ISO/TC171/SC2 **N 226 E**

Date: 2003-04-30

ISO/TC171/SC2
Document Imaging Applications
Application Issues
SECRETARIAT: ANSI

TITLE	:	**NWI Ballot for Document management – Long-term electronic preservation – Use of PDF (PDF/A)**
SOURCE	:	**ISO/TC 171 SC2 Secretariat**
PROJECT	:	**NWI**
STATUS	:	**--**
REQUESTED ACTION	:	**Member countries are requested to complete and return the attached NWI ballot to the Secretariat by 15 August 2003.**
DISTRIBUTION	:	**P, 0 and L Members**

Address Reply to:
Secretariat - ISO TC171 SC2- Association for Information and Image Management International
1100 Wayne Ave, Suite 1100, Silver Spring, MD 20910-5603
Telephone: 240-494-2682; Facsimile: 301-587-2711; e-mail: bfanning@aiim.org

NEW WORK ITEM PROPOSAL

Date of presentation	Reference number (to be given by the Secretariat)
2003-04-30	
Proposer: ANSI	ISO/TC 171 / SC 2 **N 226**
Secretariat: ANSI	

A proposal for a new work item within the scope of an existing committee shall be submitted to the secretariat of that committee with a copy to the Central Secretariat and, in the case of a subcommittee, a copy to the secretariat of the parent technical committee. Proposals not within the scope of an existing committee shall be submitted to the secretariat of the ISO Technical Management Board.

The proposer of a new work item may be a member body of ISO, the secretariat itself, another technical committee or subcommittee, or organization in liaison, the Technical Management Board or one of the advisory groups, or the Secretary-General.

The proposal will be circulated to the P-members of the technical committee or subcommittee for voting, and to the O-members for information.

See overleaf for guidance on when to use this form.

IMPORTANT NOTE: Proposals without adequate justification risk rejection or referral to originator.
Guidelines for proposing and justifying a new work item are given overleaf.

Proposal (to be completed by the proposer)

Title of proposal (in the case of an amendment, revision or a new part of an existing document, show the reference number and current title)

English title: Document management - Long-term electronic preservation - Use of PDF (PDF/A)

French title (if available):

Scope of proposed project

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 creation from these data 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.

Concerns known patented items (see ISO/IEC Directives Part 1 for important guidance)

☐ Yes ☒ No If "Yes", provide full information as annex

Envisaged publication type (indicate one of the following, if possible)

☒ International Standard ☐ Technical Specification ☐ Publicly Available Specification ☐ Technical Report

Purpose and justification (attach a separate page as annex, if necessary)

The volume of electronic information is staggering. A recent study in the United States estimates that the world's total production of information amounts to about 250 megabytes - some 100,000 pages - for each man, woman, and child on earth. The number of electronic records required to be archived is also enormous. In the United States alone, the National Archives and Records Administration (NARA) estimates that there are 36.5 billion email messages each year alone in the U.S. federal government - each of which needs to be reviewed and considered for preservation. This standard will establish a set of guidelines for archiving and preserving digital documents in PDF format that will ensure preservation of their contents over an extended period of time.

Target date for availability (date by which publication is considered to be necessary) 12-2005

Relevant documents to be considered

Relationship of project to activities of other international bodies

This project will be the result of an ISO Joint Working Group consisting of representatives from ISO TC 171 SC2, ISO TC 130, ISO TC 46 SC 11 and ISO TC 42.

Liaison organizations	**Need for coordination with:**
ISO TC 130 ISO TC 46 SC11 ISO TC 42	☐ IEC ☐ CEN ☐ Other (please specify)

Preparatory work (at a minimum an outline should be included with the proposal)

☒ A draft is attached ☐ An outline is attached. It is possible to supply a draft by

The proposer or the proposer's organization is prepared to undertake the preparatory work required ☒ Yes ☐ No

