



# PDFTron PDF/A Manager<sup>TM</sup>

## User Manual

**Version 1.x**

**LEGAL STATEMENT AND COPYRIGHT NOTICE**

PDFTron PDF/A Manager™ User Manual

Part number: PDFTRON-1-PDFAManagerCMD

Part number: PDFTRON-1-PDFAManagerSDK

Last Updated: February 1, 2010

© 2009-2010 PDFTron Systems, Inc. All Rights Reserved.

All information contained herein is the property of PDFTron Systems, Inc. ("PDFTron"). No part of this publication (whether in hardcopy or electronic form) may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of PDFTron Systems, Inc..

The information in this publication is provided for informational use only, is subject to change without notice, and should not be construed as a commitment by PDFTron. PDFTron assumes no responsibility or liability for any loss or damage that may arise from the use of any information in this publication. The software described in this user manual is furnished under License (enclosed in the software package) and may only be used or copied in accordance with the terms of that License.

PDFTron and the names of PDFTron products referenced herein are either trademarks and/or service marks and/or registered trademarks of PDFTron Systems, Inc. PDFTron, PDFNet SDK, PDF/A Manager, PDF2Image, PDF2SVG, PDF2Text, PDF2XPS, XPSConvert, PDFTron PDFSecure, PDF PageMaster, CosEdit, PDFNet SDK, PDF2Image SDK, PDF2SVG SDK, PDF2Text SDK, PDF2XPS SDK, XPSConvert SDK, PDFSecure SDK, PDF PageMaster SDK and associated Logos are either trademarks and/or service marks and/or registered trademarks of PDFTron Systems, Inc.

Any other brand or product names mentioned in this publication are the registered trademarks or trademarks of their respective holders. Mention of a product in this document does not necessarily imply endorsement of the product.

## TABLE OF CONTENTS

---

<b>Legal Statement and Copyright Notice</b>	<b>2</b>
<b>1. Introduction</b>	<b>5</b>
<b>1.1 An Introduction to PDFTron PDF/A Manager</b>	<b>5</b>
<b>1.1.1 Key Functions</b>	<b>5</b>
<b>1.1.2 Benefits - Why PDF/A Manager?</b>	<b>5</b>
<b>1.1.4 Operating Systems Supported</b>	<b>6</b>
<b>1.1.5 System Requirements</b>	<b>6</b>
<b>1.2 About This Manual</b>	<b>6</b>
<b>2. Installing and Uninstalling PDF/A Manager</b>	<b>7</b>
<b>2.1 PDF/A Manager Installation</b>	<b>7</b>
<b>2.2 Product Registration</b>	<b>7</b>
<b>2.3 Demo Version Installation</b>	<b>8</b>
<b>2.4 Uninstalling PDF/A Manager</b>	<b>8</b>
<b>3. Overview</b>	<b>9</b>
<b>3.1 Basic Syntax</b>	<b>9</b>
<b>3.2 Command-Line Summary</b>	<b>9</b>
<b>3.3 Basic Usage</b>	<b>11</b>
<b>3.3.1 How to save converted PDF/A files in a given folder?</b>	<b>11</b>
<b>3.3.2 How can I control the output name for converted PDF/A files?</b>	<b>11</b>
<b>3.3.3 How do I run PDF/A Manager in validation mode?</b>	<b>11</b>
<b>3.3.4 How do I convert/validate a password protected PDF?</b>	<b>11</b>
<b>3.3.5 How do I process PDF files in a batch?</b>	<b>12</b>
<b>3.3.6 Does PDF/A Manager have any dependencies on third party components/software?</b>	<b>13</b>
<b>3.3.7 How do I web-optimize PDF/A files for fast web or network access?</b>	<b>13</b>
<b>3.3.8 What is the difference between PDF/A-1b and PDF/A-1a?</b>	<b>13</b>
<b>3.3.9 Can I integrate PDF/A Manager with my client/server application?</b>	<b>13</b>
<b>3.3.10 I didn't find the answer to my question in the user manual. Are there any other helpful resources?</b>	<b>13</b>
<b>3.4 Exit Codes</b>	<b>14</b>
<b>4. PDF/A Manager Reporting</b>	<b>15</b>
<b>4.1 PDF/A Manager XML Report Format</b>	<b>15</b>
<b>4.2 PDF/A Manager Validation and Conversion Error Codes</b>	<b>17</b>
<b>4.2.1 PDF/A Level B Validation Errors</b>	<b>17</b>
<b>4.2.2 PDF/A Level A Validation Errors</b>	<b>23</b>
<b>5. Overview of PDF/A Manager SDK</b>	<b>24</b>
<b>5.1 Working with PDF/A Manager SDK</b>	<b>24</b>

5.2	Reporting Progress Messages and Errors	26
5.3	PDF/A Add-on module for PDFNet SDK	27
6.	Support	29
6.1	Reporting Problems	29
6.2	Contact Information	29

## 1. Introduction

### 1.1 An Introduction to PDFTron PDF/A Manager

**PDF/A Manager** is a PDF/A (ISO 19005-1) validation and conversion software. It is available as a command-line tool and as a development toolkit.

The conversion option analyses the content of existing PDF files and performs a sequence of modifications in order to produce a PDF/A compliant document. Features that are not suitable for long-term archiving (such as encryption, obsolete compression schemes, missing fonts, or device-dependent color) are replaced with their PDF/A compliant equivalents. Because the conversion process applies only necessary changes to the source file, the information loss is minimal. Also, because the converter provides a detailed report for each change, it is simple to inspect changes and to determine whether the conversion loss is acceptable.

The validation option in PDF/A Manager can be used to quickly determine whether a PDF file fully complies with the PDF/A specification according to the international standard ISO 19005-1. For files that are not compliant, the validation option can be used to produce a detailed report of compliance violations as well as a list of relevant error objects.

Like other PDFTron products, PDF/A Manager does not rely on other third-party software. PDF/A Manager can be used in server environments or as a batch conversion process. PDF/A Manager Command-Line utility is based on PDFNet SDK core technology which is available for integration in third party solutions and applications. For more information about PDFNet SDK, please contact a PDFTron representative or visit <http://www.pdftron.com/pdfnet> for more information.

