

Adobe PDF Library SDK 18.0.4 Read Me

This document contains information on installing the Adobe PDF Library SDK, as well as general information regarding the release. In addition to this document, please read the PDF Library Overview for important information on developing with and using the PDF Library. See [installation instructions](#).

Table of Contents

Major updates in Release XVIII (18.0.4)	2
Major updates in Release XVIII (18.0.3)	2
Major updates in Release XVIII (18.0.2)	2
Major updates in Release XVIII (18.0.1)	2
Major updates in Release XVIII (18.0)	2
Bugs fixed in Release XVIII (18.0.4)	3
Known issues	3
Getting started	3
Set up	4
All platforms	4
Installing on Windows	5
Installing on Mac	6
Installing on UNIX	7
Documentation	8
Third Party Software Notices and/or Additional Terms and Conditions	9

Major updates in Release XVIII (18.0.4)

- ICU libraries have been upgraded to the latest version i.e. 64.2.
- All files are now in .cpp extension. With this change, default compiler settings would work to build the entire PDFL. For backward compatibility, we have not removed .c files.
- PDEFormIsLeadingSet API has been added which checks whether the Text Leading (TL) is set in the graphics state or not.
- Parameter 'pathDPI' of PDFflattenRec structure has been enabled which affects the degree of flattening Beziars.
- Location of resources for Type3 fonts are aligned with PDF specification.

Major updates in Release XVIII (18.0.3)

- Support for UTF-8 character encoding as per PDF 2.0 specification has been added.
- From 18.0.3 onwards, Mac Builds are BAST V3 signed following the Apple Notarization Guidelines.

Major updates in Release XVIII (18.0.2)

- Support for new era glyph – Updated CMaps and Font resources to support Japanese new era characters.
- Support for Windows 7 has been discontinued.
- PDEExtGStateGetSoftMaskMatrix and PDEExtGStateSetSoftMaskMatrix APIs have been added to get and set the soft mask matrix in external gstate respectively.

Major updates in Release XVIII (18.0.1)

- Two new APIs PDEFormGetLeading and PDEFormSetLeading have been added to get and set respectively the Leading in the parent of Form.
- A new parameter weight of type unsigned integer is added in PDEFontAttrs structure.

Major updates in Release XVIII (18.0)

- Support for PDF 2.0 file version - PDF Library can now read and edit files with PDF 2.0 version. Two new APIs PDDocGetAdobePDFVersion and PDDocSetAdobePDFVersion have been introduced in this release for getting and setting new PDF version respectively.
- Support of Black point compensation (BPC) flag - UseBlackPtComp has been added to the graphic state parameter dictionary. Two new APIs PDEExtGStateGetBPC and PDEExtGStateSetBPC have been added to get and set UseBlackPtComp flag value respectively.
- Support for Page level Output Intent - User can set Output Intent for each Page as well. Two new APIs PDDocColorConvertEmbedOutputIntentEx and PDPageColorConvertEmbedOutputIntent have been added to support new output intent subtypes added in PDF 2.0. Appropriate subtype like GTS_PDFX, GTS_PDFA, and ISO_PDFA1 can be added to the output profile as per the value passed to these APIs.
- If output intent already exists at Doc/Page level, then new output intent gets added to the existing one.
- PDColorConvertPDEElementEx2 API has been added to support conversion to a PDEElement to the supplied color space.
- PDSysEncodingSetIsUTF16 API has been added to check if an encoding is UTF-16 or not.
- In PDF 2.0, versioning model has been changed and now support 2-part version system which is a combination of Major and Minor version like Major.Minor as compared to 3-part

version system Major.Minor.ExtensionLevel in PDF 1.7.

- Updated Decryption Sample to showcase the support for Unicode based password protected PDFs.
- Updated CMaps and mapping resources to support Adobe-Japan1-7 fonts.
- There is a change in the supported compiler in this release. Refer to [Getting started](#) section for details.
- There is a change in the OS supported for Linux in this release. Refer to [Getting started](#) section for details.
- Regular support on Linux-32 has been withdrawn.

