



Errata for the PDF Reference, sixth edition, version 1.7

October 2007

© 2007 Adobe Systems Incorporated. All rights reserved.

Errata for the PDF Reference, sixth edition, version 1.7

October 2007

If this guide is distributed with software that includes an end user agreement, this guide, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by any such license, no part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated. Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end-user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names, company logos and user names in sample material or sample forms included in this documentation and/or software are for demonstration purposes only and are not intended to refer to any actual organization or persons.

Adobe, the Adobe logo, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

OpenType is a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

All other trademarks are the property of their respective owners.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. Government End Users. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §§227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

Errata for the PDF Reference, sixth edition, version 1.7

This document describes errors and omissions in the *PDF Reference, sixth edition, version 1.7*.

Chapter 3, Syntax

3.1, Lexical Conventions

Page 48

Replace the first sentence in the second paragraph in section 3.1 with the following:

A non-encrypted PDF can be entirely represented using byte values corresponding to the visible printable subset of the character set defined in ANSI X3.4-1986, plus white space characters. For historical reasons, the character set defined in ANSI X3.4-1986 is hereafter referred to as ASCII.

3.2, Objects

Page 59

Replace the first note on this page with the following paragraph.

Multiple entries in the same dictionary shall not have the same key.

3.3, Filters

Page 90

Append the following sentence to the first paragraph in section 3.3.9, Crypt Filter:

The **Crypt** filter shall be the first filter in the **Filter** array entry.

Append the following paragraph to the end of section 3.3.9, "Crypt Filter":

If a stream specifies a crypt filter, then the security handler does not apply Algorithm 3.1 (page 119) to the key prior to decrypting the stream. Instead, the security handler decrypts the stream using the key as is. Section 3.3, "Filters" explains how a stream specifies filters.

3.4, File Structure

Page 98

*In Table 3.13, "Entries in the file trailer dictionary," change the description for the **ID** entry to the following:*

KEY	TYPE	VALUE
ID	array	<p>(Required if an Encrypt entry is present; optional otherwise; PDF 1.1) An array of two byte-strings constituting a file identifier (see section 10.3, "File Identifiers") for the file. If there is an Encrypt entry this array and the two byte-strings shall be direct objects and shall be unencrypted.</p> <p>Note: Because the ID entries are not encrypted, it is possible to check the ID key to assure that the correct file is being accessed without decrypting the file. The restrictions that the string be a direct object and not be encrypted assure that this is possible.</p> <p>Note: Although this entry is optional, its absence might prevent the file from functioning in some workflows that depend on files being uniquely identified.</p> <p>Note: The values of the ID strings are used as input to the encryption algorithm. If these strings were indirect, or if the ID array were indirect, these strings would be encrypted when written. This would result in a circular condition for a reader: the ID strings must be decrypted in order to use them to decrypt strings, including the ID strings themselves. The preceding restriction prevents this circular condition.</p>

3.5, Encryption

Page 115

Replace the second sentence in the first paragraph in section 3.5, "Encryption" with the following paragraphs and bullets:

Encryption applies to all strings and streams in the document's PDF file, with the following exceptions:

- The values for the **ID** entry in the file trailer
- Any strings in an **Encrypt** dictionary
- Any strings that are inside streams such as content streams and compressed object streams, which themselves are encrypted

Encryption is not applied to other object types such as integers and boolean values, which are used primarily to convey information about the document's structure rather than its contents.

Page 120

In section 3.5.2, *Standard Security Handler*, replace the second paragraph and the two bullets that follow it with the following. The first of these bullets straddles pages 120 and 121, and the second appears on page 121.

If a user attempts to open an encrypted document that has a user password, the application shall first try to authenticate the encrypted document using the padding string defined in "Encryption Key Algorithm" (default user password) on page 124 :

- If this authentication attempt is successful, the application may open, decrypt and display the document on the screen.

- If this authentication attempt fails, the application should prompt for a password. Correctly supplying either password (owner or user password) should enable the user to open the document, decrypt it, and display it on the screen.