#### 1.1.1 Key Functions

- Checks if a PDF file is compliant with PDF/A (ISO 19005-1) specification.
- Converts any PDF to a PDF/A compliant document.
- Supports both PDF/A-1a and PDF/A-1b.
- Produces a detailed report of compliance violations and associated PDF objects.
- Keeps the required changes a minimum, preserving the consistency of the original.
- Tracks all changes to allow for automatic assessment of data loss.
- Allows user to customize compliance checks or omit specific changes during PDF/A conversion.
- Preserves tags, logical structure, and color information in existing PDF documents.
- Offers automatic font substitution, embedding, and subsetting options.
- Supports automation and batch operation. PDF/A Manager is designed to be used in unattended mode in high throughput server or batch environments.

#### 1.1.2 Benefits - Why PDF/A Manager?

- PDF/A Manager is an indispensable tool for anyone working with PDF/A.
- Supports all versions of the PDF format (PDF 1.0 to ISO32000).
- Includes an option to create linearized (web-optimized) PDF/A documents.
- Fast, reliable and suitable for server use.

#### 1.1.3 Common Use Case Scenarios

- Libraries, newspaper agencies, and government institutions can use PDF/A Manager to automate PDF archiving in high-throughput web applications of server based environments.
- Maintainers or large collections of existing PDF documents can use PDF/A Manager in batch mode to ensure that all files are suitable for long-term archiving.

- Software developers can use PDF/A Manager SDK or PDFNet SDK API to add the PDF/A Export option to their existing PDF export filter.

#### **1.1.4 Operating Systems Supported**

- Windows 7, 2008, Vista, XP, 2003, 2000, NT, 98
  - Mac OSX
  - Linux

### **1.1.5 System Requirements**

- At least 20 MB of free disk space.
  - Memory requirement is heavily dependent on the nature of the document(s) being processed.

## 1.2 About This Manual

This manual is intended as a guide to the installation and use of PDF/A Manager. It is intended for system administrators, programmers and other users who are familiar with PDF documents, graphic file manipulation and general computer processes.

- [Section 1](#) introduces PDF/A Manager and describes the manual.
  - [Section 2](#) explains how to install and uninstall PDF/A Manager.
  - [Section 3](#) summarizes the command-line arguments available for PDF/A Manager and covers basic usage.
  - [Section 4](#) covers PDF/A Manager reporting.
  - [Section 5](#) contains an overview and basic usage of PDF/A Manager SDK.
  - [Section 6](#) is where you will find all the support information you may require, such as how to report a problem with the software.

## 2. Installing and Uninstalling PDF/A Manager

### 2.1 PDF/A Manager Installation

PDF/A Manager is supplied as a download from a distributor or directly from [www.pdftron.com](http://www.pdftron.com). The release is packaged as a .zip file ('pdfa.zip' or 'pdfa\_sdk.zip'). To install the software, simply unzip the archive in the desired location while making sure to preserve the directory (folder) structure during the process.

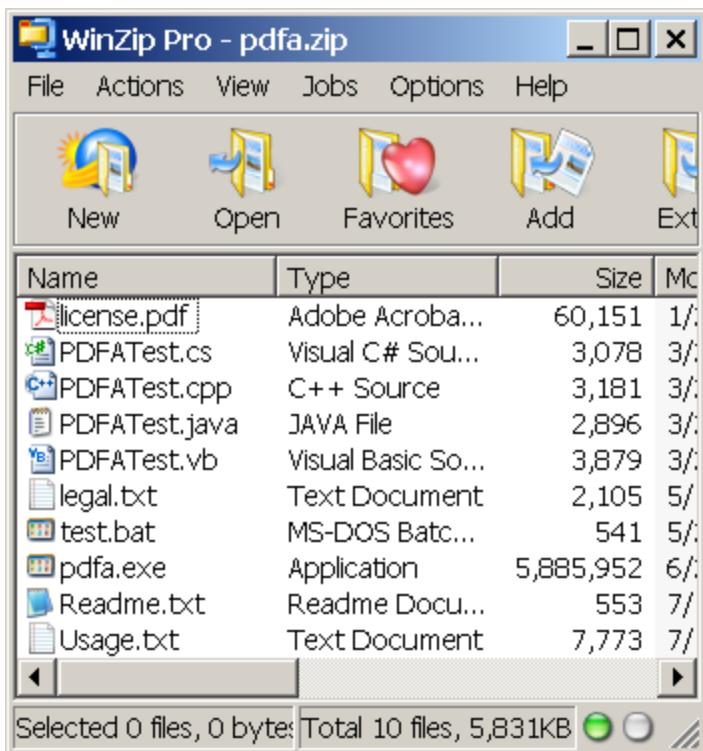


Figure 2.1  
Extracting PDF/A Manager Archive  
using WinZip

### 2.2 Product Registration

After purchasing a license of PDF/A Manager, you will receive additional registration and license information via email.

If you purchased the PDF/A Manager Command-line Application, you just need to copy the license file provided to you into the "pdfa" folder to register the software.

For PDF/A Manager SDK, you need to pass the license information during the call to PDFAInit() to successfully register the SDK and remove evaluation restrictions. For example:

```
PDFAInit("Joe Doe", "MyCompany", "sRLL4qr555sd63dnd");
```

The first parameter is your user name, the second parameter is your company name, and the last parameter is the license key supplied with your registration information.

## 2.3 Demo Version Installation

If you wish to evaluate the product, you can download the demo version of the product without any serial number or license key.

To do this, go to PDFTron's **Downloads** page at [www.pdftron.com/downloads.html](http://www.pdftron.com/downloads.html). Click on the appropriate product version/name, which will bring you to the product and the appropriate link for the demo download. Simply Download the zip file ('pdःa.zip' or 'pdःa\_sdk.zip') and extract the archive in the desired location, while making sure to preserve the directory (folder) structure when extracting the archive. This will provide you with a working copy of the software. The limitation of the evaluation version is that all pages in processed documents will have a demo stamp.

## 2.4 Uninstalling PDF/A Manager

To remove PDF/A Manager from a computer, simply delete the "pdःa" folder (for the command-line application) or the "pdःa\_sdk" folder (for the SDK).

### 3. Overview

PDFTron's PDF/A Manager is a command-line application designed to validate and convert generic PDF documents to PDF/A (ISO 19005-1) compliant format. This section covers basic usage and the command string syntax, used both in the PDF/A Manager Command-Line Application as well as in PDF/A Manager SDK.

