Generation of PDF/X and PDF/A compliant PDF's with PDFTEX — pdfx.sty

C. V. Radhakrishnan, Hàn Thê Thành, Ross Moore and Peter Selinger

QUICK LINKS

1. Introduction

This package currently supports generation of PDF/X and PDF/A compliant documents using PDFTEX, in some of their variants. Support for additional standards, such as PDF/E, PDF/VT is also available; see the complete list in Section 2.1 below. By 'supports' here we mean that this package provides the correct means to declare that a document is supposed to conform with the stated PDF variant (PDF/X, PDF/A, PDF/E, PDF/VT, etc.) along with the version or level of conformance. This package also allows appropriate Metadata and Color Profile to be specified, according to what is required by the PDF variant.

Metadata elements, most of which must ultimately be written out as XML using UTF8 encoding, is provided via a file named `\jobname.xmpdata`, for the running LATEX job. Without such a file, providing some required information as well as a large range of optional data, a fully validating PDF file cannot be achieved. The PDF can be created, having the correct visual appearance on all pages, but it will not pass validation checks. Section 2.2 describes how this file should be constructed.

What this package *does not* do is to check for all the details of document structure and type of content that may be required (or restricted) within a PDF variant. For example, PDF/VT [11] requires well-structure parts, using form XObject sections tagged as '/DPart'. Similarly PDF/A-1a (and 2a and 3a) [3, 4, 5] require a fully 'Tagged PDF', including a detailed structure tagging which envelops the complete contents of the document. This is beyond the current version of PDFTEX, as commonly shipped. So while this package provides enough to meet the declaration, Metadata and font-handling aspects for these PDF/A variants, it is not sufficient to produce fully conforming PDFs. However, with extra PDFTEX-based software that *is* capable of producing 'Tagged PDF', this package can be used as part of the overall workflow to produce fully conforming documents.

1.1. PDF standards

PDF/X and PDF/A are umbrella terms used to denote several ISO standards [12, 13, 14, 16, 17, 3, 4, 5] that define different subsets of the PDF standard [1, 6]. The objective of PDF/X is to facilitate graphics exchange between document creator and printer and therefore, has all requirements related to printing. For instance, in PDF/X, all fonts need to be embedded and all images need to be CMYK or spot colors. PDF/X-2 and PDF/X-3 accept calibrated RGB and CIELAB colors along with all other specifications of PDF/X. Since 2005 some later variants of PDF/X have emerged, as extra effects (such as layering and transparency) have been supported within the PDF standard itself. The full range of versions and conformance supported in this package is discussed below in Section 2.1 'Options'.

PDF/A defines a profile for archiving PDF documents, which ensures the documents can be reproduced the exact same way in years to come. A key element to achieving this is that PDF/A documents shall be 100% self-contained. All the information needed to display the document in the same manner every time is embedded in the file. A PDF/A document is not permitted to be reliant on information from external sources. Other restrictions include avoidance of audio/video content, JavaScript and encryption. Mandatory inclusion of fonts, color profile and standards based metadata are absolutely essential for PDF/A. Later versions allow for use of image compression and file attachments.

PDF/E is an ISO standard [8] intended for documents used in engineering workflows. PDF/VT [11] allows for high-volume customised form printing, such as with utility bills. PDF/UA ('Universal Accessibility') is emerging as a standard [10, 9] supporting Assistive Technologies, incorporating web-accessibility guidelines (WCAG) for electronic documents. In future, there may emerge PDF/H for Health records and medical-related documents. Other applications can be envisaged. Declarations and Metadata are supported for the first two of these. The others are the subject of further work; revised versions of this package can be expected in later years.

2. Usage

2.1. Options

The package can be loaded with the command:

```
\usepackage[<option>]{pdfx}
```

where the options are as follows.

2.1.1. PDF/A options

PDF/A is an ISO standard [3, 4, 5] intended for long-term archiving of electronic documents. It therefore emphasizes self-containedness and reproducibility, as well as machine readable metadata. The PDF/A standard has three conformance levels “a”, “b”, and “u”. Level “a” is the strictest, but is not yet fully implemented by the pdfx package. Conformance level “u” has the same requirements as level “b”, but with the additional requirement that all text in the document must have a Unicode mapping. However, the pdfx package produces such Unicode mappings even in level “b” files. The standard also has three different versions 1, 2, and 3, which were standardized in 2005, 2011, and 2012, respectively. Earlier versions contain a subset of the features of later versions, so for maximum portability, it is preferable to use a lower-numbered version. There is no conformance level “u” in version 1 of the standard. For many typical uses of PDF/A, it is sufficient to use PDF/A-1b.