Whether additional operations shall be allowed on a decrypted document depends on which password (if any) was supplied when the document was opened and on any access restrictions that were specified when the document was created:

- Opening the document with the correct owner password access to the document. This unlimited access includes the ability to change the document's passwords and access permissions.
- Opening the document with the correct user password (or opening a document with the default password) should allow additional operations to be performed according to the user access permissions specified in the document's encryption dictionary.

Page 122

Append the following sentence to the end of the first paragraph on the page:

Public-Key security handlers in this case shall use crypt filters named **DefaultCryptFilter** when all document content is encrypted and named **DefEmbeddedFile** when file attachments only are encrypted in place of **StdCF** name. This distinction of names shall not be used as indicator of the type of the security handler or encryption.

Page 123

Add the following paragraph just before the note:

The value of the **R** entry shall determine which flags in the **P** entry are valid. If **R** is 2, applications shall ignore all flags other than those at bit positions 3, 4, 5, and 6. If **R** is 3 or 4, applications shall ignore all flags other than those at bit positions 3, 4, 5, 6, 9, 10, 11, and 12.

Page 129

Add the following entry to Table 3.21, "Additional encryption dictionary entries for public-key security handlers":

KEY	TYPE	VALUE
P	integer	(Required) A set of flags specifying which operations shall be permitted when the document is opened with user access. If bit 2 is set to 1, all other bits are ignored and all operations are permitted. If bit 2 is set to 0, permission for operations are based on the values of the remaining flags defined in Table 3.21a.

Add the following paragraphs and table after Table 3.2, "Additional encryption dictionary entries for public-key security handlers."

The value of the **P** entry shall be interpreted as an unsigned 32-bit quantity containing a set of flags specifying which access permissions shall be granted when the document is opened with user access. Table 23 shows the meanings of these flags. Bit positions within the flag word shall be numbered from 1 (low-order) to 32 (high-order). A 1 bit in any position shall enable the corresponding access permission.

Applications shall ignore all flags other than those at bit positions 2, 3, 4, 5, 6, 9, 10, 11, and 12

Table 3.21a Public-Key security handler user access permissions

Bit position	Meaning
2	Permit change of encryption and enable all other permissions.
3	Print the document (possibly not at the highest quality level, depending on whether bit 12 is also set).
4	Modify the contents of the document by operations other than those controlled by bits 6, 9, and 11.
5	Copy or otherwise extract text and graphics from the document by operations other than that controlled by bit 10.
6	Add or modify text annotations, fill in interactive form fields, and, if bit 4 is also set, create or modify interactive form fields (including signature fields).
9	Fill in existing interactive form fields (including signature fields), even if bit 6 is clear.
10	Extract text and graphics (in support of accessibility to users with disabilities or for other purposes).
11	Assemble the document (insert, rotate, or delete pages and create bookmarks or thumbnail images), even if bit 4 is clear.
12	Print the document to a representation from which a faithful digital copy of the PDF content could be generated. When this bit is clear (and bit 3 is set), printing is limited to a low-level representation of the appearance, possibly of degraded quality.

Page 132

Add the following paragraph after the paragraph that begins “Authorization to decrypt ...”:

In the file specification dictionary (see 3.10.2, “File Specification Dictionaries”), related files (**RF**) shall use the same crypt filter as the embedded file (**EF**).

Page 134

The **Length** entry in Table 3.22, “Entries common to all crypt filter dictionaries” contains a note. Append the following sentence to that note:

Standard security handler expresses the length in multiples of 8 (16 means 128) and public-key security handler expresses it as is (128 means 128).

3.6, Document Structure

Page 146

To the description of the **Contents** entry in Table 3.27, “Entries in a page object”, append the following sentence:

Applications that produce PDF shall not create a **Contents** array containing no elements.

