The Printer Working Group

Standard for PDF Image-Streamable Format – “PDF/is”

(Formerly “PDFax”)

Proposed Standard - Working Draft 510n.y-P0.3

19 November 2002

The Printer Working Group Standard for PDF Image-Streamable Format (PDF/is) Proposed Standard - Working Draft 510n.y-P0.3

Abstract: This standard specifies a subset of PDF (Portable Document Format) 1.4 known as the PDF Image-Streamable Format (PDF/is) by formally defining a series of PDF/is "profiles" distinguished primarily by the method of image compression employed and color space used.

In summary PDF/is is an image document format intended for use by, but not limited to, the IPPFAX protocol, which is used to provide a synchronous, reliable exchange of image Documents between Senders and Receivers. PDF/is makes reference to the PDF 1.4 Reference [pdf], which describes the PDF representation of image data specified by the ITU-T Recommendations for black-and-white facsimile (see [T.4], [T.6]), the ISO/IEC Specifications for Digital Compression and Coding of Continuous-Tone Still Images (see [jpeg]), and Lossy/Lossless Coding of Bi-Level Images (see [jbig2]), and the general purpose Flate compression methods (see [RFC1950] and [RFC1951]).

This document is available electronically at:

ftp://pwg.org/pub/pwg/QUALDOCS/pwg-ifx-pdfis-P03-021119.pdf, .doc

A version showing the changes from the previous version is available at:

ftp://pwg.org/pub/pwg/QUALDOCS/pwg-ifx-pdfis-P03-021119-rev.pdf

The latest version of this specification is available at:

ftp://pwg.org/pub/pwg/QUALDOCS/pwg-ifx-pdfis-latest.pdf, .doc

About the IEEE-ISTO

The IEEE-ISTO is a not-for-profit corporation offering industry groups an innovative and flexible operational forum and support services. The IEEE-ISTO provides a forum not only to develop standards, but also to facilitate activities that support the implementation and acceptance of standards in the marketplace. The organization is affiliated with the IEEE (http://www.ieee.org/) and the IEEE Standards Association (http://standards.ieee.org/).

For additional information regarding the IEEE-ISTO and its industry programs visit http://www.ieee-isto.org.

About the IEEE-ISTO PWG

The Printer Working Group (or PWG) is a Program of the IEEE Industry Standards and Technology Organization (ISTO) with member organizations including printer manufacturers, print server developers, operating system providers, network operating systems providers, network connectivity vendors, and print management application developers. The group is chartered to make printers and the applications and operating systems supporting them work together better. All references to the PWG in this document implicitly mean “The Printer Working Group, a Program of the IEEE ISTO.” In order to meet this objective, the PWG will document the results of their work as open standards that define print related protocols, interfaces, procedures and conventions. Printer manufacturers and vendors of printer related software will benefit from the interoperability provided by voluntary conformance to these standards.

In general, a PWG standard is a specification that is stable, well understood, and is technically competent, has multiple, independent and interoperable implementations with substantial operational experience, and enjoys significant public support.

For additional information regarding the Printer Working Group visit: http://www.pwg.org

Contact information:

IFX Web Page: http://www.pwg.org/qualdocs
IFX Mailing List: ifx@pwg.org

To subscribe to the ipp mailing list, send the following email:
1) send it to majordomo@pwg.org
2) leave the subject line blank
3) put the following two lines in the message body:
subscribe ifx
end

Implementers of this specification are encouraged to join the IFX Mailing List in order to participate in any discussions of clarifications or review of registration proposals for additional names. Requests for additional media names, for inclusion in this specification, should be sent to the IFX Mailing list for consideration.

Contents

Table of Tables

1 Introduction

In summary, PDF/is is a raster image data format intended for use by, but not limited to, the IPPFAX protocol. IPPFAX is used to provide a synchronous, reliable exchange of image Documents between Senders and Receivers. PDF/is makes reference to the PDF 1.4 specification [pdf], which describes the PDF (Portable Document Format) representation of image data specified by the ITU-T Recommendations for black-and-white facsimile (see [T.4], [T.6]), the ISO/IEC Specifications for Digital Compression and Coding of Continuous-Tone Still Images (see [jpeg]), and Lossy/Lossless Coding of Bi-Level Images (see [jbig2]), and the general purpose Flate compression methods (see [RFC1950] and [RFC1951]).

PDF/is is an image-only, streamable, subset specification of PDF 1.4 [pdf] and, as such, follows all of the specification requirements of PDF.

As a streamable version of PDF, it is not required that a Renderer of a PDF/is document be able to randomly access the PDF. The format has been adopted in such a way as to allow a Renderer the ability to read the PDF/is document from the beginning to end without the necessity to cache more data than is necessary to print the current page with some exceptions, as noted.

If a Document adhering to this specification is not encrypted (does not Implement Profiles 'STD-ENC' nor 'PPK-ENC') it will Implement a conforming subset of the "PDF/X-3" specification (See [pdf-x3]) for use in digital prepress data exchange.

2 Terminology

This section defines terminology used throughout this document.

2.1 Conformance Terminology

Capitalized terms, such as **MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, MAY, NEED NOT, OPTIONAL,** and **PROHIBITED**, have special meaning relating to conformance as defined in RFC 2119 [RFC2119] and [RFC2911] section 12.1. If an implementation supports the extension defined in this document, then these terms apply; otherwise, they do not. These terms define conformance to *this document (and [RFC2911]) only*; they do not affect conformance to other documents, unless explicitly stated otherwise. To be more specific:

REQUIRED (REQ) - an adjective used to indicate that a conforming PDF/is Creator or Renderer's implementation MUST support the indicated operation, object, attribute, or attribute value. See [RFC2911] "Appendix A - Terminology for a definition of "support".

RECOMMENDED (REC) - an adjective used to indicate that a conforming PDF/is Creator or Renderer's implementation SHOULD support the indicated operation, object, attribute, or attribute value.

OPTIONAL (OPT) - an adjective used to indicate that a conforming PDF/is Creator or Renderer's implementation MAY support the indicated operation, object, attribute, or attribute value.

PROHIBITED (PROH) - an adjective used to indicate that a conforming PDF/is Creator or Renderer's implementation MUST NOT support the indicated operation, object, attribute, or attribute value.

IGNORED – an adjective used to indicate that a conforming PDF/is Creator or Renderer implementation NEED NOT support the indicated operation, object, attribute, or attribute value; but this feature MAY be added to a future version of this specification.