Proposed Project Leader (name and address)	**Name and signature of the Proposer** (include contact information)
Mr. Stephen Abrams Harvard University Library 1280 Massachusetts Ave., Ste 404 Cambridge, MA 02138 USA Mr. Stephen Levenson Administrative Office U.S. Courts 1 Columbus Circle NE Washington, DC 20544 USA	Mr. Stephen Levenson Administrative Office U.S. Courts 1 Columbus Circle NE Washington, DC 20544 USA

Comments of the TC or SC Secretariat

Supplementary information relating to the proposal

☒ This proposal relates to a new ISO document;

☐ This proposal relates to the amendment/revision of an existing ISO document;

☐ This proposal relates to the adoption as an active project of an item currently registered as a Preliminary Work Item;

☐ This proposal relates to the re-establishment of a cancelled project as an active project.

Other:

Voting information

The ballot associated with this proposal comprises a vote on:

☒ Adoption of the proposal as a new project

☐ Adoption of the associated draft as a committee draft (CD)
(see ISO Form 5, question 3.3.1)

☐ Adoption of the associated draft for submission for the enquiry vote (DIS or equivalent)
(see ISO Form 5, question 3.3.2)

Other:

Annex(es) are included with this proposal (give details)

☒ Brief statement on patents

Date of circulation	Closing date for voting	Signature of the TC or SC Secretary
2003-04-30	2003-08-15	Betsy Fanning

Use this form to propose:

a) a new ISO document (including a new part to an existing document), or the amendment/revision of an existing ISO document;

b) the establishment as an active project of a preliminary work item, or the re-establishment of a cancelled project;

c) the change in the type of an existing document, e.g. conversion of a Technical Specification into an International Standard.

This form is not intended for use to propose an action following a systematic review - use ISO Form 21 for that purpose.

Proposals for correction (i.e. proposals for a Technical Corrigendum) should be submitted in writing directly to the secretariat concerned.

Guidelines on the completion of a proposal for a new work item
(see also the ISO/IEC Directives Part 1)

a) Title: Indicate the subject of the proposed new work item.

b) Scope: Give a clear indication of the coverage of the proposed new work item. Indicate, for example, if this is a proposal for a new document, or a proposed change (amendment/revision). It is often helpful to indicate what is not covered (exclusions).

c) Envisaged publication type: Details of the types of ISO deliverable available are given in the ISO/IEC Directives, Part 1 and/or the associated ISO Supplement.

d) Purpose and justification: Give details based on a critical study of the following elements wherever practicable. *Wherever possible reference should be made to information contained in the related TC Business Plan.*

1) The specific aims and reason for the standardization activity, with particular emphasis on the aspects of standardization to be covered, the problems it is expected to solve or the difficulties it is intended to overcome.

2) The main interests that might benefit from or be affected by the activity, such as industry, consumers, trade, governments, distributors.

3) Feasibility of the activity: Are there factors that could hinder the successful establishment or general application of the standard?

4) Timeliness of the standard to be produced: Is the technology reasonably stabilized? If not, how much time is likely to be available before advances in technology may render the proposed standard outdated? Is the proposed standard required as a basis for the future development of the technology in question?

5) Urgency of the activity, considering the needs of other fields or organizations. Indicate target date and, when a series of standards is proposed, suggest priorities.

6) The benefits to be gained by the implementation of the proposed standard; alternatively, the loss or disadvantage(s) if no standard is established within a reasonable time. Data such as product volume or value of trade should be included and quantified.

7) If the standardization activity is, or is likely to be, the subject of regulations or to require the harmonization of existing regulations, this should be indicated.

If a series of new work items is proposed having a common purpose and justification, a common proposal may be drafted including all elements to be clarified and enumerating the titles and scopes of each individual item.

e) Relevant documents: List any known relevant documents (such as standards and regulations), regardless of their source. When the proposer considers that an existing well-established document may be acceptable as a standard (with or without amendment), indicate this with appropriate justification and attach a copy to the proposal.

f) Cooperation and liaison: List relevant organizations or bodies with which cooperation and liaison should exist.

ISO TC /SC N

Date: 2003-04-30

ISO/WD XXXXX.3

ISO TC /SC /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

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 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 XXXXX was prepared by Technical Committee ISO/TC , , Subcommittee SC , .