3.8, Common Data Structures

Page 160

Replace the first two sentences in the paragraph that begins “The apostrophe character ...” with the following sentences:

The prefix `D` shall be present, the year field (`YYYY`) shall be present and all other fields may be present but only if all of their preceding fields are also present. The `'` (apostrophe) following the hour offset field (`HH`) shall only be present if the `HH` field is present. The minute offset field (`mm`) shall only be present if the `'` following the hour offset field (`HH`) is present.

3.10, File Specifications

Page 186

Table 3.43, “Entries in an embedded file parameter dictionary,” change the sentence in the value column for the **Size** key to the following:

The size of the uncompressed embedded file, in bytes.

Page 189

Table 3.45, “Entries in a collection dictionary” describes a key called “Other keys”. Change the second paragraph in the Value description for this key to the following:

The type of each entry shall match the type of data identified by the collection field dictionary (see Table 8.8 on page 59) referenced by the same key in the collection schema dictionary (see Table 3.46). For example, if the corresponding collection field has a **Subtype** entry of **S**, then the entry is a text string.

Table 3.46, “Entries in a collection subitem dictionary” describes the **D** key. Change the second sentence in that description to the following:

The type of data shall match the data type identified by the corresponding collection field dictionary.

Chapter 4, Graphics

Section 4.4, Path Construction and Painting

Page 232

Add the following paragraph just before the section “Nonzero Winding Number Rule.”

For more information on scan conversion of zero-width or zero-height fill regions, see the note at the top of page 511, which is in 6.5.3, “Scan Conversion Rules.”

Chapter 5, Text

Section 5.3, Text Objects

Page 406

In Table 5.5, “Text-positioning operators,” replace the **T*** entry with the following entry:

OPERANDS	OPERATOR	DESCRIPTION
—	T*	<p>Move to the start of the next line. This operator has the same effect as the code</p> $0 \ -T_l \ T_d$ <p>where T_l is the current leading parameter in the text state. The negative of T_l is used here because T_l is the text leading expressed as a positive number. Going to the next line entails decreasing the y coordinate.</p>

Section 5.8, Embedded Font Programs

Page 466

In Table 5.23, “Embedded font organization for various font types,” in the description for the **FontFile3** key **OpenType** subtype, replace the existing entry with the following entry:

KEY	SUBTYPE	DESCRIPTION
FontFile3	OpenType	<p>(PDF 1.6) OpenType® font program, as described in the <i>OpenType Specification v.1.4</i> (see the Bibliography). OpenType is an extension of TrueType that allows inclusion of font programs that use the Compact Font Format (CFF).</p> <p>A FontFile3 entry with an OpenType subtype may appear in the font descriptor for the following types of font dictionaries:</p> <ul style="list-style-type: none">• A TrueType font dictionary or a CIDFontType2 CIDFont dictionary, if the embedded font program contains a “glyf” table. In addition to the “glyf” table, the font program must include the following tables: “head”, “hhea”, “hmtx”, “loca”, and “maxp”. The “cvt ” (notice the trailing space), “fpgm”, and “prep” tables must also be included if they are required by the font instructions.• A CIDFontType0 CIDFont dictionary, if the embedded font program contains a “CFF ” table (notice the trailing space) with a Top DICT that uses CIDFont operators (this is equivalent to subtype CIDFontType0C above). In addition to the “CFF ” table, the font program must include the “cmap” table.• A Type1 font dictionary or CIDFontType0 CIDFont dictionary, if the embedded font program contains a “CFF ” table without CIDFont operators. In addition to the “CFF ” table, the font program must include the “cmap” table. <p>The <i>OpenType Specification</i> describes a set of required tables; however, not all tables are required in the font file, as described for each type of font dictionary that can include this entry.</p> <p>Note: The absence of some optional tables (such as those used for advanced line layout) may prevent editing of text containing the font.</p>

Page 468 and 469

Remove the paragraph that begins on the bottom of the page and continues onto the next page. This paragraph begins with “Like TrueType, OpenType font programs contain a number of tables ...”

Remove the note begins with “Other tables, such as those used for advanced line layout ...”