AS SPECIFIED – is used to indicate that a conforming PDF/is Creator or Render implementation MUST, MAY, or MUST NOT support the indicated operation, object, attribute, or attribute value as is defined in the indicated specification.

OR – a conjunction that specifies a logical 'or', implying that a choice of one or more of the choices specified.

XOR – a conjunction that specifies a logical 'exclusive or', implying that a choice of one and only one of the choices specified.

2.2 Other Terminology

The following terms are introduced and capitalized in order to indicate their specific meaning:

Implement – The specified feature is present in the Document.

Support – A Creator has the capability of Implementing the feature specified, or the Renderer has the capability of understanding and acting on the Implementation.

Document – The PDF/is-formatted electronic representation of a set of one or more pages that the Sender sends to the Receiver.

Renderer – This is the agent (software, hardware or some combination) that converts the Document into a displayed or printed form.

Creator -- This is the agent (software, hardware or some combination) that creates the Document.

Interpolation – See 'Interpolation' in [pdf] pg. 273.

Forward-Reference – In indirect object reference (See [pdf] Section 3.2.9) to an object that appears later in the Document.

3 PDF/is Support

3.1.1 Image Profiles

The following tree diagram shows the relationship among PDF/is Image Profiles:

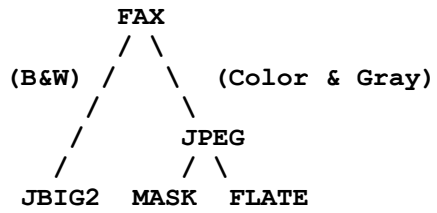


Table 3-1: Image Profiles

Profile	Image Implementation	Reference
<FAX>	'CCITTFaxDecode' Filter	[pdf] Section 3.3.5
<FLATE>	'FlateDecode' Filter	[pdf] Section 3.3.3
<JBIG2>	'JBIG2Decode' Filter	[pdf] Section 3.3.6
<MASK>	Masked Images	[pdf] Section 4.8.5
<JPEG>	'DCTDecode' Filter	[pdf] Section 3.3.7

All PDF/is Renderers and Creators MUST Support PDF/is Profile <FAX>, which is the root node of the tree. All color OR gray scale image Renderers and Creators of PDF/is MUST Support PDF/is Profile <JPEG>. Creators and Renderers that Support a particular profile MUST also Support those profiles on the path that connect it to the root node, and MAY optionally Support profiles not on the path connecting it to the root node. For example, a Creator or Renderer that Supports PDF/is Profile <FLATE> MUST also Support PDF/is Profiles <JPEG> and <FAX>, and MAY optionally Support PDF/is Profile <MASK>, OR <JBIG2>. For another example, a Creator or Renderer that Supports PDF/is Profile <JPEG> MUST also Support PDF/is Profile <FAX>, and MAY optionally Support PDF/is Profile <JBIG2>.

3.1.2 Security Profiles

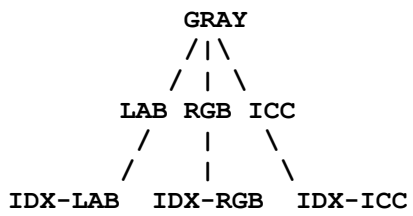
There are several options that MAY be Supported by a Creator or Renderer with regard to security:

Table 3-2: Security Profiles

Profile	Security Implementation	Reference
<STD-ENC>	'Standard' Encryption	[pdf] Section 3.5.2
<PPK-ENC>	'PPKLite' Encryption	[pdf-ppk] Section 3
<DIG-SIG>	Digital Signature	[pdf-ppk] Section 2.2

3.1.3 Color Profiles

The following tree diagram shows the relationship among PDF/is Color Profiles:



There are several color spaces that may be Supported by a Creator or Renderer. These Profiles only apply to Creators or Renderers that Support Image Profiles <JPEG> or <FLATE>. All PDF/is Renderers and Creators that Support Image Profiles <JPEG> OR <FLATE> MUST Support PDF/is Color Profiles <GRAY> and <RGB>. Other Color Profiles are OPTIONAL. Creators and Renderers that Support a particular profile MUST also Support those profiles on the path that connect it to the root node, and MAY optionally Support profiles not on the path connecting it to the root node. For example, a Creator or Renderer that Supports PDF/is Profile <IDX>-<ICC> MUST also Support PDF/is Profiles <ICC> and <GRAY>, and MAY optionally Support PDF/is Profile <LAB>, OR <RGB>, OR <IDX>-<LAB>, OR <IDX>-<ICC>.

Table 3-3: Color Profiles

Profile	Color Space Implementation	Reference
<GRAY>	'CalGray'	[pdf] Page 182
<RGB>	'CalRGB'	[pdf] Page 184
<LAB>	'Lab'	[pdf] Page 187
<ICC>	'ICCBased'	[pdf] Page 189
<IDX-LAB>	'Indexed' and 'Lab'	[pdf] Page 199, 187
<IDX-RGB>	'Indexed' and 'CalRGB'	[pdf] Page 199, 184
<IDX-ICC>	'Indexed' and 'ICCBased'	[pdf] Page 199, 189

<ICCBased> and <Indexed> Color Profiles SHOULD be compressed using a 'FlateDecode' Filter to minimize Document size (See [pdf] Section 3.3.3). If 'FlateDecode' is used in this manner, Profile <FLATE> MUST be specified as being Implemented in the Document.

3.2 PDF Object Requirements

For the table shown below, if an Object/Filter is not Implemented then its associated Profile is not Implemented.

Key:

Creator: Creator Requirement.

Renderer: Render Requirement.

Profile: If the indicated 'PDF Object/Filter' is Implemented then the Document Implements the indicated Profile.

Dependencies: In order to Implement the 'PDF Object/Filter' the Profiles indicated in the Dependencies column MUST also be implemented. Note that a comma ',' in this column indicates an 'and'.