- a-1a: generate PDF/A-1a. Experimental, not fully implemented.
- a-1b: generate PDF/A-1b.
- a-2a: generate PDF/A-2a. Experimental, not fully implemented.
- a-2b: generate PDF/A-2b.
- a-2u: generate PDF/A-2u.
- a-3a: generate PDF/A-3a. Experimental, not fully implemented.
- a-3b: generate PDF/A-3b.
- a-3u: generate PDF/A-3u.

By ‘Experimental, not fully implemented’ here we mean primarily that the document structure, as required for ‘Tagged PDF’, is not handled by this package. Using other PDFTEX-based software that *is* capable of producing such complete tagging, conforming documents can indeed be produced.

2.1.2. PDF/E options

PDF/E is an ISO standard intended for documents used in engineering workflows. There is only one version of the PDF/E standard so far, and it is called PDF/E-1.

- e-1: generate PDF/E-1.

2.1.3. PDF/VT options

PDF/VT is an ISO standard intended as an exchange format for variable and transactional printing, and is an extension of the PDF/X-4 standard. The standard specifies three PDF/VT conformance levels. Level 1 is for single-file exchange, level 2 is for multi-file exchange, and level 2s is for streamed delivery. Currently, none of the PDF/VT conformance levels are fully implemented by the `pdfx` package.

- `vt-1`: generate PDF/VT-1. Experimental, not fully implemented.
- `vt-2`: generate PDF/VT-2. Experimental, not fully implemented.
- `vt-2s`: generate PDF/VT-2s. Experimental, not fully implemented.

By 'Experimental, not fully implemented' here we mean primarily that the structuring of a document into '/DPart' sections, as Form XObjects, is not handled by this package. This *is* possible with current PDFTEX software, but not yet in a way that lends itself easily to full automation, due to requirements of knowing the internal object number of certain internal PDF constructs. All the other aspects: PDFInfo declaration, Metadata and Color Profile, of the PDF/VT variants are correctly handled.

2.1.4. PDF/X options

PDF/X is an ISO standard intended for graphics interchange. It emphasizes printing-related requirements, such as embedded fonts and color profiles. The PDF/X standard has a large number of variants and conformance levels. The basic variants are known as X-1, X-1a, X-3, X-4, and X-5. (Note that a revised version of the X-2 standard was published in 2003, but withdrawn as an ISO standard in 2011, basically due to lack of interest in using it). The PDF/X-1a standard exists in revisions of 2001 and 2003, the PDF/X-3 standard exists in revisions of 2002 and 2003, and the PDF/X-4 and PDF/X-5 standards exist in revisions of 2008 and 2010. Moreover, some of these standards have a 'p' version, which permits the use of an externally supplied color profile (instead of an embedded one), and/or a 'g' version, which permits the use of external graphical content. Moreover, PDF/X-5 has an 'n' version, which extends PDF/X-4p by permitting additional color spaces other than Grayscale, RGB, and CMYK. For many typical uses of PDF/X, it is sufficient to use PDF/X-1a.

- `x-1`: generate PDF/X-1.
- `x-1a`: generate PDF/X-1a. Options `x-1a1` and `x-1a3` are also available to specify PDF/X-1a:2001 or PDF/X-1a:2003 explicitly.
- `x-3`: generate PDF/X-3. Options `x-302` and `x-303` are also available to specify PDF/X-3:2002 or PDF/X-3:2003 explicitly.
- `x-4`: generate PDF/X-4. Options `x-408` and `x-410` are also available to specify PDF/X-4:2008 or PDF/X-4:2010 explicitly.
- `x-4p`: generate PDF/X-4p. Options `x-4p08` and `x-4p10` are also available to specify PDF/X-4p:2008 or PDF/X-4p:2010 explicitly.
- `x-5g`: generate PDF/X-5g. Options `x-5g08` and `x-5g10` are also available to specify PDF/X-5g:2008 or PDF/X-5g:2010 explicitly.
- `x-5n`: generate PDF/X-5n. Options `x-5n08` and `x-5n10` are also available to specify PDF/X-5n:2008 or PDF/X-5n:2010 explicitly. Experimental, not fully implemented.
- `x-5pg`: generate PDF/X-5pg. Options `x-5pg08` and `x-5pg10` are also available to specify PDF/X-5pg:2008 or PDF/X-5pg:2010 explicitly.