*The removed paragraph and the note duplicate the information provided in the revised description for the **FontFile3** key with a subtype of **OpenType**.*

Chapter 8, Interactive Features

8.2, Document-Level Navigation

Page 585

In Table 8.3, “Entries in the outline dictionary” replace the **Count** entry with the following entry:

KEY	SUBTYPE	DESCRIPTION
Count	integer	(Required) Total number of visible outline items at all levels of the outline. The value cannot be negative. This entry shall be omitted if there are no open outline items.

In Table 8.4, replace the **Count** entry with the following entry:

KEY	SUBTYPE	DESCRIPTION
Count	integer	(Required if the item has any descendents) If the outline item is open, Count is the sum of the number of visible descendent outline items at all levels. The number of visible descendent outline items is determined by the following recursive process: Step 1. Initialize Count to zero. Step 2. Add to Count the number of immediate children. During repetitions of this step, update only the Count of the original outline item. Step 3. For each of those immediate children whose Count is positive and non-zero, repeat steps 2 and 3. If the outline item is closed, Count is negative and its absolute value is the number of descendants that would be visible if the outline item were opened.

8.4, Annotations

Page 607

In Table 8.15, “Entries common to all annotation dictionaries,” the description for the **Border** entry contains a note. Replace that note with the following note:

Note: In PDF 1.2 or later, the dictionaries for some annotation types (such as free text and polygon annotations) can include the **BS** entry. That entry specifies a border style dictionary that has more settings than the array specified for the **Border** entry. If an annotation dictionary includes the **BS** entry, then the **Border** entry is ignored; see implementation note 82 in Appendix H.

Page 612

In Algorithm 8.1, replace the formula in Step 3 with the following formula:

$$AA = \text{Matrix} \times A$$

Page 622

In Table 8.24, “Additional entries specific to a link annotation,” add the following entry:

KEY	SUBTYPE	DESCRIPTION
BS	dictionary	<p>(Optional; PDF 1.6) A border style dictionary (see Table 8.17 on page 611) specifying the line width and dash pattern to be used in drawing the annotation’s border.</p> <p>The annotation dictionary’s AP entry, if present, takes precedence over the BS entry; see Table 8.15 on page 606 and Section 8.4.4, “Appearance Streams.”</p>

Page 624 and 625

In Table 8.25, “Additional entries specific to a free text annotation,” replace the Value descriptions for the entries shown in the following table. Notice that the **LE** type has changed.

KEY	TYPE	VALUE
CL	array	<p>(Optional; meaningful only if IT is FreeTextCallout; PDF 1.6) An array of four or six numbers specifying a callout line attached to the free text annotation. Six numbers $[x_1\ y_1\ x_2\ y_2\ x_3\ y_3]$ represent the starting, knee point, and ending coordinates of the line in default user space, as shown in Figure 8.4. Four numbers $[x_1\ y_1\ x_2\ y_2]$ represent the starting and ending coordinates of the line.</p>
IT	name	<p>(Optional; PDF 1.6) A name describing the intent of the free text annotation (see also the IT entry in Table 8.21). The following values shall be valid:</p> <ul style="list-style-type: none"> • FreeText, which means the annotation is intended to function as a plain free-text annotation. A plain free-text annotation is also known as a text box comment. • FreeTextCallout, which means that the annotation is intended to function as a callout. The callout is associated with an area on the page through the callout line specified in CL. See the implementation note on page 17. • FreeTextTypeWriter, which means that the annotation is intended to function as a click-to-type or typewriter object and no callout line is drawn. <p>Default value: FreeText</p>

KEY	TYPE	VALUE
LE	name	(Optional; meaningful only if CL is present; PDF 1.6) A name specifying the line ending style that shall be used in drawing the callout line specified in CL . The name specifies the line ending style for the endpoint defined by the pairs of coordinates (x_1, y_1) . Table 8.27 shows the possible line ending styles. Default value: None .
BS	dictionary	(Optional; PDF 1.6) A border style dictionary (see Table 8.17 on page 611) specifying the line width and dash pattern that shall be used in drawing the annotation's border. The annotation dictionary's AP entry, if present, takes precedence over the BS entry; see Table 8.15 on page 606 and Section 8.4.4, "Appearance Streams."