355 **Table 3-4: PDF Object Requirements**

PDF Object/Filter	Creator	Renderer	Dependencies	Reference
'ASCIIHexDecode' Filter	PROH	PROH		[pdf] Section (3.3.1)
'ASCII85Decode' Filter	PROH	PROH		[pdf] Section (3.3.2)
'LZWDecode' Filter	PROH	PROH		[pdf] Section (3.3.3)
'RunLengthDecode' Filter	PROH	PROH		[pdf] Section (3.3.4)
Incremental Updates	PROH	PROH		[pdf] Section (3.4.5)
Functions	PROH	PROH		[pdf] Section (3.9)
Files	PROH	PROH		[pdf] Section (3.10)
Graphics State	PROH	PROH		[pdf] Section (4.3)
Path objects	PROH	PROH		[pdf] Section (4.4)
'DeviceGray' Color Space	PROH	PROH		[pdf] Section (4.5.3)
'DeviceRGB' Color Space	PROH	PROH		[pdf] Section (4.5.3)
'DeviceCMYK' Color Space	PROH	PROH		[pdf] Section (4.5.3)
Pattern Color Space	PROH	PROH		[pdf] Section (4.5.5)
Separation Color Space	PROH	PROH		[pdf] Section (4.5.5)
DeviceN Color Space	PROH	PROH		[pdf] Section (4.5.5)
Pattern Objects	PROH	PROH		[pdf] Section (4.6)
Inline Image Objects	PROH	PROH		[pdf] Section (4.8.6)
Form Xobjects	PROH	PROH		[pdf] Section (4.9)
Postscript Xobjects	PROH	PROH		[pdf] Section (4.10)
Text Objects	PROH	PROH		[pdf] Section (5)
Transparency	PROH	PROH		[pdf] Section (7)
'CCITTFaxDecode' Filter (Image Profile <FAX>)	REQ	REQ		[pdf] Section (3.3.5)
File Header	REQ	REQ		[pdf] Section (3.4.1)
Cross-Reference Table	REQ	REQ		[pdf] Section (3.4.3)
File Trailer	REQ	REQ		[pdf] Section (3.4.4)
Document Catalog	REQ	REQ		[pdf] Section (3.6.1)
Page Tree Nodes	REQ	REQ		[pdf] Section (3.6.2)
Page Objects	REQ	REQ		[pdf] Section (3.6.2)
Content Streams	REQ	REQ		[pdf] Section (3.7.1)
Resource Dictionaries	REQ	REQ		[pdf] Section (3.7.2)

Image XObjects	REQ	REQ		[pdf] Section (4.8)
'FlateDecode' Filter (Image Profile <FLATE>)	OPT	OPT	<JPEG>	[pdf] Section (3.3.3)
'JBIG2Decode' Filter (Image Profile <JBIG2>)	OPT	OPT		[pdf] Section (3.3.6)
'DCTDecode' Filter (Image Profile <JPEG>)	OPT	OPT	<GRAY>,<RGB>	[pdf] Section (3.3.7)
Encryption Dictionary 'Standard' Encryption (Security Profile <STD-ENC>)	OPT	OPT		[pdf] Section (3.5)
Encryption Dictionary 'PPKLite' Encryption (Security Profile <PPK-ENC>)	OPT	OPT	<STD-ENC>	[pdf-ppk] Section (3)
'CalGray' Color Space (Color Profile <GRAY>)	OPT	OPT	<JPEG>	[pdf] pg. 182
'CalRGB' Color Space (Color Profile <RGB>)	OPT	OPT	<JPEG>	[pdf] pg. 184
'Lab' Color Space (Color Profile <LAB>)	OPT	OPT	<JPEG>	[pdf] pg. 187
'ICCBased' Color Space (Color Profile <ICC>)	OPT	OPT	<JPEG>	[pdf] pg. 189
'Indexed' Color Space (Color Profile <IDX>)	OPT	OPT	<LAB> OR <RGB> OR <ICC>	[pdf] pg. 199
Masked Images (Image Profile <MASK>)	OPT	OPT	<JPEG>	[pdf] Section (4.8.5)
Interactive Form Dictionary and Annotation Field Dictionary and Signature Dictionary (Security Profile <DIG-SIG>)	OPT	OPT		[pdf] Section (8.6.1-3) [pdf-ppk] Section (2)
Cached Objects	OPT	REQ		Section 3.4

3.3 PDF Field Specification

The following list describes the object field values of the REQUIRED and OPTIONAL PDF objects in PDF/is. The numbers in '( )'s refer to section numbers in the PDF Specifications [pdf], unless otherwise noted. 'AS SPECIFIED' refers to [pdf] unless otherwise noted.

3.3.1 'PDF/is' object

A new 'PDF Name Registry' (See [pdf] – Appendix E) object that is REQUIRED for a PDF/is document. The existence of this dictionary object is the one and only way to determine if the PDF in question is a PDF/is. Spec:

Table 3-5: PDF/is Object

KEY	TYPE	Specification
Fis_Profiles	Array of Numeric Objects	REQUIRED: An array consisting of [**MAJ_VER MIN_VER IMAGES SECURITY COLOR MEMORY**]
Encrypt	Dictionary	REQ_DEP<STD-ENC XOR PPK-ENC>: See 'Encrypt' key in [pdf] Table 3.12 for Specification.
Root	Dictionary	REQUIRED: See 'Root' key in [pdf] Table 3.12 for Specification.
Info	Dictionary	REQUIRED if 'File Trailer' Implements 'Info', otherwise

		PROHIBITED: See 'Info' key in [pdf] Table 3.12 for Specification.
Fis_NextPage	Dictionary	REQUIRED: An Indirect Object Reference to the first 'Page' object.

368

369 See [pdf] Section 3.2.5 for definition of an 'Array Object'. See [pdf] Section 3.2.2 for definition
370 of a 'Numeric Object'.

371 **3.3.1.1 Fis_Profiles Key**

372 **MAJ_VER**: The 'major' version number of this PDF/is specification to which the Creator
373 conforms to at the time the Document was created. The 'major' version of this
374 specification is currently '0'.

375 **MIN_VER**: The 'minor' version number of this PDF/is specification to which the Creator
376 conforms to at the time the Document was created. The 'minor' version of this
377 specification is currently '3'.

378 **IMAGES, SECURITY, COLOR:**

379 Each value in the array MUST be a 'Numeric Integer Object' (See [pdf] Section 3.2.2) that
380 is the sum of all of the Integer equivalents of the binary 'Bit Positions' for the Profiles that
381 are Implemented in the Document, as indicated under the appropriate section below.
382 The 'Bit Positions' are numbered from 1 (low-order) to 32 (high-order). A '1' in a 'Bit
383 Position' indicates the Profile in indicated. All other Bit Positions for each element MUST
384 be 0. Note that PDF Numeric Integer Objects in fact are represented in signed twos-
385 complement form.
386
387 For example, to indicate that the IMAGES Profiles 'FLATE' (bit position 3 or 100 binary)
388 and 'MASK' (bit position 5, or 10000 binary), the value of '20' (10100 binary) should be
389 used as the value for the 'IMAGES' field.
390
391 The Creator of the Document MUST NOT Implement a Profile that is not indicated in this
392 field. The Creator of the Document MAY Implement all Profiles indicated in this field, but
393 is NOT REQUIRED.