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 creation from these data 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.

4 Conformance

4.1 General

The base-line criteria 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 or profiles: minimally conforming and fully conforming.

4.2 Minimal conformance profile

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 profile."

4.3 Full conformance profile

A fully conforming PDF/A document shall adhere to all of the criteria for minimal compliance, and in addition, meet 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, font metadata, and logical document structure based on the Tagged PDF framework. In the subsequent sections of this standard, 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 profile."

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 is intended to address the overall file format issues and the base elements which form the general makeup of a PDF/A file. These elements include; but, are not limited to the following: Line Endings, White-Space, File Header, Trailer, Cross Reference Table, Document Information Dictionary, Cos Strings, Cos Names, Cos Streams, and Indirect Objects.

5.2 Line Endings

Because of inconsistent line endings within a PDF file; it is easy to misinterpret the start of data within a stream; thus, rendering the data incorrectly.

Line ending characters should be limited to only using the carriage-return, line feed two character combination. This allows for consistent use throughout a file. This combination is also the most frequently used line ending.

The use of multiple, sequential line ending sequences is forbidden. There should be no blank lines within a PDF file.

5.3 White-Space

The **NULL** character is not a valid white-space character.

The **Form Feed** character is not a valid white-space character.

White-space characters are limited to the space character and the horizontal tab character.

White-spaces are to be kept to a single instance of the space or tab character. Runs of multiple white-space characters are forbidden.

5.4 File Header

The first two lines of the PDF file are the file header.

The first four characters of the file must be **%PDF**.

The binary data, in the second line, is required.

The binary data must be located in the second line of the PDF file immediately following the line containing the PDF Version number.

Comments are not allowed between the version number line and the binary data line.

Author comment The binary data should be explicitly defined by the PDF/A standard.

5.5 Trailer

The trailer of a PDF file enables an application reading the file to quickly find the cross reference table and certain special objects. A conforming PDF/A document must contain all of the items listed in Table 1. The keyword **Encrypt** is forbidden in the document trailer.

Author comment The Range Arrays have been added to allow search engines to more easily find metadata information inside a PDF/A file without requiring the development of a full scale PDF parser to access the metadata.

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.
InfoRange Array	Range Array (See 14.4.1.)
MetaData Dictionary	Indirect reference to the document matadata dictionary.
MetaDataRange Array	Range Array (See 14.4.1.)
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.

5.5.1 Range arrays

The arrays used by **InfoRange** and **MetaDataRange** contains integer doublets. The first number in the doublet represents the binary offset from the beginning of the file to the beginning of the relevant dictionary. The second number in the doublet is the length of the dictionary item from the beginning dictionary up to, but not including, the line termination characters following the **endobj** keyword.

- All offsets must be in the range 0 to 2147483647.
- All lengths must be in the range 1 to 2147483647.
- Offset[n+1] must be strictly greater than offset[n] + length[n].

The use of an array containing doublets allows referenced data to exist in multiple locations within the PDF/A file.

5.6 Cross reference table

The cross reference table needs to be correct.

A cross reference subsection header; starting object number, and range are to be separated by a single space character.

The **xref** keyword and the cross reference subsection header are to be separated by a single line termination.

Documents which contain malformed cross reference tables are to be rejected.

5.7 Document information dictionary

The Document Information Dictionary and several items located in the Document Information Dictionary are currently optional according to the *PDF Reference*. A conforming PDF/A document must contain all of the items listed in Table 2.

Table 2 — Document information dictionary values

Document Information Dictionary	Key Type Value
CreationDate date	The date and time the document was created.
ModDate date	The date and time the document was modified.
Creator text string	Name of software product used to create the PDF/A file.
Producer text string	
Changes array	Array of indirect references to Change dictionaries added chronologically to the array. New change dictionaries are added to the end of the array. (See Table 3.)
Context text string	

5.8 Change dictionary

In addition to the huge number of tools used to create PDF files; there are also a great many tools available with the ability to modify existing PDF files. The purpose of modification may be to stamp information on the PDF pages, compress images, modify page content, etc. Many of these modification tools leave no record of the tool used to make a modification. Yet, these very tools may, with their modification, cause a PDF file to no longer be valid and/or PDF/A compliant. Therefore, a specific mechanism needs to be available where information on the tools used to modify a specific PDF file can be stored.