Page 641

In Table 8.39, "Additional entries specific to a widget annotation," add the following entry:

KEY	SUBTYPE	DESCRIPTION
Parent	dictionary	(Required if this widget annotation is one of multiple children in a field; absent otherwise) An indirect reference to the widget annotation's parent field. A widget annotation can have at most one parent; that is, it can be included in the Kids array of at most one field.

8.6, Interactive Forms

Page 677

Add the following paragraph before the third paragraph on this page. That paragraph begins with the phrase "Thus, all fields descended from a common ancestor..."

Because the period is used as a separator for fully qualified names, a partial name shall not contain a period.

Page 696

Replace the last sentence in the second paragraph with the following sentences. The sentence to replace begins with "Signature fields that are not intended ..."

Signature fields that are not intended to be visible should have an annotation rectangle that has zero height and width. Viewer applications must treat such signatures as not visible. Viewer applications should also treat signatures as not visible if either the **Hidden** bit or the **NoView** bit of the **F** entry is true. The **F** entry is described in Table 8.15, and annotation flags are described in Table 8.16.

Page 698

In Table 8.83, “Entries in a signature field seed value dictionary,” replace the Value descriptions for the entries shown in the following table.

KEY	TYPE	VALUE
Reasons	array	<p>(Optional) An array of text strings specifying possible reasons for signing a document. If specified, the reasons supplied in this entry replace those used by viewer applications.</p> <ul style="list-style-type: none"> • If the Reasons array is provided and the Ff entry indicates that Reasons is a required constraint, one of the reasons in the array shall be used for the signature dictionary; otherwise, signing shall not take place. If the Ff entry indicates Reasons is an optional constraint, one of the reasons in the array may be chosen or a custom reason can be provided. • If the Reasons array is omitted or contains a single entry with the value period (".") and the Ff entry indicates that Reasons is a required constraint, the Reason entry shall be omitted from the signature dictionary (see Table 8.102).
MDP	dictionary	<p>(Optional; PDF 1.6) A dictionary containing a single entry whose key is P and whose value is an integer between 0 and 3. A value of 0 defines the signature as an approval signature (previously called an <i>ordinary signature</i>) (see Section 8.7, “Digital Signatures”). The values 1 through 3 shall be used for certification signatures (previously called <i>author signatures</i>) and correspond to the value of P in a DocMDP transform parameters dictionary (see Table 8.104).</p> <p>If this entry is not present or does not contain a P entry, no rules shall be defined regarding the type of signature or its permissions.</p>

Page 699

In Table 8.83, “Entries in a signature field seed value dictionary,” the description for the **Ff** entry describes supported bit positions. Remove the value 7 (**DigestMethod**).

8.7 Digital Signatures

Beginning with PDF 1.7, the use of object digests for digital signatures is deprecated. As a result, the **UR** entry in the permissions dictionary is also deprecated. The **UR3** entry should be used in its place. The **UR** and **UR3** entries refer to a **UR** transform parameter dictionary. The **UR** transform method itself and its attendant parameter dictionary is to remain in PDF; only the specific **UR** entry in the permissions dictionary is deprecated.

The presence of a **UR** entry in the permissions dictionary requires an object digest to be computed. But the presence of a **UR3** entry in the permissions dictionary does not require an object digest; modifications in that case are detected with a byte range digest and analysis of any changes made since the signature was applied.

Page 725

Add the following paragraph after the third paragraph:

PDF 1.5 specified a method for computing an object digest over a subtree of objects in memory and storing the resulting digest in entries named **DigestValue** and **DigestLocation** in the signature

reference dictionary. (The digest algorithm was documented in Appendix I, "Computation of Object Digests.") This method is deprecated and should not be used. All mentions of object digests in Section 8.7, "Digital Signatures", should be disregarded.

Replace the bullet at the bottom of the page with the following bullet and paragraph. The bullet to replace begins with "An object digest ..."