Bugs fixed in Release XVIII (18.0.4)

- 3687350 : Curve got flattened by PDFL Flattener.
- 4176414 : PDPAGEDrawContentsToMemory() crashes after printing.
- 4176466 : Wrong optional content flattening with OCMDs in PDFL.
- 4176469 : Characters shift while using PDEFORMSetLeading().
- 4176474 : PDPAGEDrawContentsToMemoryEx() reports Access violation in bib.dll.

Known issues

- **PDF/A-b conversion fails if embedded fonts have empty files**
Until PDFL SDK 15.0.6, text marked with rendering mode 3 was deleted from a PDF when it was converted to PDF/A-1b. While this problem has been fixed in 15.0.7, the conversion fails if the embedded fonts have empty font files.
- **Loss of text in Type 3 fonts during PDF/A and PDF/X conversion**
This is a known issue with PDF files having images or path elements inline with text in Type 3 fonts. The text following the inline element is lost during conversion to the PDF/A or the PDF/X format.
- **Stamp annotations missing on conversion to PDF/X**
If the value of Rect¹ lies outside of the BleedBox, on conversion to the PDF/X format stamp annotations in the source are not retained in the resultant PDF.

Getting started

1. Set up the PDF Library SDK on your machine as described in the [Set up](#) section.
2. Before you begin, read the PDF Library Overview. The PDF Library Overview describes the contents of each directory in the SDK installation, lists available code samples, and provides platform-specific information on how to set up the development environment.
3. Examine, compile, and run the relevant code samples or snippets for your platform. The code samples and the SnippetRunner are in the AdobePDFLSDK18.0.4/samples directory.
4. On all platforms, read the comments in the source code before compiling and running the sample to understand the functionality demonstrated by the sample.

¹ Rect – The annotation rectangle, defining the location of the annotation on the page in default user space units.

The following operating systems and compilers are supported by PDFL SDK XVIII (18.0.4):

Platform	Processor	Compiler	Additional OS
Windows-10 32-bit and 64-bit	Intel	Visual Studio 2017 (version 15.3.3)	Win 8.1
Linux 64-bit (CentOS 7.3)	Intel	Clang 3.9.1, glibc 2.17	
Mac 64-bit (Mac OS X 10.12)	Intel	XCode 8.2.1	Mac OS X 10.11, Mac OS X 10.15

Note: Support for Windows 7 has been discontinued.

Important: While it may be possible to use the library in other development environments, such use is not supported by Adobe. The PDF Library has been tested only with the compiler versions listed in the table above.

Set up

All platforms

The following is the directory structure for all platforms:

```
/AdobePDFLSDK18.0.4
  /Docs
  /Include
    /Headers
    /Source
  /Libs/PLATFORM
  /Plugins
    /Flattener
      /Include
      /Libs/PLATFORM
      /Samples
    /PDFProcessor
      /Include
      /Libs/PLATFORM
      /Samples
    /XPS2PDF
      /Include
      /Libs/PLATFORM
      /Resource
      /Samples
  /Resource
  /Samples/SAMPLE_NAME
```

Modify the MyPDFLibUtils.cpp file so that the folder lists passed to PDFLDataRec's dirList, cMapDirectory, unicodeDirectory, and colorProfileDirList members in the PDFLInitHFT() via PDFLGetDirList, PDFLGetCMapDir, PDFLGetUnicodeDir, and PDFLGetColorProfileDirList function calls correctly locates the font, CMap, unicode, color profile files on your system. This is needed for samples that require font, CMap, unicode, or color profile files.

