PDF 1.4 features with regard to general file format,

graphics, fonts, transparency, hyperlinks, annotations, and metadata. 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-section, and described fully in subsequent sections, 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

A fully conforming PDF/A document is a PDF document that meets all of the requirements specified in this standard to insure that the document logical structure and content text stream, in natural reading order, are preservable over extended time periods. All fully conforming PDF/A documents shall be minimally conforming PDF/A documents that meet additional requirements, which 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. In the subsequent sections of this standard (7.9 and 12), the requirements for full compliance above those needed for minimal compliance are clearly indicated.

The requirements for full conformance may place greater burdens on PDF/A creators, 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-section, and described fully in subsequent sections, 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 behavior specified in this standard. A conforming reader shall raise an error condition if applied against a non-conforming PDF document. A conforming reader shall clearly indicate its adherence to either the minimal or full PDF/A conformance levels, or both.

5 File Format

5.1 General

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

5.2 Document catalog

5.3 Line Endings

All End-of-Line (EOL) markers shall be the two character sequence of **CARRIAGE RETURN** (Unicode U+000D) followed immediately by **LINE FEED** (U+000A). No EOL marker shall immediately follow another EOL marker without at least one intervening character or comment.

NOTE Inconsistent line endings within a PDF file make; it is easy to misinterpret the start of data within a stream. This combination is the most frequently used EOL marker.

5.4 White-Space

The **NULL** (U+0000) and **FORM FEED** (U+0012) characters shall not be used as white-space characters. White-space characters are limited to **SPACE** (U+0020), **HORIZONTAL TABULATION** (U+0009), and the EOL marker.

No white-space character shall immediately follow another white-space character without at least one intervening non-white-space character or comment.

NOTE The requirements of this section only apply to white-space between PDF operators and objects; they do not apply to the contents of literal string or hexadecimal string objects.

5.5 File Header

No data shall precede the file header.

The file header line shall be immediately followed by a comment containing only the characters ¢ (U+00A2), £ (U+00A3), © (U+00A9), and µ (U+00B5), in that order.

NOTE A conforming PDF/A file will start with the following structure:

```
%PDF-1.4
%¢£©µ
```

which has the binary form (in hexadecimal):

```
255044462D312E340D0A25A2A3A9B50D0A
```

5.6 Trailer

The trailer of a PDF file enables an application reading the file to quickly find the cross reference table and certain special objects. The file trailer dictionary of a conforming PDF/A document shall contain all of the items listed in Table 1. The keyword **Encrypt** shall not be used in the document trailer dictionary.

Author comment The algorithm used to produce the ID is not fully specified in the *PDF Reference Manual*. The PDF/A standard should fully specify the ID algorithm.

Table 1 — Trailer values

Document Trailer	Key Type Value
Info Dictionary	Indirect reference to the document information dictionary.
ID Array	Array of two hex strings containing the file identifier.
Size Integer	Total number of entries in the cross reference table. Must be correct for the file to be valid.

No data shall following the trailer end-of-file marker.

NOTE A conforming PDF/A file will end with the following structure:

```
%%EOF
```

which has the binary form (in hexadecimal):

```
2525454F460D0A
```

5.7 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.

5.8 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 Section 11.

5.9 Extraneous binary data

No data shall exist between the end of one dictionary object and the beginning of the next dictionary object.

5.10 String object

A **literal string** is a sequence of characters, enclosed in parentheses. A **String object** can also be a sequence of hexadecimal data enclosed in <>.

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

To represent a new line character in the body of the **literal string** the escape sequence “\n” (U+005C, U+006E) shall be used.

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.

5.11 Name Objects

A **Name Object** is an internal symbol defined by a sequence of characters. The slash character introduces the character sequence used in all **Name Objects**.

A **Name object** shall only be composed of characters available in the **PDFDocEncoding** character set in the following value ranges:

- 48 to 57, 65 to 90, and 97 to 122 decimal
- 060 to 071, 101 to 132 and 141 to 172 octal
- **0** to **9**, **A** to **Z** and **a** to **z**.


Empty names shall not be permitted.

Hexadecimal codes shall not be permitted within names.

5.12 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 **Length** specified in the stream dictionary [the remainder of this sentence got garbled in the updating process after the last meeting. Does anyone remember what the decision was? Length must be correct? Length shall be ignored?]. 

5.13 Indirect objects

The object number, generation number, and **obj** keyword shall be located on a single line and the individual items 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.

All referenced objects shall exist within the document from which they are referenced.

NOTE The main body of the PDF file is composed of a collection of indirect objects. These indirect objects are labeled using two numbers followed by the **obj** keyword. The first number is the object number. The second number is the object generation number.