#### 3.1 Basic Syntax

The basic command-line syntax is:

```
pdaf [options] file1 file2 folder1 file3 ...
```

#### 3.2 Command-Line Summary

Option	Parameter	Description
-x or --noxml	--noxml	<p>Disable XML report generation.</p> <p>By default, PDF/A Manager will generate an XML report (report.xml) in the output folder with all of the details of the validation and conversion results. The XML file can be viewed within most modern web-browsers.</p> <p>If this parameter is specified, validation and conversion results will be reported as part of the console output.</p>
-o or --output	e.g. -o myfolder -o c:\foo -o ../../bar	The output directory. The folder can be relative to the current working folder. If the folder does not exist, PDF/A Manager will attempt to create the required path. If this parameter is not specified, all files will be saved relative to the current working folder.
--subfolders		Process all sub-directory for every directory specified in the argument list. By default, sub-directories are not processed.
-f or --fname		The explicit name for the converted file. The parameter is used only when converting one file at a time. The output file will be stored in the output folder.
-p or --pass	-p "my pass"	The password to process the input file. The password is required only in the conversion mode and when the input document is encrypted.
--suffix	--suffix "_PDF_A"	A string to append to all output filenames. This option is typically used to tag output filenames during multi-file or batch conversion. For example, if input file is "my.pdf" and suffix is "_pdaf" the output filename will be "my_pdaf.pdf".
-l or --level	-l B	<p>PDF/A Conformance Level: [A or B]</p> <p>Level A conforming files must adhere to all of the requirements of ISO 19005. A file meeting this conformance level is said to be a 'conforming PDF/A-1a file.'</p>

		<p>Level B conforming files shall adhere to all of the requirements of ISO 19005 except those of 6.3.8 and 6.8. A file meeting this conformance level is said to be a 'conforming PDF/A-1b file'. The Level B conformance requirements are meant to support archiving by preserving exact rendered visual appearance of the document.</p> <p>Default Conformance Level is 'B'.</p>
-c or --convert	-c	Convert input PDF files to PDF/A compliant format. If this parameter is not specified, PDF/A manager will work in PDF/A validation mode without producing or creating any PDF output.
--nr		Do not revalidate after conversion. By default, PDF/A Manager will perform additional validation after conversion that can catch any omissions in the conversion process. Specifying this option skips this step.
-z or --linearize		Web-optimize (linearize) the converted PDF/A documents.
--extension	--extension ".mypdf"	The default file extension used to process PDF documents. The default extension is ".pdf".
--verb	--verb 3	<p>Set the verbosity level. The number argument is in the range 0-3 and it can be used to control the amount of information emitted during conversion.</p> <p>0 – silent 1 – normal 2 – debug</p>
-v or --version		Print the version information.
-h or --help		Print a listing of available options.

### 3.3 Basic Usage

This section covers the basic usage of PDF/A Manager, explaining all the available options and illustrating examples.

#### 3.3.1 How to save converted PDF/A files in a given folder?

By default, PDF/A Manager stores converted files in the current working folder. To specify another output location, use the '-o' (or --output) parameter. For example:

```
pdafa -o "c:\My Output" --convert 1.pdf 2.pdf 3.pdf
```

Note: If the specified path does not exist, PDF/A Manager will attempt to create the necessary folders.

#### 3.3.2 How can I control the output name for converted PDF/A files?

By default, PDF/A Manager will create a single output PDF/A document based on every input PDF file. The output filename is constructed by appending a suffix to the name of the input PDF file. For example, 'my.pdf' will be saved as 'my\_pdfa.pdf'. It is possible to change the suffix using --suffix option. For example:

```
pdafa -o OUT --convert --suffix "converted" "c:\mypdfdir"
```

#### 3.3.3 How do I run PDF/A Manager in validation mode?

To run PDF/A Manager in validation mode simply omit '-c' or '--convert' from the command line. For example,

```
pdafa --noxml my.pdf
```

#### 3.3.4 How do I convert/validate a password protected PDF?

PDF/A Manager will, without user intervention, decrypt and convert documents secured with a master/owner password. If the document is secured using a user (or 'file open') password, PDF/A Manager will prompt you to enter the password.

For unattended conversion, the password can also be specified directly on the command-line using the '-p' (or --password) option. For example:

```
pdafa -p secret -c secured.pdf
```

The above command line will convert PDF to PDF/A and will use the provided password ('secret') to open the secured document (i.e. 'secured.pdf').

Note: PDF/A Manager supports all standard security options available in PDF, including 40 and 128 bit RC4 encryption, Crypt filters, and 128 AES (Advanced Encryption Standard) encryption.

### 3.3.5 How do I process PDF files in a batch?

PDF/A Manager supports batch conversion and validation of many PDF files in a single pass. To convert all PDF files in a given folder(s) you can use the following syntax:

```
pdfa myfolder1 myfolder2 ...
```

The '--subfolders' option can be used to recursively process all subfolders. For example, the following line will convert all documents in 'myfolder1' and 'myfolder2' as well as all subfolders:

```
pdfa --subfolders myfolder1 myfolder2
```

By default, PDF/A Manager will convert all files with the extension '.pdf'. To select different files based on the extension use the '--extension' parameter. For example, to convert all PDF documents with a custom extension '.blob', you could use the following line:

```
pdfa --extension .blob --subfolders myfolder1
```

Wildcard characters can also be used to process multiple input files.

For example, if a directory contains the following PDF documents:

```
C:\test1 >dir
Directory of C:\test1
01/04/2009  03:35 PM    <DIR>      .
01/04/2009  03:35 PM    <DIR>      ..
05/21/2009  02:27 PM      A1.pdf
05/03/2009  09:38 AM      A2.pdf
05/20/2009  08:46 AM      B1.pdf
05/15/2009  12:50 PM      B2.pdf
```

To process all PDF documents in this folder, you could specify:

```
c:\>pdfa --noxml c:/test1/*.pdf
```

To process all PDF documents starting with 'A', you could specify:

```
pdfa --noxml c:/test1/A*.pdf
```

Or to process all PDF documents ending with '1', you could specify:

```
pdfa --noxml c:/test1/*1.pdf
```

You can use either of the two standard wildcards — the question mark (?) and the asterisk (\*) — to specify filename and path arguments on the command line.