2.1.5. Other options

These options are experimental and should not normally be used.

- useBOM: generate an explicit UTF-8 byte-order marker in the embedded XMP metadata, and make the XMP packet writable. Neither of these features are required by the PDF/A standard, but there exist some PDF/A validators (reportedly validatepdfa.com) that seem to require them. Note: the implementation of this feature is experimental and may break with future updates to the xmpincl package.
- noBOM: do not generate the optional byte-order marker. (default)
- pdf13: use PDF 1.3, overriding the version specified by the applicable standard. This may produce a non-standard-conforming PDF file.
- pdf14: use PDF 1.4, overriding the version specified by the applicable standard. This may produce a non-standard-conforming PDF file.
- pdf15: use PDF 1.5, overriding the version specified by the applicable standard. This may produce a non-standard-conforming PDF file.
- pdf16: use PDF 1.6, overriding the version specified by the applicable standard. This may produce a non-standard-conforming PDF file.
- pdf17: use PDF 1.7, overriding the version specified by the applicable standard. This may produce a non-standard-conforming PDF file.

2.2. Data file for metadata

As mentioned above, standards compliant PDF documents need metadata to be included. The pdfx package expects the metadata to be supplied in a special data file called \jobname.xmpdata. Here, \jobname is usually the basename of the document's main .tex file. For example, if your document source is in the file main.tex, then the metadata must be in a file called main.xmpdata. None of the individual metadata fields are mandatory, but for most documents, it makes sense to define at least the title and the author. Here is an example of a short .xmpdata file:

```
\Title{Baking through the ages}
\Author{A. Baker\sep C. Kneader}
\Keywords{cookies\sep muffins\sep cakes}
\Publisher{Baking International}
```

You should note that multiple authors and keywords have been separated by \sep. The \sep macro is only permitted in within the \Author, \Keywords, and \Publisher fields.

After processing, the local directory contains a file named such as pdfa.xmpi or pdfx.xmpi according to the PDF variant required. This file is the complete XMP Metadata packet. It can be checked for validity, using an online validator, such as at www.pdflib.com.

2.3. List of supported metadata fields

Here is a complete list of user-definable metadata fields currently supported, and their meanings. More may be added in the future. These commands can only be used in the .xmpdata file.

2.3.1. General information:

- `\Author`: the document's human author. Separate multiple authors with `\sep`.
- `\Title`: the document's title.
- `\Keywords`: list of keywords, separated with `\sep`.
- `\Subject`: the abstract.
- `\Publisher`: the publisher.

2.3.2. Copyright information:

- `\Copyright`: a copyright statement.
- `\CopyrightURL`: location of a web page describing the owner and/or rights statement for this document.
- `\Copyrighted`: "True" if the document is copyrighted, and "False" if it isn't. This is automatically set to "True" if either `\Copyright` or `\CopyrightURL` is specified, but can be overridden. For example, if the copyright statement is "Public Domain", this should be set to "False".

2.3.3. Publication information:

- `\PublicationType`: The type of publication. If defined, must be one of book, catalog, feed, journal, magazine, manual, newsletter, pamphlet. This is automatically set to "journal" if `\Journaltitle` is specified, but can be overridden.
- `\Journaltitle`: The title of the journal in which the document was published.
- `\Journalnumber`: The ISSN for the publication in which the document was published.
- `\Volume`: Journal volume.
- `\Issue`: Journal issue/number.
- `\Firstpage`: First page number of the published version of the document.
- `\Lastpage`: Last page number of the published version of the document.
- `\Doi` : Digital Object Identifier (DOI) for the document, without the leading "doi:".
- `\CoverDisplayDate`: Date on the cover of the journal issue, as a human-readable text string.
- `\CoverDate`: Date on the cover of the journal issue, in a format suitable for storing in a database field with a "date" data type.