A conforming PDF/A document, if modified, must contain a Change dictionary for each time the document is saved. The Change dictionary will contain the items listed in Table 3.

Table 3 — Change dictionary values

Change Dictionary	Key Type Value
ModDate date	The date and time the document was modified.
Product text string	Name of software product used to modify the PDF/A file.
Version text string	Version number of software product used to modify the PDF/A file.
Vendor text string	Name of the author or company who created the above software product.
Comments text string (Optional)	Any additional information the vendor wishes to add; i.e., the nature of the change made.

5.9 Binary garbage

No random data can exist following the **%%EOF** at the end of the file.

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

5.10 Cos strings

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

Cos Strings which are broken across lines must contain a line continuation character (“\”) immediately before the end of line character sequence.

To represent a new line character in the body of the **Cos String**; the escape sequence “\n” (backslash n) must be used.

Octal representations of single characters will contain exactly four (4) characters: the backslash character followed by three (3) digits in the range of zero to seven (0 to 7).

Hexadecimal strings must contain an even number of characters in the following ranges: (0 to 9) and either (A to F) or (a to f).

White-space in a hexadecimal string is forbidden.

5.11 Cos names

A **Cos Name** is an internal symbol defined by a sequence of characters. The slash character is the first character used in all **Cos Names**.

A **Cos Name** may 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.

A single slash character (“/”) is not a valid name.

Hexadecimal codes within names are forbidden.

5.12 Cos streams

A **Cos Stream** is a sequence of bytes delimited by the **stream** and **endstream** keywords. The basic problem is determining exactly where the stream starts and stops when extraneous white-space and line endings are used.

The **stream** keyword must be followed by a carriage-return, line feed two character combination.

The **endstream** keyword must be preceded by a carriage-return, line feed two character combination.

The **Length** specified in the stream dictionary must be correct or the file will be rejected as invalid.

5.13 Indirect objects

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.

The object number, generation number, and **obj** keyword are to be located on a single line and the individual items are each to be separated by a single white space character.

The object number must be preceded by a carriage-return, line feed two character combination.

The **obj** keyword must be followed by a carriage-return, line feed two character combination.

The **endobj** keyword must be preceded by a carriage-return, line feed two character combination.

The **endobj** keyword must be followed by a carriage-return, line feed two character combination.

If a referenced object does not exist within the document; the document is invalid.

5.14 Balanced pages trees

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

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

An individual **Pages** node is not to contain more then twelve (12) entries.

5.15 Linearization

Information on linearization is ambiguous, contradictory and incomplete.

Linearization is forbidden.

5.16 Binary file format

The PDF/A file format is strictly a binary file format; thus, a PDF/A file can not be represented as a purely ASCII file.

5.17 Filters

Filters are primarily used to compress data in streams and strings found in the PDF file.

Use of the LZW compression algorithm is forbidden.

PDF is a binary file format. The use of the **ASCII85Decode** filter is forbidden.

The use of the **ASCIIHexDecode** filter is forbidden.

5.18 Character encoding

How characters are encoded directly affects the meaning of the text the characters represent. In PDF it is possible to use alternate character mappings which display the characters correctly; but internally use non-standard character encodings which result in the meaning of the text being lost.

Custom character encodings, which alter the meaning of the encoded text, are forbidden.

Author comment: It is probably more appropriate to place this item in some other section of the PDF/A specification.

5.19 Streams

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

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

[NOTE: This restriction is not really specific to rendering but is included here to make sure it is not lost from the spec as a whole.]

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, Transparency, Thumbnails, Streams, Rendering Intents, and Content Streams.