The wildcards are expanded in the same manner as operating system commands. (Please refer to your operating system user's guide if you are unfamiliar with wildcards). Enclosing an argument in double quotation marks (" ") suppresses the wildcard expansion. Within quoted arguments, you can represent quotation marks literally by preceding the double-quotation-mark character with a backslash (\). If no matches are found for the wildcard argument, the argument is passed literally.

### **3.3.6 Does PDF/A Manager have any dependencies on third party components/software?**

PDF/A Manager is a completely stand alone application and does not include any dependencies on third-party components or software.

### 3.3.7 How do I web-optimize PDF/A files for fast web or network access?

PDF/A Manager can linearize output PDF documents for fast web or network access. This can greatly improve document loading time if you plan to share documents on the web or local network. To linearize (i.e web-optimize) all output documents add -l or --linearize option in the command-line.

### **3.3.8 What is the difference between PDF/A-1b and PDF/A-1a?**

PDF/A-1b is a subset of PDF/A-1a specification. As a result every PDF/A-1a file is also a valid PDF/A-1b document. The main difference is that PDF/A-1a additionally requires that PDF specifies logical structure as well as Unicode mapping for all text in the document.

The primary purpose of logical structure is to offer accessibility features for visually impaired people (similar to 'Alt' tags in HTML).

Compared to PDF/A-1b there is no well defined process to validate or convert PDF/A-1a (besides a handful of simple checks). Because the semantic information may be missing from input document and because there is no universally agreed way to reconstruct or to represent this information the usefulness of PDF/A-1a is open to debate.

By default PDF/A Manager will preserve tags, logical structure, and color information in input PDF files.

### **3.3.9 Can I integrate PDF/A Manager with my client/server application?**

PDF/A Manager can be easily integrated into third-party client and server-based applications.

Besides command-line utility PDF/A Manager is part of PDFNet SDK and can be accessed from any programming language (including C#, VB.Net, C/C++, Java, VB6, Perl, Python, Ruby, Delphi, etc).

### **3.3.10 I didn't find the answer to my question in the user manual. Are there any other helpful resources?**

You may want to search the PDF/A Manager Knowledge Base forum (which can be accessed via PDFTron's website at: [www.pdftron.com](http://www.pdftron.com)) or to simply forward your question to PDFTron's technical support team via [support@pdftron.com](mailto:support@pdftron.com).

### 3.4 Exit Codes

To provide additional feedback, PDF/A Manager returns exit codes after completing processing. The exit codes can be used to provide user feedback for logging, etc. This is particularly important for applications running in an unattended environment.

The following table lists possible exit codes and their description:

Exit Code	Description
0	All files converted successfully.
1	Document is secured. Need a valid password to open the document.
2	Error opening the input file(s).
3	An unknown exception encountered.

All codes other than '0' indicate that there was an error during the conversion process.

The following illustrates a sample Windows batch script that processes exit codes:

```
@echo off rem convert all PDF files in 'data' folder

pdfa ./data
if errorlevel 1 goto passwd
if errorlevel 2 goto inputerr
if errorlevel 3 goto othererror
if errorlevel 0 goto exit

:passwd
echo Document is protected. Need a valid password to open the document.
goto exit

:inputerr
echo No input files specified.
goto exit

:othererror
echo An error encountered during processing.
goto exit

:exit
```

## 4. PDF/A Manager Reporting

This section covers PDF/A Manager report formats and all the possible error conversion and validation options.

### 4.1 PDF/A Manager XML Report Format

By default PDF/A Manager stores details of validation and conversion processing in an XML file. The report is saved as 'report.xml' in the output directory. Similarly, the style sheet for the report is saved as 'report.xsl' in the output directory. Double clicking on 'report.xml' will launch a web browser display HTML report as shown in the following figure:

**Figure 4.1**  
**Typical PDF/A Manager Report**

**PDF/A Validation Results (Before Conversion):**

Run DateTime: 2008-Jul-13 10:53:45

**Fail:**

license.pdf		
Error ID	Message	Obj Refs
e_PDFA185	'obj' not followed by EOL marker	1, 2, 3, 4, 5, 6, 7, 8, 9, 12
e_PDFA341	The font is not embedded	16, 17, 18
e_PDFA723	The XMP Metadata stream is not valid	39
e_PDFA2331	Device-specific color space used, but no GTS_PDFA1 OutputIntent	1, 4, 14

Manual.pdf		
Error ID	Message	Obj Refs
e_PDFA14	Contains compressed object streams	179, 694
e_PDFA42	Transparency used (XObject with soft mask)	21, 75, 81
e_PDFA183	Object number not preceded by EOL marker	1162
e_PDFA185	'obj' not followed by EOL marker	1, 130, 131, 132, 133, 134, 141, 145, 148, 156
e_PDFA244	Image with Interpolate key set to true	9
e_PDFA341	The font is not embedded	5, 7, 15, 108, 118
e_PDFA354	The font descriptor dictionary does not include a CIDSet stream for CIDFont subset	101
e_PDFA723	The XMP Metadata stream is not valid	1164
e_PDFA2331	Device-specific color space used, but no GTS_PDFA1 OutputIntent	3, 17, 23, 63, 71, 77, 79, 85, 91, 93

**Pass:**

---

**PDF/A Revalidation Results (After Conversion):**

Run DateTime: 2008-Jul-13 10:53:45

**Fail:**

**Pass:**

license_pdfa.pdf
Manual_pdfa.pdf

Generated using PDFTron PDF/A Manager V1.00.

Done My Computer 100% 15

The report includes the following information:

- The date and time when the report was generated
- A listing of files that fail validation prior to conversion. The information in this table includes the input PDF filename, error code, error message, and a list of relevant PDF object numbers.
- A listing of files that pass validation and for which no conversion is necessary.
- A listing of files that fail validation after conversion. Usually this list includes files that can't be fully converted to PDF/A without significant information loss.
- A final list of files that pass validation after conversion.