394 Rationale: Since this object must be Implemented at the beginning of the
395 Document, it may not be known for certain which Profiles will be Implemented.
396 This field is an advisory indicator to a Renderer as to which Profiles they MUST
397 Support in order to be able to render the Document for certain. If all Profiles
398 indicated are not Supported, the Document may still be rendered if a non-
399 Supported Profile is indicated but is not actually Implemented in the Document.

400 Note that even though a Profile is higher in the Image Profile tree it SHOULD NOT be
401 indicated in this object unless that feature is Implemented in the document. For example,
402 if the document contained 'FLATE' (FlateDecode) images but no 'JPEG' (DCTDecode)
403 images, only Profile 'FLATE' should be indicated.

404

405 **Table 3-6: PDF/is Object 'IMAGES' Element**

Profile	Bit Position
<FAX>	1
<JBIG2>	2
<FLATE>	3

<JPEG>	4
<MASK>	5

406 **Table 3-7: PDF/is Object 'SECURITY' Element**

Profile	Bit Position
<STD-ENC>	1
<PPK-ENC>	2
<DIG-SIG>	3

407 **Table 3-8: PDF/is Object 'COLOR' Element**

Profile	Bit Position
<GRAY>	1
<RGB>	2
<LAB>	3
<ICC>	4
<IDX>	5

408

409

410 **MEMORY**: A 'Numeric Object' that is the decimal value of the minimum amount of cache
411 memory the Renderer will need to cache all objects necessary to render any particular
412 page. This memory MUST be available for PDF/is data file caching and MUST not be
413 part of any image processing or page buffer memory.

414 The value specified for 'MEMORY' is in addition to a base memory requirement of 2
415 Megabytes (2^21 bytes).

416

417 An example of the PDF/is object for a Document containing a CalRGB color space (Profile
418 <RGB>), masked (Profile <MASK>), JPEG image (Profile <JPEG>) that's Standard
419 encrypted (Profile <STD-ENC>) would look like this:

```
420 1 0 obj
421 <<
422     /Fis_Profiles [0 3 24 1 1 0]
423     /Encrypt 2 0 R
424     /Root 3 0 R
425     /Info 4 0 R
426     /Fis_NextPage 5 0 R
427 >>
428 endobj
```

429

430 ### 3.3.2 'FlateDecode' Filter

431 See [pdf] Section 3.3.3, [RFC1950], and [RFC1951].

432 **Table 3-9: FlateDecode Filter**

Field	Specification
<All Fields>	AS SPECIFIED

433

3.3.3 'CCITTFaxDecode' Filter

See [pdf] Section 3.3.5, [T.4], and [T.6]. Note that only Group 4 images are Supported by PDF/is, see 'K', below.

Table 3-10: CCITTFaxDecode Filter

Field	Specification
'K'	MUST have a value of -1.
'EndOfLine'	AS SPECIFIED
'EncodedByteAlign'	AS SPECIFIED
'Columns'	AS SPECIFIED
'Rows'	AS SPECIFIED
'EndOfBlock'	AS SPECIFIED
'BlackIs1'	AS SPECIFIED
'DamagedRowsBeforeError'	AS SPECIFIED

3.3.4 'JBIG2Decode' Filter

See [pdf] Section 3.3.6, [jbig2], and [T.89].

Table 3-11: JBIG2Decode Filter

Field	Specification
<All Details>	AS SPECIFIED, except as noted below.

- The Creator MUST NOT Implement any JBIG2 feature that is NOT specified in **Profile 4** (0x00000104 Medium lossy/lossless arithmetic) of [T.89].
- All Renderers MUST support at least "Level 2" Memory (See [T.89], Table 1, Item 18).
- The Creator MUST adhere to the Function and Memory constraints as specified in [T.89].

3.3.5 'DCTDecode' Filter

See [pdf] Section 3.3.7, [ps-jpeg], [ps], and [jpeg]. PDF/is supports both the JPEG Baseline DCT and Extended sequential DCT compressed image formats.

Table 3-12: DCTDecode Filter

Field	Specification
<All Details>	AS SPECIFIED, except as noted below.

- Images MUST NOT have interleaved scans.
- Images MUST NOT be encoded using 'Progressive JPEG'.
- The Renderer MUST adhere to the Memory requirements specified in Section 11 "RAM Requirements" of [ps-jpeg] for the Renderers Supported image resolution(s).

3.3.6 File Trailer

See [pdf] Table 3.12.

Table 3-13: File Trailer

Field	Specification
'Size'	AS SPECIFIED
'Prev'	PROHIBITED
'Root'	AS SPECIFIED
'Encrypt'	AS SPECIFIED, but PROHIBITED if the Document is to be PDF/X-3 Compliant (See [pdf-x3]).
'Info'	REQUIRED.
'ID'	REQUIRED. MUST use a pseudo-random number in place of 'File Size' when generating this value. See [pdf] Section 9.3 for guidelines on how to generate this value. Rationale: Using a random number in place of file size is due to the requirements of using this field in generating the encryption key for the 'standard encryption' algorithm ([pdf] Step 5 of Algorithm 3.2, pg. 78): file size will not be known at the time this field is needed.

3.3.7 Encryption Dictionary

See [pdf] Table 3.13 and [pdf-ppk] Table 3.

Note that if a Document is Standard encrypted (Profile <STD-ENC>), the 'ID' field of the File Trailer MUST be calculated before the Encryption Dictionary is written. The 'ID' MUST then be cached until the 'File Trailer' is written.

Table 3-14: Encryption Dictionary

Field	Specification
'Filter'	MUST have a value of either 'Standard' or 'Adobe.PPKLite'.
'V'	MUST have a value of '2'.
'Length'	AS SPECIFIED
'R'	AS SPECIFIED
'O'	REQ if <STD-ENC>, PROH otherwise
'U'	REQ if <STD-ENC>, PROH otherwise
'P'	REQ if <STD-ENC>, PROH otherwise
'SubFilter'	MUST be 'adbe.pkcs7.s4' if <PPK-ENC>, PROH otherwise
'Recipients'	REQ if <STD-ENC>, PROH otherwise

3.3.8 Document Catalog

See [pdf] Table 3.16.