6.2 Colorspaces

All colors must 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 Manual*, 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 the PDF Reference Manual, section 6.2.1.

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 the PDF Reference Manual, section 6.2.2.

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 or other data not documented in the PDF Reference Manual, 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 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 within a PDF/A file shall be embedded within that file.

All Type 0 **CIDFont** programs shall be in the compact font format. Type 1 font programs can 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 publicly identified as legally embeddable in a file for unlimited, universal viewing and printing 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 The requirement for embedded Type 1 font programs applies to the 14 standard Type 1 fonts. Note that only fonts whose characters are referenced with 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.

NOTE 2 Embedding the font programs allows any PDF/A compliant renderer to reproduce correctly all glyphs in the manner in which they were originally published without reference to possibly ephemeral external resources.

NOTE 3 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 4 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 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 Type 1 and TrueType fonts, a compliant PDF/A renderer 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

For all fonts that:

1) do not 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 not 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 not use the Adobe-GB1, Adobe-CNS1, Adobe-Japan1, or Adobe-Korea1 character collections,

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*.

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

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.

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

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 both unfiltered and unencrypted.

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 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.

11.8 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.

11.9 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.]

11.10 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.]

11.11 Character Property Metadata

[Note: Adobe has stated that XMP cannot support this implementation, so another approach is under development.] The CMap mechanism defined in Section 7.8 allows the recovery of semantic information about Unicode characters. If non-Unicode characters are referenced within a PDF/A file, their semantic properties should be provided in an XMP metadata stream. The properties for each non-Unicode equivalent character should be placed in a separate rdf:Description element, with the about attribute set to the font name, a single space character, and the four hexadecimal digit character code of the character. This character metadata shall be included in the XMP metadata stream defined by the **Metadata** entry of the document catalog

The namespace prefix for the PDF/A schema is pdfaChar.

Table 5 — Namespace prefix for PDF/A schema

Property	Valid Type	Description	Categor
pdfaChar:Name	Text	Descriptive name of character.	Internal
pdfaChar:Property	Choice (closed)	Unicode General Category property of character:	Internal

		"Lu"	Letter, uppercase	
		"Ll"	Letter, lowercase	
		"Lt"	Letter, titlecase	
		"Lm"	Letter, modifier	
		"Lo"	Letter, other	
		"Mn"	Mark, nonspacing	
		"Mc"	Mark, spacing combining	
		"Me"	Mark, enclosing	
		"Nd"	Number, decimal digit	
		"Nl"	Number, letter	
		"No"	Number, other	
		"Pc"	Punctuation, connector	
		"Pd"	Punctuation, dash	
		"Ps"	Punctuation, open	
		"Pe"	Punctuation, close	
		"Pi"	Punctuation, initial quote	
		"Pf"	Punctuation, final quote	
		"Po"	Punctuation, other	
		"Sm"	Symbol, math	
		"Sc"	Symbol, currency	
		"Sk"	Symbol, modifier	
		"So"	Symbol, other	
		"Zs"	Separator, space	
		"Zl"	Separator, line	
		"Zp"	Separator, paragraph	
		"C"	Other	
pdfaChar:Directionality	Choice (closed)	Directionality of character:		Internal
		"left-to-right"	if the display ordering of this character flows from left to right;	
		"right-to-left"	if the display ordering of this character flows from right to left.	
		"neutral"	if the display ordering of this character is inherited from previous characters.	
pdfaChar:NumericValue	Rational	Numeric value of this character. (Only applicable if the character has a numeric value.)		Internal
pdfaChar:Mirrored	Boolean	**true** if this character's glyph is mirrored horizontally in right-to-left text; otherwise **false**.		Internal

pdfaChar:Alphabetic	Boolean	**true** if this character is a primary unit of an alphabet or syllabary; otherwise **false**.	Internal
pdfaChar:Ideograhic	Boolean	**true** if this character is an ideograph; otherwise **false**.	Internal

NOTE 1 Although the semantic properties of non-Unicode characters are not required, their absence may seriously impact the ability of a PDF/A file to be preserved usefully over archival time-spans.