5.14 Linearized PDF

Conforming PDF/A files shall not be linearized.

NOTE Information on linearized PDF is ambiguous, contradictory and incomplete.

5.15 Filters

The LZW compression algorithm and the **ASCII85Decode** and **ASCIIHexDecode** filters shall not be permitted.

5.16 Streams

A **stream object** dictionary shall not contain any of the following keys:

- **F**
- **FFilter**
- **FDecodeParams**

6 Graphics

6.1 General

This section describes restrictions placed on both PDF files that comply with the specification (compliant files) and applications that render such files (compliant readers). It is intended to address graphical rendering issues that do not involve fonts and interactive elements. The topics addressed are Colorspaces, Images, Form XObjects, Reference XObjects, PostScript XObjects, Extended Graphics State, Thumbnails, Streams, Rendering Intents, and Content Streams.

6.2 Colorspaces

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

6.2.1 ICCBased colorspaces

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

A compliant reader shall render ICCBased colorspaces as specified by the ICC specification, and shall not use the **Alternate** colorspace specified in an ICC profile stream dictionary.

6.2.2 Uncalibrated colorspaces

A compliant file may use the DeviceGray colorspace, and at most one of the other two uncalibrated colorspaces defined in PDF: **DeviceRGB**, and **DeviceCMYK**. If an uncalibrated space is used in the file, then the file's **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 which contains an ICC profile defining a colorspace which has the same number of components as the device dependent colorspace that is used in the file, and which conforms to the requirements for ICCBased colorspaces used as source color specifications.

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

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

When rendering colors specified in a device dependent colorspace, a compliant reader shall use the profile specified in the **OutputIntents** array as the source colorspace.

6.2.3 Named colorants in Separation and DeviceN colorspaces

When rendering colorspaces based on **DeviceN** or **Separation** spaces, a compliant reader shall follow the following rules:

- If the named colorants in the space 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 space specified by the **OutputIntent** and the alternate space shall not be used.
- If the named colorants in the space 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 space specified by the **OutputIntent** and the alternate space shall not be used.
- If the only named colorant is **Gray**, and the document's **OutputIntent** is a Gray profile, the colorant shall be treated as the component of the space specified by the **OutputIntent**, and the alternate space shall not be used.
- In all other cases, the **Alternate** colorspace shall be used.

6.3 Images

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

An **Image** dictionary shall not contain 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 Section 6.8 below, Rendering Intents.

6.4 Form XObjects

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

6.5 Reference XObjects

A compliant file shall not contain any **Reference XObjects**.

6.6 PostScript XObjects

A compliant 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 shall ignore any instance of the **HT** key in an **ExtGState** dictionary.

[DISCUSSION ITEM: some members of the Rendering Group recommend banning the **HT** key; and others recommend it be both allowed and used. The rule stated here follows the example of PDF/X.]

Use of the **RI** key shall be as described in Section 6.8 below, Rendering Intents.

6.8 Thumbnails

[DISCUSSION ITEM: Some members of the Rendering group recommend making the inclusion of thumbnails mandatory.]

6.9 Rendering Intents

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

- **RelativeColorimetric**,
- **AbsoluteColorimetric**,
- **Perceptual**, or
- **Saturation**.

6.10 Content Streams

A Content Stream shall not contain any operators not documented in the PDF Reference Manual, even if such operators are bracketed by the **BX/EX** compatibility operators. A Content Stream should not contain other data not documented in the PDF Reference Manual.

7 Fonts

7.1 General

The intent of the requirements stated in this section is to insure 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 standard, 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 a conformant PDF/A writer to ensure the compliance of all fonts. This standard does not prescribe the manner in which compliance 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 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 define 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 compliant rendering processes shall use the embedded fonts, rather than other locally resident, substituted, or simulated fonts, for the visual reproduction of all text.

NOTE 1 Only fonts whose characters are referenced within a file need to be embedded in that file. Furthermore, as stated in Section 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 compliant 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 2 The standard 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.

7.4.1 Metadata

The requirements for font metadata are described in Section 11.9.

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 identify 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 compliant 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 subsection is applicable only for full compliance with this standard. For minimal compliance the requirements of this subsection 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, as described in Section 5.9 of the *PDF Reference*.


Fonts meeting the following conditions are exempted from this requirement:

- 1) do use the predefined encodings **MacRomanEncoding**, **MacExpertEncoding**, **WinAnsiEncoding**, or do use the predefined Identity-H or Identity-V CMaps; or
- 2) are 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*; or
- 3) are Type 0 fonts whose descendent **CIDFont** does use the Adobe-GB1, Adobe-CNS1, Adobe-Japan1, or Adobe-Korea1 character collections.