Table 3-15: Document Catalog

Field	Specification
'Type'	AS SPECIFIED
'Version'	AS SPECIFIED
'Pages'	AS SPECIFIED

'PageLabels'	IGNORED
'Names'	IGNORED.
'Dests'	IGNORED.
'ViewerPreferences'	IGNORED.
'PageLayout'	IGNORED.
'PageMode'	IGNORED.
'Outlines'	IGNORED.
'Threads'	IGNORED.
'OpenAction'	IGNORED.
'AA'	IGNORED.
'URI'	IGNORED.
'AcroForm'	REQ if <DIG-SIG>, PROH otherwise
'Metadata'	IGNORED.
'StructTreeRoot'	IGNORED.
'MarkInfo'	IGNORED.
'Lang'	IGNORED.
'SpiderInfo'	IGNORED.
'OutputIntents'	PROHIBITED.

3.3.9 Page Tree Nodes

See [pdf] Table 3.17.

Table 3-16: Page Tree Nodes

Field	Specification
'Type'	AS SPECIFIED
'Parent'	AS SPECIFIED
'Kids'	AS SPECIFIED
'Count'	AS SPECIFIED
<All 'Page Object' Fields, see [pdf] Table 3.18>	PROHIBITED

3.3.10 Page Objects

See [pdf] Table 3.18.

Table 3-17: Page Objects

Field	Specification
'Type'	AS SPECIFIED
'Parent'	AS SPECIFIED
'LastModified'	AS SPECIFIED
'Resources'	MUST NOT be inherited
'MediaBox'	MUST NOT be inherited
'CropBox'	MUST NOT be inherited. If Present, the TrimBox MUST NOT extend beyond the boundaries of the CropBox.
'BleedBox'	AS SPECIFIED. If Present, the TrimBox MUST NOT extend beyond the boundaries of the BleedBox.
'TrimBox'	REQUIRED.
'ArtBox'	PROHIBITED.
'BoxColorInfo'	PROHIBITED.

'Contents'	AS SPECIFIED.
'Rotate'	MUST NOT be inherited
'Group'	PROHIBITED.
'Thumb'	IGNORED.
'B'	IGNORED.
'Dur'	IGNORED.
'Trans'	IGNORED.
'Annots'	IGNORED.
'AA'	IGNORED.
'Metadata'	IGNORED.
'PieceInfo'	IGNORED.
'StructParents'	IGNORED.
'ID'	IGNORED.
'PZ'	IGNORED.
'SeparationInfo'	PROHIBITED.
'Type'	AS SPECIFIED
'Fis_NextPage'	REQUIRED: An Indirect Object Reference to the next 'Page' object or a 'Page Node' if this is the last page.

3.3.11 Content Stream Operators

See [pdf] Table 4.1. A conforming Renderer MUST be able to parse the Content Stream operators listed below, but only must be able to act upon the operators that are not listed as IGNORED.

Table 3-18: Content Stream Operators

Field	Specification	Reference
'q'	AS SPECIFIED	[pdf] Table 4.7
'Q'	AS SPECIFIED	[pdf] Table 4.7
'cm'	MUST be [Sx 0 0 Sy Tx Ty], See Below	[pdf] Table 4.7
'Do'	AS SPECIFIED	[pdf] Table 4.34
'MP'	IGNORED	[pdf] Table 9.8
'DP'	IGNORED	[pdf] Table 9.8
'BMC'	IGNORED	[pdf] Table 9.8
'BDC'	IGNORED	[pdf] Table 9.8
'EMC'	IGNORED	[pdf] Table 9.8
'BX'	AS SPECIFIED	[pdf] Table 3.20
'EX'	AS SPECIFIED	[pdf] Table 3.20
<All other Operators>	PROHIBITED	

cm: See [pdf] Section 4.2.3.

Given:

Wi = Width (X-direction) of the Image in inches.

Hi = Height (Y-direction) of the Image in inches.

Xi = Horizontal translation, in inches, from the left edge of the page to the top of the image.

Yi = Vertical translation, in inches, from the top edge of the page to the top of the image.

The Creator MUST ensure that the following is true:

Sx = Wi * 72

Sy = Hi * 72

Tx = Xi * 72

Ty = Yi * 72

Do:

Given:

Img = The 'Image XObject' associated with the 'Do' operator.

Cm = The current 'cm' operation in effect for 'Img'.

Wp = 'Width' field of 'Img'.

Hp = 'Height' field of 'Img'.

Sx = 'Sx' value of 'Cm'.

Sy = 'Sy' value of 'Cm'.

The following MAY be assumed by either the Creator or the Renderer:

Rx = (Wp * 72 / Sx) = The resolution, in the X-direction, of 'Img', in dots per inch.

Ry = (Hp * 72 / Sy) = The resolution, in the Y-direction, of 'Img', in dots per inch.

The values for Rx and Ry for all images in a conforming Document MUST have a value greater than or equal to 200.

3.3.12 Resource Dictionaries

See [pdf] Table 3.21.

The Resource Dictionary MUST reference all Image XObjects and ColorSpaces that are used on the current page. The position of the image objects, their masks, and color spaces with respect to each other is defined in the Image XObject section of this specification.

Table 3-19: Resource Dictionaries

Field	Specification
'ExtGState'	PROHIBITED.
'ColorSpace'	AS SPECIFIED.
'Pattern'	PROHIBITED.
'Shading'	PROHIBITED.
'XObject'	AS SPECIFIED.
'Font'	PROHIBITED.
'ProcSet'	'Text' Proc Sets PROHIBITED, all others AS SPECIFIED.
'Properties'	IGNORED.