2.4. Symbols permitted in metadata

Within the metadata, all printable ASCII characters except `\`, `{` and `}` , and `%` represent themselves. Also, all printable Unicode characters from the basic multilingual plane (i.e., up to code point U+FFFF) can be used directly with the UTF-8 encoding. (Please note: encodings other than UTF-8 are not currently supported in the metadata). Consecutive whitespace characters are combined into a single space. Whitespace after a macro such as `\copyright`, `\backslash`, or `\sep` is ignored. Blank lines are not permitted. Moreover, the following markup can be used:

- "`\ `": a literal space (for example after a macro)
- `\%`: a literal `%`
- `\{`: a literal `{`
- `\}`: a literal `}`

- `\backslash`: a literal backslash "\"
- `\copyright`: the © copyright symbol

The macro `\sep` is only permitted within `\Author`, `\Keywords`, and `\Publisher`. It's intention is to separate multiple authors, keywords, etc. However for validation purposes, multiple authors and keywords must not really be separated. The package takes care of this, even when `\sep` is used.

Other TEX macros actually can be used, provided the author is very careful and not ask for too complicated TEX or LATEX expansions into internal commands or non-character primitives; basically just accents, macros for Latin-based specil characters, and simple textual replacements, perhaps with a parameter. A special macro `\pdfxEnableCommands{...}` is provided to help resolve any difficulties that may arise.

Here is an example of the use of `\pdfxEnableCommands`, which occurs with the name of one of our authors (Hàn Thế Thanh) due to the doubly-accented letter ế. It is usual to define a macro such as: `\def\thanh{H\`an Th\'{\^e} Thanh}`.

In previous versions of the `pdfx` package, use of such a macro within the `.xmpdata` file, in the `Copyright` information say, could result in the accent macros expanding into internal primitives, such as

```
H\unhbox \voidb@x \bgroup \let \unhbox \voidb@x \setbox \@tempboxa ...
```

going on for many lines. This clearly has no place within the XMP Metadata. To get around this, one could try using simplified macro definitions

```
\pdfxEnableCommands{
    \def\‘#1{#1^^cc^80}\def\’#1{#1^^cc^81}\def\^#1{#1^^cc^82}}
```

where the `^^cc^80`, `^^cc^81`, `^^cc^82` cause TEX to generate the correct UTF8 bytes for 'combining accent' characters.

This works fine for metadata fields that appear just in the XMP packet. However, it is not sufficient for the PDF `/Author` key, which must exactly match with the `dc:creator` Metadata element. What is needed instead is

```
\pdfxEnableCommands{
     \def\thanh{H^^c3^^a0n Th\eee Thanh}\def\eee{^^c3^^aa^^cc^^81 }}
```

or the above with 'à' typed directly as UTF8 instead of `^^c3^^a0` and 'ê' in UTF8 for `^^c3^^aa`. The reason for this is due to the `\pdfstringdef` command, which constructs the accented latin letters as single combined characters à and ê, without resorting to combining accents, wherever possible. If the Metadata does not have the same, irrespective of Unicode normalisation, then validation fails.

With this latest version of the `pdfx` package, such difficulties have been overcome, at least for characters used in Western European, Latin-based languages. The input encoding used when reading the `.xmpdata` file now includes interpretations of TEX's usual accent commands to produce the required UTF-8 byte sequences. Work is ongoing to extend this input encoding to include macro definitions for more special characters (e.g., extended Latin, Cyrillic, Greek, etc.). A significant portion of the Unicode Basic Plane characters can be covered this way. Modules could even be provided for CJK character sets and mathematical symbols, etc. However, this can become memory intensive, so significant testing will be required before this becomes a standard part of the `pdfx` package.

2.5. Color profiles

Most standards compliant PDF documents require a *color profile* to be embedded within the file. In a nutshell, such a profile determines precisely how the colors used in the document will be rendered when printed to a physical medium. This can be used to ensure that the document will look exactly the same, even when it is printed on different printers, with different paper types, etc. The inclusion of a color profile is necessary to make the document completely self-contained.

Since most LATEX users are not graphics professionals and are not particularly picky about colors, the pdfx package includes default profiles that will be included when nothing else is specified. Therefore, the average user doesn't have to do anything special about color.