The general structure of XML report is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<?xmlstylesheet type="text/xsl" href="report.xsl"?>
<PDFAManagerReport RunDateTime="2009-Jul-16 22:46:04">
  <Validation>
    <Fail FileName="my.pdf" FileNameAndPath="c:\my.pdf">
      <Error Code="e_PDFA232" Message="Not a valid ICC color profile"
        Refs="3"/>
    </Fail>
  </Validation>
  <Validation>
    <Fail FileName="license.pdf" FileNameAndPath="c:\ license.pdf">
      <Error Code="e_PDFA185" Message="'obj' not followed by EOL marker"
        Refs="1, 2, 3, 4, 5, 6, 7, 8, 9, 12"/>
      <Error Code="e_PDFA341" Message="The font is not embedded"
        Refs="16, 17, 18"/>
      <Error Code="e_PDFA353" Message="Embedded TrueType font program..."
        Refs="16, 17, 18"/>
      <Error Code="e_PDFA723" Message="The XMP Metadata stream is not
        valid" Refs="39"/>
      <Error Code="e_PDFA2331" Message="Device-specific color space used,
        but no GTS_PDFA1 OutputIntent" Refs="1, 4, 14"/>
    </Fail>
  </Validation>
</PDFAManagerReport>
```

To disable report generation, simply include '--noxml' in the command-line string. Disabling XML report generation will direct processing results to the console window.