524

525 **3.3.13 Color Spaces**

526 See [pdf] Section 4.5.

527 **Table 3-20: Color Spaces**

Field	Specification
'Lab'	AS SPECIFIED
'DeviceGray'	PROHIBITED
'DeviceRGB'	PROHIBITED
'DeviceCMYK'	PROHIBITED
'CalGray'	AS SPECIFIED
'CalRGB'	AS SPECIFIED
'ICCBased'	AS SPECIFIED, but may be compressed using 'FlateDecode' if Profile <FLATE> is Implemented.
'Indexed'	AS SPECIFIED, but may be compressed using 'FlateDecode' if Profile <FLATE> is Implemented.
'Pattern'	PROHIBITED
'Separation'	PROHIBITED
'DeviceN'	PROHIBITED

528

529 **3.3.14 Image XObjects**

530

532 See [pdf] Table 4.35 for description of the following table.

533 **Table 3-21: Image XObjects**

Field	Specification
'Type'	MUST be 'XObject'
'Subtype'	MUST be 'Image'
'Width'	AS SPECIFIED
'Height'	AS SPECIFIED
'ColorSpace'	AS SPECIFIED, and see below.
'BitsPerComponent'	AS SPECIFIED
'Intent'	PROHIBITED.
'ImageMask'	AS SPECIFIED, if Profile <MASK>
'Mask'	AS SPECIFIED, if Profile <MASK>, and see below.
'SMask'	PROHIBITED.
'Decode'	AS SPECIFIED.
'Interpolate'	MUST be 'true'
'Alternates'	IGNORED
'Name'	IGNORED.
'StructParent'	IGNORED.
'ID'	IGNORED.
'OPI'	PROHIBITED.
'Metadata'	IGNORED.

534

- An ‘ImageMask’, if indicated in an Image XObject, MUST appear in the Document before the Image XObject that references it.
- If an ‘ICCBased’ or ‘Indexed’ color space is indicated in an Image XObject, the data for the color space MUST appear in the Document before the Image XObject that references it.

3.3.15 Masked Images

See [pdf] Section 4.8.5.

Table 3-22: Masked Images

Field	Specification
<All Fields>	AS SPECIFIED

3.3.16 Interactive Form Dictionary

See [pdf] Table 8.47.

Table 3-23: Interactive Form Dictionary

Field	Specification
‘Fields’	MUST be an indirect object of an ‘Annotation Field Dictionary’.
‘NeedAppearances’	PROHIBITED
‘SigFlags’	MUST be ‘3’
‘CO’	PROHIBITED
‘DR’	PROHIBITED
‘DA’	PROHIBITED
‘Q’	PROHIBITED

3.3.17 Annotation Field Dictionary

See [pdf] Tables 8.10 & 8.49. This dictionary consists of entries from both a ‘Annotation Dictionary (Table 8.10) and a ‘Field Dictionary’ (Table 8.49).

Table 3-24: Annotation Field Dictionary

Field	Specification
‘Type’	MUST be ‘Annot’
‘Subtype’	MUST be ‘Widget’
‘Contents’	IGNORED
‘P’	IGNORED
‘Rect’	MUST be ‘[0 0 0 0]’
‘NM’	IGNORED
‘F’	IGNORED
‘BS’	IGNORED
‘Border’	IGNORED
‘AP’	IGNORED

'AS'	IGNORED
'C'	IGNORED
'CA'	IGNORED
'T'	IGNORED
'Popup'	IGNORED
'A'	IGNORED
'AA'	IGNORED
'StructParent'	IGNORED
'FT'	MUST be 'Sig'
'Parent'	PROHIBITED.
'Kids'	PROHIBTED.
'T'	AS SPECIFIED.
'TU'	AS SPECIFIED.
'TM'	IGNORED.
'Ff'	MUST be '1'.
'V'	MUST be an indirect object to a 'Signature Dictionary'.
'DV'	IGNORED.
'AA'	IGNORED.

553
554

555 **3.3.18 Signature Dictionary**

556 See [pdf] Table 8.60 and [pdf-ppk] Table 2.

557 The Digital Signature format MUST only be in the 'Raw Format', see [pdf-ppk] Section 2.2.

558 **Table 3-25: Signature Dictionary**

Field	Specification
'Type'	MUST be 'Sig'
'Filter'	MUST be 'Adobe.PPKLite'
'SubFilter'	MUST be 'adbe.x509.rsa_sha1'
'Name'	AS SPECIFIED.
'Reason'	AS SPECIFIED.
'Location'	AS SPECIFIED.
'M'	AS SPECIFIED.
'ByteRange'	PROHIBITED (Implies all bytes in the Document with the exclusion of the bytes represented by the value of the 'Cert' field. See [pdf] for this field)
'Contents'	AS SPECIFIED.
'Cert'	AS SPECIFIED.
'R'	AS SPECIFIED.
'V'	AS SPECIFIED.
'ADBE_Build'	AS SPECIFIED.
'ADBE_AuthType'	AS SPECIFIED.
'ADBE_PwdTime'	AS SPECIFIED.

559

560 **3.3.19 Document Information Dictionary**

561 See [pdf] Table 9.2.

562 **Table 3-26: Document Information Dictionary**

Field	Specification
'Title'	REQUIRED
'Author'	REQUIRED
'Subject'	AS SPECIFIED
'Keywords'	AS SPECIFIED
'Creator'	AS SPECIFIED
'Producer'	AS SPECIFIED
'CreationDate'	REQUIRED
'ModDate'	REQUIRED
'Trapped'	REQUIRED, MUST be either 'TRUE' or 'FALSE'. Partially Trapped files are PROHIBITED.
'GTS_PDFXVersion'	PROHIBITED if Profile <STD-ENC> or <PPK-ENC> is Implemented; otherwise MUST be "(PDF/X-3:2002)"

563

564 3.4 Cached Objects

565 If an object MAY be used for more than a single page, it may be practical to maintain the object in
566 the Renderer's memory. To accomplish this, the Creator should invoke the 'Cache Hold'
567 mechanism. Once an object is cached, it no longer has to abide by 'Creator Conformance
568 Requirements' 7 and 8 (See Section 4.1).

569 An object that is held in the Renderers cache by the 'Cache Hold' mechanism MUST be
570 maintained in the cache until one of the following conditions is met:

571 The 'Cache Release' mechanism is invoked.

572 The 'Document Catalog' is reached.

573 3.4.1 Cache Hold

574 To specify that an object should not be discarded once the current page is rendered, the object to
575 be 'cached' should have the following 'Name Object' ([pdf] Section 3.2.4) in its 'Dictionary' ([pdf]
576 Section 3.2.6):

577 /Fis_Cache

578 3.4.2 Cache Release

579 To release an object from the Renderer's memory; the following 'Name Object' MUST be placed
580 in the 'Page Object' of the first page in which the object is no longer needed. For example, if the
581 object is question was first found on page 1 and was last used on page 3, the 'Cache Release'
582 should occur in the 'Page Object' for page 4.
583
584 /Fis_Cache OBJECTS
585 Where:
586 OBJECTS: is an array (contained in '[]'s) of indirect object references of the objects that were
587 previously cached and are no longer needed. Indication of an object number that was never
588 cached MUST be ignored.
589 Example:

```
590     3 0 obj
591     /Fis_Cache                  %First object to be cached.
592     …
593     endobj
594     …
```



```
7 0 obj                          %Second object to be cached.
/Fis_Cache
…
endobj
…                                %One or more Page objects in between.
45 0 obj
/Type /Page                      %Page object
/Fis_Cache [3 0 R 7 0 R]         %Objects 3 and 7 are no longer needed.
. . .
```

4 Conformance Requirements

This section specifies the conformance requirements for Renderers and Creators.