The CMap shall not map any character code to the Unicode private use area. The semantic properties of characters that do not have a Unicode equivalent should be described in an XMP metadata stream, as defined in Section 11.10.

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

8 Transparency

The use of transparency is prohibited which means the  wing.

An **ExtGState** dictionary shall not contain the **SMask** key.

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

An **ExtGstate** dictionary shall not contain the **ca** key with a value other than 1.0.

An **ExtGstate** dictionary shall not contain the **BM** key with a value other than **Normal** or **Compatible**.

An **Image XObject** dictionary shall not contain the **SMask** key.

[Wording from ISO 15930-3, above will be deleted after the committee meeting.](#)

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 the file. The visual effect of partially transparent graphics may 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/X compliant.

9 Hyperlinks

9.1 Retention of internal and simple links

Links within a PDF file and simple links to external PDF files should be retained; given that they are specified using a relative path.

GoTo	Allowed
GoToR	Allowed with conditions. Must use a partial path name relative to the current document.
Launch	Forbidden
URI	Forbidden
URI	Actionable URI forbidden
URI	Allowed, but compliant readers are not required to utilize it; thus it is advised that the complete text of the URI also be visible as text.
URI	Fully documented so that action could be taken.
URI	Converted to non static display
URI	Undecided
SubmitForm	Forbidden
JavaScript	Forbidden

9.2 Destinations and actions

Destinations and actions must conform to the restrictions placed on **GoToR**.

9.3 Hyperlinks and metadata

Adobe XMP, which permits document metadata to link to schemas that are publicly accessible via a persistent identifier, should be used. Such a schema is the Dublin Core Metadata Element Set at <http://dublincore.org/2002/08/23/dces#> (resolved from <http://purl.org/dc/elements/1.1/>).

9.4 Limits

Limits placed on links for PDF/A file content should not conflict with metadata linking to external sources.

Questions raised:

Will an external link to a URL exist in 500 years? If not, then no external links of any type should be allowed.

URL links of any type are also a very significant security threat; and, for that reason alone should be banned.

If you have an XML schema that needs to be included; it needs to be part of a standard that can be referenced by PDF/A or included directly in the PDF/A standard.

10 Annotations

This standard should guarantee that when a page from a PDF/A file is rendered on a screen by a PDF viewing application, the visual impression of the actual page is exactly what the author intended. This should include all annotation types that are contained within the file. Annotation types that invoke actions outside of the file being viewed should not be allowed. Note: *Notation of these types of annotations could become part of the record though. Possibly by a special character and a text notation in an external file that lists the page, location coordinates, external file information, and other annotation information.*

Table 2 — Annotation Type Description

Text	Text Annotation	OK
Link	Link Annotation	NO to action links; limited to specific destinations also. See:
FreeText (PDF 1.3)	Free text annotation	OK
Line (PDF 1.3)	Line annotation	OK
Square (PDF 1.3)	Square annotation	OK
Circle (PDF 1.3)	Circle annotation	OK
Highlight (PDF 1.3)	Highlight annotation	OK
Underline (PDF 1.3)	Underline annotation	OK
Squiggly (PDF 1.4)	Squiggly-underline annotation	OK
StrikeOut (PDF 1.3)	Strikeout annotation	OK
Stamp (PDF 1.3)	Rubber stamp annotation	OK
Ink (PDF 1.3)	Ink annotation	OK
Popup (PDF 1.3) Pop-up annotation	Popup annotation	NO ???
FileAttachment (PDF 1.3)	File attachment annotation	Not at this time. (Note: Handled differently...entry noted in place and file created externally?)
Sound (PDF 1.2)	Sound annotation	Not at this time. (Note: Handled differently...entry noted in place and file created externally?)
Movie (PDF 1.2)	Movie annotation	Not at this time. (Note: Handled differently...entry noted in place and file created externally?)
Widget (PDF 1.2)	Widget annotation	NO
PrinterMark (PDF 1.4)	Printer's mark annotation	OK
TrapNet (PDF 1.3)	Trap network annotation	OK