For users who have a specific color profile they wish to use, it is possible to do so by including a \setRGBcolorprofile or \setCMYKcolorprofile command in the .xmpdata file. Note that PDF/A and PDF/E require an RGB color profile, and PDF/X and PDF/VT require a CMYK color profile. Use the following commands to specify an RGB or CMYK color profile, respectively:

```
\setRGBcolorprofile{⟨filename⟩}{⟨identifier⟩}{⟨info string⟩}{⟨registry URL⟩}
\setCMYKcolorprofile{⟨filename⟩}{⟨output intent⟩}{⟨identifier⟩}{⟨registry URL⟩}
```

Within the arguments of these macros, the characters <, >, &, ^, _, #, $, and ~ can be used as themselves, but % must be escaped as \%. The defaults are:

```
\setRGBcolorprofile{sRGB_IEC61966-2-1_black_scaled.icc}
                   {sRGB_IEC61966-2-1_black_scaled}
                   {sRGB IEC61966 v2.1 with black scaling}
                   {http://www.color.org}

\setCMYKcolorprofile{coated_FOGRA39L_argl.icc}
                    {Coated FOGRA39}
                    {FOGRA39 (ISO Coated v2 300\% (ECI))}
                    {http://www.argyllcms.com/}
```

Some color profile files may be obtained from the International Color Consortium. Please take a look at http://www.color.org/iccprofile.xalter.

Alternatively, color profiles are shipped with many Adobe software applications; these are then available for use also with non-Adobe software. Now the pdfx package includes coding to streamline inclusion of these profiles in PDF documents, or to specify them as 'external' profiles, with PDF/X-4p and PDF/X-5pg variants. Two files AdobeColorProfiles.tex and AdobeExternalProfiles.tex are distributed with the pdfx package. The latter is for use with PDF/X-4p and PDF/X-5pg, which do not require color profiles to be embedded, while the former can be used with other PDF/X variants. Both define commands to use Color Profiles as follows.

\FOGRAXXXIX	Coated FOGRA39 (ISO 12647-2:2004)
\SWOPCGATSI	U.S. Web Coated (SWOP) v2
\JapanColorMMICoated	Japan Color 2001 Coated
\JapanColorMMIUncoated	Japan Color 2001 Uncoated
\JapanColorMMIINewspaper	Japan Color 2002 Newspaper
\JapanWebCoatedAd	Japan Web Coated (Ad)
\CoatedGRACoL	Coated GRACoL 2006 (ISO 12647-2:2004)
\SNAPCGATSII	CGATS TR 002
\SWOPCGATSIII	CGATS TR 003
\SWOPCGATSV	CGATS TR 005
\ISOWebCoated	Web Coated FOGRA28 (ISO 12647-2:2004)
\ISOCoatedECI	ISO Coated v2 (ECI)
\CoatedFOGRA	Coated FOGRA27 (ISO 12647-2:2004)
\WebCoatedFOGRA	Web Coated FOGRA28 (ISO 12647-2:2004)
\UncoatedFOGRA	Uncoated FOGRA29 (ISO 12647-2:2004)
\IFRAXXVI	ISOnewspaper26v4 ISO/DIS 12647-3:2004
\IFRAXXX	ISOnewspaper30v4 ISO/DIS 12647-3:2004

As of the time of writing, only the first six of these result in PDFs which can validate with external profiles (i.e., for PDF/X-4p and PDF/X-5pg) using current versions of Adobe Acrobat Pro software. It is unclear whether the others (incl. \IFRAXXVI and \IFRAXXX) fail due to incorrect data or problems in the validation software. All but those last two can be used for valid embedded profiles, providing the corresponding files can be found. The following macro is used to set the (absolute or relative) path, on the local operating system, to the location of color profile files.

```
\pdfxSetColorProfileDir{⟨path to Adobe color profiles⟩}
```

2.6. Notes on the internal representation of metadata

Within the PDF file, metadata is deposited in two places: some data goes into the native PDF /Info dictionary, and some data goes into an XMP packet stored separately within the file. XMP is Adobe's Extensible Metadata Platform, and is an XML-based format. See Adobe XMP Development Center for more exhaustive information about XMP. An XMP Toolkit SDK which supports the GNU/Linux, Macintosh and Windows operating systems is also provided under modified BSD licence.

Some of the metadata, such as the author, title, and keywords, are stored **both** in the XMP packet and in the /Info dictionary. For the resulting file to be standard compliant, the two copies of the data must be identical. All of this is taken care of automatically by the pdfx package.