4.1 Creator conformance requirements

In order to conform to this specification, a Document Creator:

1. MUST specify the version of PDF (See [pdf] Section 3.4.1) as being ‘PDF 1.4’.

2. MUST place the ‘PDF/is’ object as the first object in the PDF.

3. MUST place any ‘Encryption Dictionary’ object as the second object in the PDF/is Document, if the Document is encrypted.

4. MUST NOT include any private ‘PDF Name Registry’ values/objects (See [pdf] – Appendix E) that effect printed output.

5. MUST place the objects: ‘Interactive Form Dictionary’, ‘Field Dictionary’ and ‘Digital Signature’ object as the last three objects (in that order) in the Document, if the Document is Digitally Signed. Note that in a situation where the Renderer cannot cache the entire document before rendering, the detection of a valid or invalid Digital Signature will only occur after rendering of the entire Document.

6. MUST ensure that each non-IGNORED object have at least one Forward-Reference to such object. The only object that does not have to follow this rule is the ‘PDF/is Object’. Rationale: This will aid the Renderer with knowing which objects will need to be cached and which can be ignored.

7. MUST ensure that all non-IGNORED objects appear in the PDF AFTER the object in which they are first referenced (Satisfied by Requirement 6) and BEFORE the next ‘Page Object’ unless the object is a Cached Object (See Section 3.4).

8. MUST ensure that all object identifiers ([pdf] Section 3.2.9) start at the beginning of a line.

9. MUST ensure that all ‘endobj’ keywords ([pdf] Section 3.2.9) start at the beginning of a line.

10. MUST ensure that all ‘stream’ data ([pdf] Section 3.2.7) does not contain a line beginning with the word “endstream”, aside from the required “endstream” that delimits the end of the stream.

4.2 Renderer conformance requirements

In order to conform to this specification, a Document Renderer:

1. MUST Support all of the REQUIRED PDF/is objects.

2. MUST cache all REQUIRED or Supported OPTIONAL objects as they are encountered (sequentially) in the Document until the next 'Page Object' is encountered. At that point, the page can be rendered and the cache emptied of all non-Cached objects.

3. MUST Interpolate images up or down in resolution, as required, to match the Renderer's Supported image resolution(s).

4. MAY ignore all IGNORED objects that the Creator added to the PDF/is Document.

4.3 File Layout

Given that a Document is fully compliant with this specification, a PDF/is Document will, nominally, take on the following format:

Table 4-1: File Layout

	Object
A	Header (See [pdf], Section 3.4.1)
B	Encryption Object (if Profile <STD-ENC> XOR <PPK-ENC>)
C	Page object for page 1
D	Resources for page 1
E	Content object for page 1
F	Color Space(s) for page 1 (if Profile <FLATE> or <JPEG>)
G	Image Mask(s) for page 1 (if Profile <MASK>)
H	Image XObject(s) for page 1
I	[Repeat C – H for all remaining pages, in order]
J	Document Catalog
K	Page Node(s)
L	Interactive Form Dictionary (if Profile <DIG-SIG>)
M	Annotation Field Dictionary (if Profile <SIG-SIG>)
N	Signature Dictionary (if Profile <DIG-SIG>)
O	File Trailer

5 Issues

- None currently.

6 Sample PDF/is PDFs

The 'source' of all of the sample documents in this section can be viewed with any text editor but should only be modified with a binary editor, as the stream data contained therein is not compatible with text editors. Comments on the format of the documents are contained within the documents themselves.

All of the samples are different versions of the same document.

1: The first sample is an unencrypted, single page, 'CCITTFaxDecode' masked, 'DCTDecode' color ICCBased color space foreground image with a 'FlateDecode' gray scale Indexed ICCBased color space background image. The images use 'FlateDecode' compression on the 'ICCBased' and 'Indexed' Color Spaces.

ftp://pwg.org/pub/pwg/QUALDOCS/SamplePDFax/base-02.pdf

2: The next sample has been encrypted with 'Standard' encryption. The 'user' password is '12345'; the 'owner' password is '54321'. The document has also been Digitally Signed: the document will fail a digital signature check since it has been tampered with. To see the digital signature in Acrobat (or Acrobat Reader), select the 'Signature' tab on the left side of the screen.

ftp://pwg.org/pub/pwg/QUALDOCS/SamplePDFax/stdEncryptSigned-02.pdf

7 Normative References

[pdf]
Adobe Systems, "PDF Reference, third edition, Adobe Portable Document Format Version 1.4", Addison-Wesley, December 2001, http://partners.adobe.com/asn/developer/acrosdk/docs/filefmtspecs/PDFReference.pdf. Also see errata: http://partners.adobe.com/asn/developer/acrosdk/docs/PDF14errata.txt.

[pdf-ppk]
Pravetz, J., "PDF Public-Key Digital Signature and Encryption Specification", Version 3.2, Adobe Systems, September 2001, http://partners.adobe.com/asn/developer/pdfs/tn/ppk_pdfspec.pdf

[pdf-x3]
ISO/TC 130, "Complete exchange suitable for colour-managed workflows (PDF/X-3)", ISO 15930-3:2002, September 2002.

[ps-jpeg]
Adobe Systems Incorporated, "Supporting the DCT Filters in PostScript Level 2", November 1992, http://partners.adobe.com/asn/developer/pdfs/tn/5116.DCT_Filter.pdf

[ps]
Adobe Systems Incorporated, "PostScript Language Reference third edition", Addiseon-Wesley, 1999, http://partners.adobe.com/asn/developer/pdfs/tn/PLRM.pdf. Also see errata: http://partners.adobe.com/asn/developer/pdfs/tn/PSerrata.txt.