Table 3 — Action Type Description

GoTo	Go to a destination in the current document.	OK
GoToR	("Go-to remote") Go to a destination in another document.	With caveats, see M. Gavin.
Launch	Launch an application.	NO
Thread	Begin reading an article thread.	OK – subject to file specifications as in GoToR
URI	Resolve a uniform resource identifier.	NO
Sound (PDF 1.2)	Play a sound.	NO
Movie (PDF 1.2)	Play a movie.	NO
Hide (PDF 1.2)	Set an annotation's Hidden flag.	OK
Named (PDF 1.2)	Execute an action predefined by the viewer application.	OK for supported named actions (NextPage, PrevPage, FirstPage, LastPage)
SubmitForm (PDF 1.2)	Send data to a uniform resource locator.	NO
ResetForm (PDF 1.2)	Set fields to their default values.	NO
ImportData (PDF 1.2)	Import field values from a file.	NO
JavaScript (PDF 1.3)	Execute a JavaScript script.	NO

11 Metadata/XML

11.1 General

This section 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 for a compliant 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.

11.3 Normalization

Metadata shall be entered, saved, and retained in a normalized fashion to facilitate interchange and support consistent depiction of metadata by compliant PDF/A readers. All normalization shall be defined by schemas. The following normalizations are mandatory: *[Note: Adobe states that some of these normalizations need to be modified.]*

- 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 blank (U+0020) character.
- 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 the XMP – Extensible Metadata Platform.

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` *[either in PDF/A or XMP]* 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 specification.

[Author note: The PDF/A metadata group is discussing with Adobe how to best provide for an `xap:Identifier` property in either the PDF/A specification 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 to from Acrobat 1.3 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, Extension Schema Template; or 2) a machine readable format that conforms with the W3C RDF Schema Specification *[assuming Adobe agrees to support this]*. 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 compliant PDF/A reader shall parse and display all properly formed extension metadata and extension schemas.

Table 4 — Extension Schema Template

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

Author note: The PDF/A metadata group is discussing proposals to modify the XMP specification, including 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 *[if Adobe can support this]* whenever a file is saved or resaved.

[Author note: The PDF/A metadata group is discussing with Adobe 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.

[Author comment: There does not appear to be an appropriate place to add a similar metadata entry for Type 3 fonts, which do not have a font file stream. Metadata dictionaries are explicitly disallowed from font dictionaries, which would be the other logical place.]

12 Logical structure

This section is applicable only for full compliance with this standard. For minimal compliance the requirements of this section can be ignored.

The intent of the requirements stated in this section is to insure 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.

12.1 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.

12.1.1 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.

12.1.2 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*.

12.1.3 Hyphenation

Word hyphenation that is an incidental artifact of text layout shall be indicated by the use of a **SOFT HYPHEN** character (Unicode U+00AD). Use of the **HYPHEN-MINUS** character (U+002D) shall be restricted to instances of explicit hyphenation as the intent of the content creator.

12.1.4 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), **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.

12.1.5 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.

Compliant PDF/A generators 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.

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.

12.1.6 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.

12.2 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*.

The use of private use identifiers is strongly discouraged. Compliant PDF/A generators should make the greatest effort possible to identify languages using ISO 639/ISO 3166 or IANA registered identifiers. In the event that a language is truly unknown, the identifier **x-unknown** shall be used, in which instance no accompanying metadata is required.

NOTE 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. Compliant PDF/A generators should attempt to comply with this intent to the fullest extent possible.

12.2.1 Text strings

All text strings encoded in Unicode whose language is not the default natural language for document, or if 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*.

12.3 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.

12.4 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.

12.5 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

Form fields can exist in PDF/A documents.

Form fields must have valid appearances for the data they represent.

The "NeedAppearances" flag must not be present or be false.

The actual field data should be used strictly for retrieval as data and not for rendering.

All fonts must be embedded.

Disallow JavaScript.

Disallow submit and reset.

Disallow execute menu item, movie, open file.

All actions should be disallowed in forms.

A PDF/A viewer should not implement any feature that would allow the document appearance to change.

NOTE A PDF/A document should be tagged as a PDF/A document.

In short: There can be no disconnect between the form field data and its rendered representation. Form fields can not be used to change the rendered representation of the page or the content of the document at any time. Form fields can not perform actions of any type.

Annex A
(informative)

Security Issues

Annex B (informative)

Ways to make PDF Files more efficient

13.1 Balanced pages trees

A PDF/A file should contain a balanced pages tree.

An individual **Pages** node should not contain more than twelve (12) entries.

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

13.2 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.

Bibliography

Halasz, Frank and Schwartz, Mayer. "The Dexter Hypertext Reference Model," *Communications of the ACM* 37 (February 1994): 30-39.

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

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

ISO 639-1, *Codes for the representation of names of languages*

ISO 3166-1, *Codes for the representation of names of countries and their subdivisions*

ISO/IEC 8859-1, *Information technology — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1*