In principle, users can resort to alternate ways to create an XMP file for inclusion in PDF. In this case, users should create a file pdfa.xmp or pdfx.xmp (etc., depending on the PDF flavor) containing the pre-defined data. However, this is an error prone process and is not recommended for most users. If there is a particular field of metadata that you need and that is not currently supported, please contact the authors.

pdfx makes use of the xmpincl package to include xmp data into the PDF. The documentation of xmpincl package may help interested users to understand the process of xmp data inclusion.

2.7. Tutorials and technical notes

A tutorial with step-by-step instructions for generating PDF/A files can be found at: http://www.mathstat.dal.ca/~selinger/pdfa/.

Some technical notes about production problems the authors have encountered while generating PDF/A compliant documents are available here: http://support.river-valley.com/wiki/index.php?title=Generating_PDF/A_compliant_PDFs_from_pdftex.

3. Installing

The file pdfx.dtx is a composite document of program code and documentation in LaTeX format in the tradition of *literate programming*. To get the documentation that you are reading now, run (PDF)LaTeX on the file pdfx.dtx. Or better, use the included Makefile, which will also regenerate the index.

To install the package, first extract the program code, i.e., the file pdfx.sty, by running LaTeX or TeX on the file pdfx.ins. Create a directory named pdfx under $TEXMF/tex/latex and copy the files pdfx.sty, 8bit.def, glyphtounicode-cmr.tex, as well as the *.icc and *.xmp files, to it. Then update TeX's file database using the appropriate command for your distribution and operating system (such as texhash or mktexlsr, or similar).

3.1. Limitations and dependencies

pdfx.sty works with PDFTeX and also LuaTeX. It further depends on the following other packages:

1. xmpincl for insertion of metadata into PDF.
2. hyperref for hyperlinking, bookmarks, etc.
3. glyphtounicode.tex maps glyph names to corresponding Unicode.

3.2. Files included

The following files are included in the archive:

3.2.1. Package files

- pdfx.sty — main package file generated from pdfx.dtx.
- pdfa.xmp — specimen xmp template for PDF/A.
- pdfe.xmp — specimen xmp template for PDF/E.
- pdfvt.xmp — specimen xmp template for PDF/VT.
- pdfx.xmp — specimen xmp template for PDF/X.
- 8bit.def — custom input encoding.
- l8uenc.def — input encoding macro declarations.
- glyphtounicode-cmr.tex — maps glyph names to corresponding Unicode for Computer Modern and other TeX-specific fonts.
- coated_FOGRA39L_argl.icc — a freely distributable CMYK color profile.
- sRGB_IEC61966-2-1_black_scaled.icc — a freely distributable RGB color profile.
- ICC_LICENSE.txt — the license for the color profiles.
- AdobeColorProfiles.tex — macros for inclusion of Adobe-supplied color profiles.
- AdobeExternalProfiles.tex — macros for use of external color profiles.

3.2.2. Documentation

- README — Readme file.
- manifest.txt — file list.
- sample.tex, sample.xmpdata — a sample file with sample metadata.
- small2e-pdfx.tex, small2e-pdfx.xmpdata — another sample file with sample metadata.

3.2.3. Sources

- src/pdfx.dtx — composite package and documentation.
- src/pdfx.ins — installer batch file.
- src/rvdtx.sty — used by pdfx.dtx.
- src/Makefile — a Makefile for building the documentation.

3.3. Miscellaneous information

The package is released under the LaTeX Project Public Licence. Bug reports, suggestions, feature requests, etc., may be sent to the original authors at cvr@river-valley.org and/or thanh@river-valley.org, or to the more recent contributors at ross.moore@mq.edu.au and/or selinger@mathstat.dal.ca.

4. Bibliography

References

[1] Adobe Systems Inc.; PDF Reference 1.7, November 2006. Also available as [6]. http://www.adobe.com/devnet/pdf/pdf_reference.html.

[2] Dublin Core Metadata Element Set, Version 1.1, October 2010 http://dublincore.org/documents/dces/

[3] ISO 19005-1:2005; Document Management — Electronic document file format for long term preservation — Part 1: Use of PDF 1.4 (PDF/A-1); Technical Committee ISO/TC 171/SC 2 (Sept. 2005). Revisions via Corrigenda: ISO 19005-1:2005/Cor 1:2007 (March 2007); ISO 19005-1:2005/Cor 2:2011 (Dec. 2011). http://www.iso.org/iso/catalogue_detail?csnumber=38920.