```
c:\PDFA\>pdfa --noxml -c -o OUT my.pdf
Processing...
VLD-[FAIL]: my.pdf
  - e_PDFA185: 'obj' not followed by EOL marker
    Obj Refs:1, 2, 3, 4, 5, 6, 7, 8, 9, 12
  - e_PDFA341: The font is not embedded
    Obj Refs:16, 17, 18
  - e_PDFA723: The XMP Metadata stream is not valid
    Obj Refs:39
  - e_PDFA2331: Device-specific color space used, but no GTS_PDFA1 Output Intent
    Obj Refs:1, 4, 14
CNV-[PASS]: my_pdfa.pdf
```

## 4.2 PDF/A Manager Validation and Conversion Error Codes

PDF/A Manager analyzes the content of existing PDF files and performs a sequence of modifications in order to produce a PDF/A compliant document. Features that are not suitable for long-term archiving (such as encryption, obsolete compression schemes, missing fonts, or device-dependent color) are replaced with their PDF/A compliant equivalents.

### 4.2.1 PDF/A Level B Validation Errors

The following table is a non-exhaustive list of checks and conversions performed by PDF/A Manager for Level B Validation:

Error Code	Description
e_PDFA0_1_1	Corrupt document
e_PDFA0_1_2	Corrupt content stream.
e_PDFA0_1_3	Using JPEG2000 compression (PDF 1.4 compatibility).
e_PDFA0_1_4	Contains compressed object streams (PDF 1.4 compatibility).
e_PDFA0_1_5	Contains cross-reference streams (PDF 1.4 compatibility).
e_PDFA1_2_1	Document does not start with % character.
e_PDFA1_2_2	File header line not followed by % and 4 characters > 127.
e_PDFA1_3_1	The trailer dictionary does not contain ID.
e_PDFA1_3_2	Trailer dictionary contains Encrypt.
e_PDFA1_3_3	Data after last EOF marker.
e_PDFA1_3_4	Linearized file: ID in 1st page and last trailer are different.
e_PDFA1_4_1	Subsection header: starting object number and range not separated by a single space.
e_PDFA1_4_2	'xref' and cross reference subsection header not separated by a single EOL marker.
e_PDFA1_6_1	Invalid hexadecimal strings used.
e_PDFA1_7_1	The 'stream' token is not followed by CR and LF or a single LF.
e_PDFA1_7_2	The 'endstream' token is not preceded by EOL.
e_PDFA1_7_3	The value of Length does not match the number of bytes.
e_PDFA1_7_4	A stream object dictionary contains the F, FFilter, or FDecodeParms keys.
e_PDFA1_8_1	Object number and generation number are not separated by a single

	white-space.
e_PDFA1_8_2	Generation number and 'obj' are not separated by a single white-space.
e_PDFA1_8_3	Object number not preceded by EOL marker
e_PDFA1_8_4	'endobj' not preceded by EOL marker
e_PDFA1_8_5	'obj' not followed by EOL marker
e_PDFA1_8_6	'endobj' not followed by EOL marker
e_PDFA1_10_1	Using LZW compression.
e_PDFA1_11_1	A file specification dictionary contains the EF key.
e_PDFA1_11_2	Contains the EmbeddedFiles key
e_PDFA1_12_1	Array contains more than 8191 elements
e_PDFA1_12_2	Dictionary contains more than 4095 elements
e_PDFA1_12_3	Name with more than 127 bytes
e_PDFA1_12_4	Contains an integer value outside of the allowed range
e_PDFA1_12_5	Exceeds the maximum number (8,388,607) of indirect objects in a PDF file.
e_PDFA1_12_6	The number of nested q/Q operators is greater than 28
e_PDFA1_13_1	Optional content (layers) not allowed.
e_PDFA2_2_1	DestOutputProfile-s in OutputIntents array do not match.
e_PDFA2_3_2	Not a valid ICC color profile.
e_PDFA2_3_3	The N entry does not match the number of color components in the embedded ICC profile.
e_PDFA2_3_3_1	Device-specific color space used, but no GTS_PDFA1 OutputIntent.
e_PDFA2_3_3_2	Device-specific color space, does not match OutputIntent.
e_PDFA2_3_4_1	Device-specific color space used in an alternate color space.
e_PDFA2_4_1	Image with Alternates key.
e_PDFA2_4_2	Image with OPI key.
e_PDFA2_4_3	Image with invalid rendering intent.
e_PDFA2_4_4	Image with Interpolate key set to true

e_PDFA2_5_1	XObject with OPI key.
e_PDFA2_5_2	PostScript XObject.
e_PDFA2_6_1	Contains a reference XObject.
e_PDFA2_7_1	Contains an XObject that is not supported (e.g. PostScript XObject)
e_PDFA2_8_1	Contains an invalid Transfer Curve in the extended graphics state.
e_PDFA2_9_1	Use of an invalid rendering intent
e_PDFA2_10_1	Illegal operator.
e_PDFA3_2_1	Embedded font is damaged.
e_PDFA3_3_1	Incompatible CIDSystemInfo entries
e_PDFA3_3_2	Type 2 CIDFont without CIDToGIDMap
e_PDFA3_3_3_1	CMap not embedded
e_PDFA3_3_3_2	Inconsistent WMode in embedded CMap dictionary and stream.
e_PDFA3_4_1	The font is not embedded.
e_PDFA3_5_1	Embedded composite (Type0) font program does not define all font glyphs.
e_PDFA3_5_2	Embedded Type1 font program does not define all font glyphs.
e_PDFA3_5_3	Embedded TrueType font program does not define all font glyphs.
e_PDFA3_5_4	The font descriptor dictionary does not include a CIDSet stream for CIDFont subset.
e_PDFA3_5_5	The font descriptor dictionary does not include a CharSet string for Type1 font subset.
e_PDFA3_6_1	Widths in embedded font are inconsistent with /Widths entry in the font dictionary.
e_PDFA3_7_1	A non-symbolic TrueType font must use WinAnsiEncoding or MacRomanEncoding.
e_PDFA3_7_2	A symbolic TrueType font must not specify encoding.
e_PDFA3_7_3	A symbolic TrueType font does not have exactly one entry in cmap table.
e_PDFA4_1	Transparency used (ExtGState with soft mask).
e_PDFA4_2	Transparency used (XObject with soft mask).

e_PDFA4_3	Transparency used (Form XObject with transparency group).
e_PDFA4_4	Transparency used (Blend mode is not 'Normal').
e_PDFA4_5	Transparency used ('CA' value is not 1.0).
e_PDFA4_6	Transparency used ('ca' value is not 1.0).
e_PDFA5_2_1	Unknown annotation type.
e_PDFA5_2_2	FileAttachment annotation is not permitted.
e_PDFA5_2_3	Sound annotation is not permitted.
e_PDFA5_2_4	Movie annotation is not permitted.
e_PDFA5_2_5	Redact annotation is not permitted.
e_PDFA5_2_6	3D annotation is not permitted.
e_PDFA5_2_7	Caret annotation is not permitted.
e_PDFA5_2_8	Watermark annotation is not permitted.
e_PDFA5_2_9	Polygon annotation is not permitted.
e_PDFA5_2_10	PolyLine annotation is not permitted.
e_PDFA5_2_11	Screen annotation is not permitted.
e_PDFA5_3_1	An annotation dictionary contains the CA key with a value other than 1.0.
e_PDFA5_3_2_1	An annotation dictionary is missing F key.
e_PDFA5_3_2_2	An annotation's 'Print' flag is not set
e_PDFA5_3_2_3	An annotation's 'Hidden' flag is set.
e_PDFA5_3_2_4	An annotation's 'Invisible' flag is set.
e_PDFA5_3_2_5	An annotation's 'NoView' flag is set.
e_PDFA5_3_3_1	An annotation's C entry present but no OutputIntent present
e_PDFA5_3_3_2	An annotation's C entry present but OutputIntent has non-RGB destination profile
e_PDFA5_3_3_3	An annotation's IC entry present but no OutputIntent present
e_PDFA5_3_3_4	An annotation's IC entry present and OutputIntent has non-RGB destination profile
e_PDFA5_3_4_1	An annotation AP dictionary has entries other than the N entry.

e_PDFA5_3_4_2	An annotation AP dictionary does not contain N entry
e_PDFA5_3_4_3	AP has an N entry whose value is not a stream
e_PDFA6_1_1	Contains an action type that is not permitted.
e_PDFA6_1_2	Contains a non-predefined Named action.
e_PDFA6_2_1	The document catalog dictionary contains AA entry.
e_PDFA6_2_2	Contains the JavaScript key.
e_PDFA6_2_3	Invalid destination.
e_PDFA7_2_1	The document catalog does not contain Metadata stream.
e_PDFA7_2_2	The Metadata object stream contains Filter key.
e_PDFA7_2_3	The XMP Metadata stream is not valid.
e_PDFA7_2_4	XMP property not predefined and no extension schema present.
e_PDFA7_2_5	XMP not included in 'xpacket'.
e_PDFA7_3_1	Document information entry 'Title' not synchronized with XMP.
e_PDFA7_3_2	Document information entry 'Author' not synchronized with XMP.
e_PDFA7_3_3	Document information entry 'Subject' not synchronized with XMP.
e_PDFA7_3_4	Document information entry 'Keywords' not synchronized with XMP.
e_PDFA7_3_5	Document information entry 'Creator' not synchronized with XMP.
e_PDFA7_3_6	Document information entry 'Producer' not synchronized with XMP.
e_PDFA7_3_7	Document information entry 'CreationDate' not synchronized with XMP.
e_PDFA7_3_8	Document information entry 'ModDate' not synchronized with XMP.
e_PDFA7_3_9	Wrong value type for predefined XMP property.
e_PDFA7_5_1	'bytes' and 'encoding' attributes are allowed in the header of an XMP packet.
e_PDFA7_8_1	XMP Extension schema doesn't have a description.
e_PDFA7_8_2	XMP Extension schema is not valid. Required property 'namespaceURI' might be missing in PDF/A Schema value Type.
e_PDFA7_8_3	'pdfeExtension:schemas' not found.
e_PDFA7_8_4	'pdfeExtension:schemas' is using a wrong value type.

e_PDFA7_8_5	'pdfaExtension:property' not found.
e_PDFA7_8_6	'pdfaExtension:property' is using a wrong value type.
e_PDFA7_8_7	'pdfaProperty:name' not found.
e_PDFA7_8_8	'pdfaProperty:name' is using a wrong value type.
e_PDFA7_8_9	A description for a property is missing in 'pdfaSchema:property' sequence.
e_PDFA7_8_10	'pdfaProperty:valueType' not found.
e_PDFA7_8_11	The required namespace prefix for extension schema is 'pdfaExtension'.
e_PDFA7_8_12	The required field namespace prefix is 'pdfaSchema'.
e_PDFA7_8_13	The required field namespace prefix is 'pdfaProperty'.
e_PDFA7_8_14	The required field namespace prefix is 'pdfaType'.
e_PDFA7_8_15	The required field namespace prefix is 'pdfaField'.
e_PDFA7_8_16	'pdfaSchema:valueType' not found.
e_PDFA7_8_17	'pdfaSchema:valueType' is using a wrong value type.
e_PDFA7_8_18	Required property 'valueType' missing in PDF/A Schema Value Type.
e_PDFA7_8_19	'pdfaType:type' not found.
e_PDFA7_8_20	'pdfaType:type' is using a wrong value type.
e_PDFA7_8_21	'pdfaType:description' not found.
e_PDFA7_8_22	'pdfaType:namespaceURI' not found.
e_PDFA7_8_23	'pdfaType:field' is using a wrong value type.
e_PDFA7_8_24	'pdfaField:name' not found.
e_PDFA7_8_25	'pdfaField:name' is using a wrong value type.
e_PDFA7_8_26	'pdfaField:valueType' not found.
e_PDFA7_8_27	'pdfaField:valueType' is using a wrong type.
e_PDFA7_8_28	'pdfaField:description' not found.
e_PDFA7_8_29	'pdfaField:description' is using a wrong type.
e_PDFA7_8_30	Required description for 'pdfaField::valueType' is missing.
e_PDFA7_8_31	A property doesn't match its custom schema type.

e_PDFA7_11_1	Missing PDF/A identifier.
e_PDFA7_11_2	Invalid PDF/A identifier namespace
e_PDFA7_11_3	Invalid PDF/A conformance level.
e_PDFA7_11_4	Invalid PDF/A part number.
e_PDFA7_11_5	Invalid PDF/A amendment identifier.
e_PDFA9_1	An interactive form field contains an action.
e_PDFA9_2	The NeedAppearances flag in the interactive form dictionary is set to true.

#### 4.2.2 PDF/A Level A Validation Errors

The following table is a non-exhaustive list of checks and conversions performed by PDF/A Manager for Level A Validation:

Error Code	Description
e_PDFA3_8_1	The font dictionary is missing 'ToUnicode' entry and additional font requirements for are not met.
e_PDFA8_2_2	The PDF is not marked as Tagged PDF.
e_PDFA8_3_3_1	Bad StructTreeRoot
e_PDFA8_3_3_2	Each structure element dictionary in the structure hierarchy must have a Type entry with the name value of StructElem.
e_PDFA8_3_4_1	A non-standard structure type does not map to a standard type.

## 5. Overview of PDF/A Manager SDK

Besides being offered as a command-line utility, PDF/A Manager is also available for integration with third party applications through two software development kits (SDKs): ‘PDF/A Manager SDK’ and ‘PDFNet SDK PDF/A add-on module’.

PDF/A Manager SDK has a very simple API which resembles options in the PDF/A Manager Command-Line Application. The 'PDFNet SDK PDF/A add-on module' offers a more programmatic interface that is better integrated with other PDF processing options available in the PDFNet PDF library (<http://www.pdftron.com/pdfnet>).

## 5.1 Working with PDF/A Manager SDK

PDF/A Manager SDK offers a simple to use API for PDF/A validation and conversion. The SDK is available as a plain 'C DLL' and can be easily accessed from any programming language (including C#, VB.NET, C/C++, Java, VB6, Perl, Python, Ruby, Delphi, etc).

The entire API consists of only two functions: PDFAInit and PDFARun.

PDFAInit is called only once per process session to initialize the library and register the component. After initializing the library, PDFARun can be called many times to process PDF documents or folders with PDF documents.

The following is the simplest application that can be built using PDF/A Manager SDK:

```
// Using C# or C/C++
void main() {
    PDFAInit( "username" , "company" , "lic_key" );
    PDFARun( "-o c:/out -c -z c:/test/tiger.pdf" , null , new IntPtr(0) );
}
```

This application essentially executes a hardcoded operation that converts 'tiger.pdf' to PDF/A Compliant format (i.e. tiger\_pdfa.pdf stored in 'c:/out'). To convert all PDF documents in 'test' folder simply delete 'tiger.pdf' from the command string.

The first parameter of the PDFARun() function is a command string which is exactly the same as the general syntax used for the PDF/A Manager Command-Line application. For a detailed explanation of all options, please refer to section ‘Basic Syntax’ of this manual. The PDF/A Manager Command-Line application is a great tool to get to know all the available options. In fact, building a command-line application using PDF/A Manager SDK is as simple as the following listing:

```
// Using C#
static void Main(string[] args) {
    PDFAInit("username", "company", "lic_key");

    String s = "";
    foreach (string arg in args) {
        s += arg + " ";
    }
    PDFARun(s, null, new IntPtr(0));
}
```

It is also possible to build the command string dynamically (e.g. based on the user or dynamic input), as illustrated in the following code snippet:

```
// Using C#
static void Main(string[] args)
{
    PDFAInit(username, company, lic_key);

    // See the next section for more info regarding PDFACallback.
    PDFACallback mycallback = new PDFACallback(MyCallback);

    string output_folder = "Output";
    string open_password = "secret";

    bool noxml = false;      // Disable XML report generation?
    bool linearize = true;   // Linearize the converted files?
    bool convert = true;     // Convert or validate?
    bool revalidate = true;  // Revalidate after conversion?
    bool pdfa_la = false;    // Level A or B?

    // -----
    // Given the above settings build a command string.
    string s = "";

    if (output_folder != "") s += "-o " + output_folder + " ";
    if (open_password != "") s += "-p " + open_password + " ";

    if (noxml) s += "--noxml ";
    if (linearize) s += "-z ";
    if (convert) s += "-c ";
    if (revalidate) s += "--nr ";
    if (pdfa_la) s += "-l A ";

    // specify input PDF files and folders...
    s += "../../license.pdf ";
    s += "Output ";
    // s += "c:/my_pdf_folder "; etc ...

    //an array list to store the saved file path(s)
    ArrayList saved_files=new ArrayList();

    //create a GCHandle to allow MyCallback to access
    //the object saved_files
    GCHandle gch = GCHandle.Alloc(saved_files);
    // Execute the command string.
    PDFARun(s, mycallback, (IntPtr)gch);
    gch.Free();

    //print the saved file(s)
    Console.WriteLine("The Saved Files Are:");
    for(int i=0; i<saved_files.Count; i++)
        Console.WriteLine(saved_files[i]);
}
```

## 5.2 Reporting Progress Messages and Errors

To find out if PDF/A Manager processing was successful, the application can query the status code returned by PDFARun().

For example,

```
int ret = PDFARun(...);
if (ret == PDFA_OK) {
    // No errors...
}
else if (ret == PDFA_ERR_BADKEY ) {
    // bad license key...
}
else if (ret == PDF2A_ERR_DIRCREATE ) {
    // Failed to create the output directory
}
else {
    // Other error
}
```

A non-zero value indicates that an error was encountered. You can find the listing for all error code in 'include/pdfa.h' header.

For more detailed error and message reporting, you can pass a pointer to the custom callback function in the second parameter of PDFARun(). The last parameter in PDFARun is a pointer to custom data that you may want to pass to the callback function.

A sample callback function may look as follows:

```
// Using C/C++
char* MyCallback(int mode, char* msg, void* user_data) {
    if (mode == PDFA_ERROR) {
        cout << "Error: " << msg << endl;
    }
    else if (mode == PDFA_MSG) {
        cout << msg;
    }
    else if (mode == PDFA_GETPASS) {
        static string gl_pass;
        cin >> gl_pass;
        return (char*)gl_pass.c_str();
    }
    else if (mode == PDFA_OUT_FILENAME) {
        if(user_data != 0) {
            //stores the filename in the given vector
            vector<string>* saved_files=(vector<string>* )user_data;
            saved_files->push_back(string(msg));
        }
    }
    return 0;
}

' or in VB.NET...
Public Function MyCallback(ByVal mode As Integer, ByVal msg As String,
    ByVal user_data As Int32) As Int32
```

```

If mode = PDF2IMAGE_ERROR Then
    Console.WriteLine("Error: {0}", msg)
ElseIf mode = PDF2IMAGE_MSG Then
    Console.Write("{0}", msg)
ElseIf mode = PDFA_OUT_FILENAME Then
    If user_data.ToInt32 <> 0 Then
        'add the next saved filename to the given ArrayList
        Dim gch As GCHandle = GCHandle.op_Explicit(user_data)
        Dim list As ArrayList = CType(gch.Target, ArrayList)
        list.Add(msg)
    End If
End If
Return 0
End Function      'MyCallback

```

Please note that, by default, PDF/A Manager SDK will also generate ‘report.xml’ outlining validation and conversion results for each file. To disable report generation simply include ‘--noxml’ in the command-line string.

### 5.3 PDF/A Add-on module for PDFNet SDK

All of functionality available in PDF/A Manager SDK is available as an add-on module to PDFNet SDK (<http://www.pdftron.com/pdfnet>). PDFNet SDK is an amazingly comprehensive, high-quality PDF developer toolkit for working with PDF files at all levels. Using the PDFNet PDF library, developers can flexibly implement and create powerful PDF solutions and applications that can generate, manipulate, view, render and print PDF documents without any third-party software dependencies.

PDFNet SDK is available as a .NET component and as a cross-platform Java and C/C++ PDF library available on a wide range of platforms.

This section only illustrates the basic use of the PDFACCompliance class in PDFNet SDK. To obtain the full sample code or PDFNet API Reference, you can simply download PDFNet SDK for your development platform from: <http://www.pdftron.com/pdfnet/downloads.html>.

```

// C# sample code (C/C++, JAVA, and VB6 samples are also available).

using System;
using pdftron;
using pdftron.PDF;
using pdftron.PDF.PDFA;

namespace PDFATestCS
{
    class Class1 {
        // Relative path to the folder containing test files.
        static string input_path = "../../../../../TestFiles/";
        static string output_path = "../../../../../TestFiles/Output/";

        static void Main(string[] args)
        {
            PDFNet.Initialize();
            PDFNet.SetColorManagement(true); // Required for PDFA validation.

            // Example 1: PDF/A Validation

```

```
try {
    string filename = "newsletter.pdf";
    PDFACCompliance pdf_a = new PDFACCompliance(false,
input_path+filename, null, PDFACCompliance.Conformance.e_Level1B,
null, 10, false);
    PrintResults(pdf_a, filename);
    pdf_a.Dispose();
}
catch (pdftron.Common.PDFNetException e) {
    Console.WriteLine(e.Message);
}

// Example 2: PDF/A Conversion
try {
    string filename = "fish.pdf";
    PDFACCompliance pdf_a = new PDFACCompliance(true,
input_path+filename, null, PDFACCompliance.Conformance.e_Level1B,
null, 10, false);
    filename = output_path + "pdfa.pdf";
    pdf_a.SaveAs(filename, true);
    pdf_a.Dispose();

    // Re-validate the document after the conversion...
    pdf_a = new PDFACCompliance(false, filename, null,
PDFACCompliance.Conformance.e_Level1B, null, 10, false);
    PrintResults(pdf_a, filename);
    pdf_a.Dispose();
}
catch (pdftron.Common.PDFNetException e) {
    Console.WriteLine(e.Message);
}
PDFNet.Terminate();
}

static void PrintResults(PDFACCompliance pdf_a, String filename) {
    int err_cnt = pdf_a.GetErrorCount();
    if (err_cnt == 0)
        Console.WriteLine("{0}: OK.\n", filename);
    else {
        Console.WriteLine("{0} is NOT a valid PDFA.", filename);
        for (int i=0; i<err_cnt; ++i) {
            PDFACCompliance.ErrorCode c = pdf_a.GetError(i);
            Console.WriteLine(" - e_PDFA{0}: {1}.",
                c,
                PDFACCompliance.GetPDFAErrorMessage(c));
        }
    }
}
```

## 6. Support

### 6.1 Reporting Problems

If you encounter a problem or question regarding PDFTron's PDF/A Manager, which is not addressed on PDFTron's website, please submit a problem report to PDFTron's Support group at <http://www.pdftron.com/support/reportproblem.html>.

When submitting a problem you will be asked to provide the following information:

- Contact details
- Product and Version of the product
- Detailed description of problem
- Problem file(s)
- Whether you have an AMS (Annual Maintenance Subscription)
- Any other information that may be related

### 6.2 Contact Information

To contact PDFTron directly, please use the contact information below:

Tel: 1-604-730-8989  
Fax: 1-604-676-2477

Web site: [www.pdftron.com](http://www.pdftron.com)

Email Contacts:

General Business Inquiries: [info@pdftron.com](mailto:info@pdftron.com)  
Sales & Licensing: [sales@pdftron.com](mailto:sales@pdftron.com)  
Product Support: [support@pdftron.com](mailto:support@pdftron.com)  
Professional Services: [services@pdftron.com](mailto:services@pdftron.com)  
Website related questions: [webmaster@pdftron.com](mailto:webmaster@pdftron.com)  
Press & News: [press@pdftron.com](mailto:press@pdftron.com)