- Modification detection shall be specified by a signature reference dictionary. The **TransformMethod** entry shall specify the general method for modification detection, and the **TransformParams** entry shall specify the variable portions of the method.

Page 726

Change the introductory phrase for the first bullet on the page to the following:

- One or more approval signatures.

Replace the second bullet on the page with the following:

- At most one certification signature (PDF 1.5). The signature dictionary of a certification signature shall be the value of a signature field and shall contain a **ByteRange** entry. It may also be referenced from the **DocMDP** entry in the permissions dictionary (see Section 8.7.3, "Permissions"). The signature dictionary shall contain a signature reference dictionary (see Table 8.103) that has a **DocMDP** transform method. See "DocMDP" on page 731 for information on how these signatures are created and validated.

A signature dictionary for a certification or approval signature may also have a signature reference dictionary with a **FieldMDP** transform method; see "FieldMDP" on page 736.

Page 730

*In Table 8.103, "Entries in a signature reference dictionary," add the following sentence to the Value descriptions for the **DigestValue**, **DigestLocation** and **DigestMethod** keys.*

These entries are part of the deprecated object digest feature and should not be used. (See implementation note 140a on [page 17](#) of this errata.)

*In Table 8.103, "Entries in a signature reference dictionary" append the following sentence to the **TransformMethod** value **Identity**.*

The **Identify** transform method is deprecated and should not be used.

Page 731

Replace the second sentence in the last paragraph on the page with the following sentences. The sentence to replace begins with "Such signatures enable detection ..."

Certification signatures use the **DocMDP** transform method to enable detection of changes the author disallows.

Page 732

Replace the note at the top of this page with the following paragraph:

A certification signature should have a legal attestation dictionary (see Section 8.7.4, "Legal Content Attestations") that specifies all content that might result in unexpected rendering of the document contents, along with the author's attestation to such content. This dictionary may be used to establish an author's intent if the integrity of the document is questioned.

Replace the section title “Validating MDP Signatures” with the following title:

Validating signatures that use the DocMDP transform method

Replace the first sentence in the section titled “Validating MDP signatures” with the following sentence:

To validate a signature that uses the DocMDP transform method, an application shall first verify the byte range digest. Next, it shall verify that any modifications that have been made to the document are permitted by the transform parameters.

Add the following sentence at the end of the first bullet. That bullet begins with “PDF 1.5 required the calculated value ...”

This validation method is deprecated and should not be used.

Page 733

At the beginning of the section titled UR, add the following note after the first sentence:

Note: The use of a **UR** entry in a permissions dictionary is deprecated. Usage rights should be specified only with a **UR3** entry.

Page 735

In the **Form** entry in Table 8.105, “Entries in the UR transform parameters dictionary,” add the following names to the list of valid names defined in PDF 1.5:

Add	Permits the user to add form fields to the document
Delete	Permits the user to delete form fields from the document

Add the following sentence to the Value description for the **FormEx** entry in Table 8.105, “Entries in the UR transform parameters dictionary”:

This entry is deprecated and should not be used.

Page 737

Replace the first bullet on this page with the following:

- The author specifies that form fields can be filled in without invalidating the approval or certification signature.

Add the following sentence to the description for the **Identity** transform method:

This transform method is deprecated and should not be used.

Page 741

Add the following sentence to the Value description for the **UR** entry in Table 8.107, “Entries in a permissions dictionary”.

Using the **UR** entry to specify the signature dictionary is deprecated and should not be used; the **UR3** entry should be used instead.

Page 742

In the second and third paragraphs on this page, change the term author signatures to certification signature.

Chapter 10, Document Interchange

Section 10.7, Tagged PDF

Page 907

Replace the paragraph that begins with the phrase, “A link element may contain several link annotations ...” with the following paragraphs, example, and note.

When a **Link** structure element describes a span of text to be associated with a link annotation and that span wraps from the end of one line to the beginning of another, the **Link** structure element shall include a single object reference that associates the span with the associated link annotation. Further, the link annotation shall use the **QuadPoint** entry to denote the active areas on the page.