[4] ISO 19005-2:2011; Document Management — Electronic document file format for long term preservation — Part 2: Use of ISO 32000-1 (PDF/A-2); Technical Committee ISO/TC 171/SC 2 (June 2011). http://www.iso.org/iso/catalogue_detail?csnumber=50655.

[5] ISO 19005-3:2012; Document Management — Electronic document file format for long term preservation — Part 3: Use of ISO 32000-1 with support for embedded files (PDF/A-3); Technical Committee ISO/TC 171/SC 2 (October 2012). http://www.iso.org/iso/catalogue_detail?csnumber=57229.

[6] ISO 32000-1:2008; Document management — Portable document format (PDF 1.7); Technical Committee ISO/TC 171/SC 2 (July 2008). Also available as [1]. http://www.iso.org/iso/catalogue_detail?csnumber=51502.

[7] ISO 32000-2-20140220; Document management — Portable document format — Part 2: PDF 2.0; Technical Committee ISO/TC 171/SC 2, in draft form (Feb. 2014).

[8] ISO 24517-1:2008; Document Management — Engineering document format using PDF — Part 1: Use of PDF 1.6 (PDF/E-1); Technical Committee ISO/TC 171/SC 2 (May 2008). http://www.iso.org/iso/catalogue_detail?csnumber=42274.

[9] ISO 14289-1:2012; Document management applications — Electronic document file format enhancement for accessibility — Part 1: Use of ISO 32000-1 (PDF/UA-1); Technical Committee ISO/TC 171/SC 2 (July 2012). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=54564.
Revised as ISO 14289-1:2014 (December 2014): http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=64599.

[10] PDF/UA Technical Implementation Guide: Understanding ISO 14289-1 (PDF/UA-1). AIIM Global Community of Information Professionals. http://www.aiim.org/Research-and-Publications/standards/committees/PDFUA/Technical-Implementation-Guide.

[11] ISO 16612-2:2010; Graphic technology — Variable data exchange — Part 2: Using PDF/X-4 and PDF/X-5 (PDF/VT-1 and PDF/VT-2). Technical Committee ISO/TC 130 (December 2005). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=38013.

[12] ISO 15930-1:2001; Graphic technology — Prepress digital data exchange — Use of PDF — Part 1: Complete exchange using CMYK data (PDF/X-1 and PDF/X-1a). Technical Committee ISO/TC 130 (December 2001). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=29061.

[13] ISO 15930-3:2002; Graphic technology — Prepress digital data exchange — Use of PDF — Part 3: Complete exchange suitable for colour-managed workflows (PDF/X-3). Technical Committee ISO/TC 130 (September 2002). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=34941.

[14] ISO 15930-4:2003; Graphic technology — Prepress digital data exchange — Use of PDF — Part 4: Complete exchange of CMYK and spot colour printing data using PDF 1.4 (PDF/X-1a). Technical Committee ISO/TC 130 (December 2003). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=39938.

[15] ISO 15930-6:2003; Graphic technology — Prepress digital data exchange — Use of PDF — Part 6: Complete exchange of printing data suitable for colour-managed workflows using PDF 1.4 (PDF/X-3). Technical Committee ISO/TC 130 (December 2003). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=39940.

[16] ISO 15930-7:2010; Graphic technology — Prepress digital data exchange — Use of PDF — Part 7: Complete exchange of printing data (PDF/X-4) and partial exchange of printing data with external profile reference (PDF/X-4p) using PDF 1.6. Technical Committee ISO/TC 130 (July 2010). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=55843.

[17] ISO 15930-8:2010; Graphic technology — Prepress digital data exchange — Use of PDF — Part 8: Partial exchange of printing data using PDF 1.6 (PDF/X-5). Technical Committee ISO/TC 130 (July 2010). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=55844. Revision via Corrigendum: ISO 15930-8:2010/Cor 1:2011 (August 2011); http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=60210.

[18] ISO 16684-1:2012; Graphic technology — Extensible metadata platform (XMP) specification — Part 1: Data model, serialization and core properties. Technical Committee ISO/TC 130 (February 2012). http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=57421.