NOTE 2 The metadata properties are derived primarily from the Unicode normative and informative character properties.

[Author comment: Does ISO have a policy regarding the specification of XML namespaces for schemas defined in standards? Can we use the www.iso.org domain in a namespace URI? If not, can we use www.aiim.org or www.npes.org?]

11.12 Natural language private use identifier metadata

Section 12.2 specifies that all private use identifiers for natural languages should be accompanied by appropriate metadata. The description of each identifier should be placed in a separate **rdf:Description** element, with the **about** attribute set to the private use identifier. This metadata shall be included in the XMP metadata stream defined by the **Metadata** entry of the document catalog.

The namespace prefix for the PDF/A natural language private use identifier metadata schema is pdfLanguage.

Property	Valid Type	Description	Category
pdfLanguage:Description	Text	Description of the language specified by the private use identifier	Internal

12 Logical structure

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 conformant PDF/A file shall meet of all the requirements set forth for Tagged PDF in Section 9.7 of the *PDF Reference*. Any process that purports to determine PDF/A compliance shall report an error if any violation of a requirement imposed by the Tagged PDF conventions is discovered within a PDF/A file.

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**. Any process that purports to determine PDF/A compliance shall report an error condition if a PDF file's document catalog does not have a **MarkInfo** dictionary, or if that dictionary does not have a **Marked** entry, or if that entry's value is not **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) is 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.

Author Comment: For completeness, do we want to try to include deal with word breaks indicated by punctuation characters? For example, should the sequence of any of the following punctuation characters: EXCLAMATION MARK (U+0021), LEFT PARENTHESIS (U+0028), RIGHT PARENTHESIS (U+0029), COMMA (U+002C), FULL STOP (U+002E) COLON (U+003A), SEMICOLON (U+003B), LESS-THAN SIGN (U+003C), GREATER-THAN SIGN (U+003E), QUESTION MARK (U+003F), LEFT SQUARE BRACKET (U+005B), RIGHT SQUARE BRACKET (U+005D), LEFT CURLY BRACKET (U+007B), RIGHT CURLY BRACKET (U+007D), INVERTED EXCLAMATION MARK (U+00A1), INVERTED QUESTION MARK (U+00BF); immediately followed by one or more of the previously enumerated spacing characters be considered a single spacing character for the purposes of indicating word breaks.

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*. Any process that purports to determine PDF/A conformance shall report an error condition if a valid structure hierarchy is not discovered in a PDF file.

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. Any process that purports to determine PDF/A conformance shall report an error condition if a non-standard structure type not mapped directly or indirectly to a standard type is discovered in a PDF file.

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. Any process that purports to determine PDF/A conformance shall report an error condition if a default natural language is not defined for a PDF file.

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*. Any process that purports to determine PDF/A conformance shall report an error condition if a language identifier that does not comply with RFC 1766 is discovered within a PDF file.

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. If a private use identifier is used, it shall be accompanied at its first reference within a PDF document with descriptive metadata defining the language as fully as possible, using the metadata schema specified in Section 5.11. 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.

12.6 Compression of images and text

The PDF generator should use no compression that will inhibit the recreation of the original hard copy printed document. Some levels of image and text compression may not affect the on-screen viewing of a PDF, yet will render a printed copy that does not resemble the original. Recreating a document from an archived PDF may be inhibited where the resolution is reduced/downsampled too far or a compression scheme is used that is no longer supported and cannot be embedded in the PDF file.

12.6.1 Image compression

The process of creating the PDF shall not include image downsampling/resolution lower than the equivalent of 300 dpi. Graphics and images used in the PDF document may have a lower resolution, but the PDF generator shall not downsample or reduce image resolution below that of 300 dpi.

12.6.2 Text compression

No text compression shall be used to create the PDF.

12.7 Encryption and digital signatures

No encryption shall be used, including digital signatures, security features, access limits, or functional limitations. These functions prevent or limit access to the document, printing, changing the document, selecting text and graphics, or adding or changing notes and form fields.

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.

Bibliography

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*