Figure 9.19 Wrapping active area associated with a link annotation

Here is some text [with a link](#) inside.

In the above example, the **Link** structure element references a link annotation that includes a **QuadPoint** entry that boxes the strings “with a” and “link”. That is, the **QuadPoint** entry contains 16 numbers: the first 8 numbers describe a quadrilateral for “with a”, and the next 8 describe a quadrilateral for “link.”

Note: Beginning with PDF 1.7, use of the **Link** structure element to enclose multiple link annotations is deprecated.

Section 10.10, Prepress Support

Page 970

The second sentence in the second paragraph in section 10.10.4, “Output Intents,” begins with “For example, one production ...” Replace that sentence with the following sentence:

For example, one production facility might process files conforming to a recognized standard such as PDF/X-1, while another uses the PDF/A standard to produce RGB output for document distribution on the Web.

Replace the last paragraph on the page with the following paragraph. That paragraph continues onto page 971.

The following output intent subtypes are defined:

- GTS_PDFX corresponds to the PDF/X format standard specified in ISO 15930 (see the Bibliography).
- GTS_PDFA1 corresponding to the PDF/A format standard as defined by ISO 19005 (see the Bibliography)
- ISO_PDFE1 corresponding to the PDF/E format standard as defined by ISO 24517 (see the Bibliography)

Table 10.51 shows the contents of this type of output intent dictionary. Other subtypes may be added in the future; the names of any such additional subtypes shall conform to the naming guidelines described in Appendix E.

Page 971

In Table 10.51, “Entries in a PDF/X output intent dictionary,” replace the description for the **S** entry shown in the following table.

KEY	SUBTYPE	DESCRIPTION
S	name	(Required) The output intent subtype. Supported values shall be GTS_PDFX , GTS_PDFA1 or ISO_PDFE1 .

Appendix C, Implementation Limits

Page 991

Append the following sentence to the paragraph that appears after the bullets and that begins “PDF itself has one architectural limit...”:

This limit does not apply in a PDF file that uses a cross-reference stream (see 3.4.7, “Cross- Reference Streams”) instead of a cross reference table.

Appendix H, Compatibility and Implementation Notes

Page 1114

After the section titled, 8.4.5, “Annotation Types” (Link Annotations), add the following section heading and implementation note:

8.4.5, “Annotation Types” (Free Text Annotations)

In Acrobat 7 and later versions, a failure may occur when Acrobat re-draws a free text annotation whose **IT** entry has a value of **FreeTextCallout** and whose **CL** entry is omitted.

Page 1121

Add the following implementation note, after implementation note 140:

140a. Acrobat 7 and Acrobat 8 require the **DigestValue** and **DigestLocation** entries to be present, with dummy values that are not used. This requirement is eliminated in Acrobat 8.1 and later.

Appendix I, Computation of Object Digests

Page 1131

Add the following sentence after the first sentence in this appendix:

This method for detecting modifications is deprecated and should not be used. Additionally, the description of the algorithm is known to contain significant errors.

Bibliography

Page 1152

Replace the document title OpenType Font Specification *with* OpenType Specification v.1.4.

Page 1154

Add the qualifier 1st Edition to the ECMA-363 specification, as follows:

Ecma International, Standard ECMA-363, *Universal 3D File Format, 1st Edition*. This document is available at www.ecma-international.org.

Add the following entry:

ANSI X3.4-1986, *Information Systems -- Coded Character Sets 7-Bit American National Standard Code for Information Interchange (7-Bit ASCII)*

Page 1156

Add the following entry to the list of specifications available on International Organization for Standardization (ISO):

ISO 19005-1:2005, *Document management -- Electronic document file format for long-term preservation -- Part 1: Use of PDF 1.4 (PDF/A-1)*

ISO/PRF 24517-1, *Document management -- Engineering document format using PDF -- Part 1: Use of PDF 1.6 (PDF/E-1)*