[ifx]
Moore, Songer, Hastings, "IPPFAX/1.0 Protocol" PWG Draft Standard D0.12, 2002, ftp://pwg.org/pub/pwg/QUALDOCS/pwg-ifx-PDF/is-D12-021028.pdf

[ifx-req]
Moore, P., "IPP Fax transport requirements", October 16, 2000, ftp://pwg.org/pub/pwg/QUALDOCS/requirements/ifx-transport-requirements-01.pdf

[T.4]
ITU-T Recommendation T.4, "Standardization of group 3 facsimile apparatus for document transmission", October 1997

[T.6]
ITU-T Recommendation T.6, “Facsimile coding schemes and coding control functions for group 4 facsimile apparatus”, November 1988

[T.89]
ITU-T Recommendation T.89, “Application profiles for Recommendation T.88 – Lossy/lossless coding of bi-level images (JBIG2) for facsimile”, September 2001

[RFC2119]
Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, September 2000, ftp://ftp.rfc-editor.org/in-notes/pdfrfc/rfc2911.txt.pdf.

[RFC2911]
Hastings, Herriot, deBry, Isaacson, Powell, “Internet Printing Protocol/1.1: Model and Semantics”, September 2000, ftp://ftp.rfc-editor.org/in-notes/pdfrfc/rfc2911.txt.pdf.

[jpeg]
JTC 1/SC 29, “Information technology – Digital compression and coding of continuous-tone images: Requirements and guidelines”, ISO/IEC 10918-1:1994, 1994.

[jbig2]
JTC 1/SC 29, “Information technology – Lossy/lossless coding of bi-level images”, ISO/IEC 14492:2001, December 2001.

[RFC1950]
Deutsch, Gailly, “ZLIB Compressed Data Format Specification version 3.3”, May 1996, ftp://ftp.isi.edu/in-notes/rfc1950.pdf.

[RFC1951]
Deutsch, “DEFLATE Compressed Data Format Specification version 1.3”, May 1996, ftp://ftp.isi.edu/in-notes/rfc1951.pdf.

8 Informative References

[RFC2542]
Masinter , "Terminology and Goals for Internet Fax", RFC2542, March 1999, ftp://ftp.rfc-editor.org/in-notes/pdfrfc/rfc2542.txt.pdf.

9 Revision History (to be removed when standard is approved)

Revision	Date	Author	Notes
1	10/9/02	Rick Seeler, Adobe Systems	Initial version
2	10/23/02	Rick Seeler, Adobe Systems	
3		Rick Seeler, Adobe Systems	

10 Contributors

John Pulera - Minolta mailto:jpulera@minolta-mil.com
Gail Songer - Peerless mailto:gsonger@peerless.com
Tom Hastings - Xerox mailto:hastings@cp10.es.xerox.com

Rob Buckley - Xerox mailto:rbuckley@crt.xerox.com
Lloyd McIntyre - Xerox mailto:Lloyd.McIntyre@pahv.xerox.com

11 Acknowledgments

Kari Poysa - Xerox mailto:Kari.Poysa@usa.xerox.com

12 Author's Address

Rick Seeler
Adobe Systems Incorporated
321 Park Ave., E13
San Jose, CA 95110
Phone: 1+408 536-4393
Fax: 1+408 537-8077
e-mail: mailto:rseeler@adobe.com

13 Appendix A

13.1 Intellectual Property Statement – Adobe Systems Incorporated

The following statement is in addition to the Intellectual Property Statement in the PDF Reference (See [pdf] Section 1.4).

Patent Clarification Notice Specific to Use of PDF for IPP FAX Protocol

Adobe has a number of patents covering technology that is disclosed in the Portable Document Format (PDF) Specification, version 1.4 and later, as documented in PDF Reference and associated Technical Notes (the "PDF Specification"). Adobe desires to promote the use of PDF as the file format for a future, IPP FAX Protocol to be proposed, recommended, finalized and published by the IEEE Printer Working Group (the "IPP FAX Standard").

This Patent Clarification Notice is in addition to the permissions statement set forth in Section 1.4 of the PDF Reference which shall also apply to Adobe's contribution to the IPP FAX Standard.

Accordingly, Adobe agrees to provide a Royalty Free License to all Essential Claims solely for the purpose of implementing the IPP FAX Standard. Adobe and the IEEE Printer Working Group will identify and establish, within the final, published release of the IPP FAX Standard, a process whereby implementers of the IPP FAX Standard can request and obtain the above license.

No license shall be extended to those implementing only draft versions of the IPP FAX Standard.

A "Royalty Free License" shall mean a license that:

i) shall be available to all implementers of the IPP FAX Standard worldwide, whether or not members of the IEEE Printer Working Group;
ii) shall extend to all Essential Claims owned or controlled by Adobe and its Affiliates;
iii) shall not be conditioned on payment of royalties, fees or other consideration except as described in (iv) and (v) below;
iv) may be conditioned on a grant of a reciprocal license on identical terms to all Essential

Claims owned or controlled by the licensee and its Affiliates; and

v) may include reasonable, customary terms relating to operation or maintenance of the license relationship including but not limited to the following: choice of law, dispute resolution, and patent notices.

"Essential Claims" shall mean all claims in any patent or patent application, in any jurisdiction in the world, that (A) Adobe and/or its Affiliates own and (B) that would be necessarily infringed by implementation of the IPP FAX Standard. A claim is necessarily infringed hereunder only when a licensee can prove that it is not possible to avoid infringing it because there is no non-infringing alternative for implementing the required portions of the IPP FAX Standard. Existence of a non-infringing alternative shall be judged based on the state of the art at the time a licensee implements the IPP FAX Standard.

The following are expressly excluded from and shall not be deemed to constitute Essential Claims:

1) any claims other than as set forth above even if contained in the same patent as Essential Claims; and
2) claims that would be infringed only by
   a) portions of an implementation that are not required by the IPP FAX Standard
   b) enabling technologies that may be necessary to make or use any product or portion thereof that complies with the IPP FAX Standard but are not themselves expressly set forth in the IPP FAX Standard; or
   c) the implementation of technology developed elsewhere and merely incorporated by reference into the IPP FAX Standard.

For purposes of the Essential Claims definition, the "IPP FAX Standard" shall be deemed to include only architectural and interoperability requirements and shall not include any implementation examples or any other material that merely illustrates the requirements of the IPP FAX Standard.

An "Affiliate" of a first entity is a second entity that is controlled (greater than 50%) by, in control of, or under common control with the first entity.