Important:

1. It is recommended that all PDFLSDK API calls should go from Host Function Table (HFT). To enable HFT, all projects should be initialized with PDFLInitHFT and should be terminated with PDFLTermHFT APIs respectively. Host Function Table (HFT) has been in place since PDFL 7.0 and all new APIs added after 7.0 are accessible through HFT only.
2. From 18.0 onwards, all PDFLSDK client projects should enable C++ based Exception Handling. To achieve that kindly set USE_CPLUSPLUS_EXCEPTIONS_FOR_ASECEPTIONS flag to 1.
3. To enable HFT and C++ based Exception Handling efficiently, please include the platform specific project-setting files available in Samples/Utils folder.
4. Samples are tested and built with the valid values in the PDFLDataRec structure. If you do not have the valid font, CMap, unicode, or color profile folders on your system, samples may crash or throw an exception. You must either install the font and color profile files in the correct folder or you must pass NULL to these members for the PDF library not to look for the fonts or color profile files.

Note: On Windows platform, kindly include the PDFLInitCommon.cpp file available in Include/Source folder to enable C++ Exception Handling across multiple HFT calls.

Installing on Windows

The following two archives are shipped for Microsoft Windows:

- AdobePDFLSDKMinSize18.0.4.zip - For standard version of the PDF Library SDK.
- AdobePDFLSDKMaxSpeed18.0.4.zip - For the speed-optimized version of the PDF Library SDK.

a) Verifying the archive

The Windows archives are now signed and users can verify them for authenticity. To verify the signature of the ZIP file:

1. Install JDK (preferably JDK 1.6 or higher) on your machine.
2. From the *bin* folder of the JDK installation, execute the following command:
jarsigner.exe -verify [-verbose -certs] %ZIP_File_Path%

Note: The following parameters are optional: {-verbose, -certs}

b) Installation

1. Unzip the preferred ZIP file:
 - For standard version of the PDF Library SDK, use AdobePDFLSDKMinSize18.0.4.zip.
 - For the speed-optimized version of the PDF Library SDK, use AdobePDFLSDKMaxSpeed18.0.4.zip.

Important: If for some reason, both the above variants need to exist on the disk, ensure that they are extracted into separate locations.
2. To ensure that applications run successfully outside the Microsoft Visual Studio C++ .NET IDE, make sure that the executable always find the dynamic linked libraries at run time:
 - Copy the libraries supplied in the Libs folder to the folder where your executable resides.
 - Place the path for the libraries into the PATH environment variable.
3. To ensure that the plugins run successfully:
 - Copy the libraries supplied in the Plugins\[PLUGIN_NAME]\Libs folder to the folder where your executable resides.
 - Place the path for the libraries into the PATH environment variable.

Executing SnippetRunner in UI mode (64-bit)

You must explicitly set the 64BitMode parameter to 1 in the pdfsdk.config file in your home directory. This variable is initialized to 0 (zero) by default.

Installing on Mac

The DMG file, AdobePDFLSDK18.0.4.dmg, is shipped for the Mac platform.

a) Verifying AdobePDFLSDK18.0.4.dmg package for Mac

From 18.0.2 onwards, the signing process on Mac platform has been changed. Now, SIG file is not required. To view the signature of the DMG file, you can use the following codesign command:

```
codesign -dvv $DMG_File_Path
```

Example: codesign -dvv AdobePDFLSDK18.0.4.dmg

It should show the signature information.

If not signed, code object is not signed at all will be displayed

b) Installation

1. Mount the DMG image for AdobePDFLSDK18.0.4.dmg.
2. Copy the AdobePDFLSDK18.0.4 folder to a local drive.

By default, all libs are present under the AdobePDFLSDK18.0.4/Libs/folder; categorized under mac_x64 for 64-bit target configuration. Aliases or proper paths must be set to ensure that the sample applications run properly.

Installing on UNIX

To install the Adobe PDF Library SDK on the UNIX platform:

1. Copy the file AdobePDFLSDK18.0.4.tar.gz to a directory.
2. Run the following command in the directory that you want to unpack the SDK:
`gunzip < AdobePDFLSDK18.0.4.tar.gz | tar xvf`
3. Change directory to AdobePDFLSDK18.0.4/Samples/Utils and modify the appropriate .mak file to point to your installed Clang directory and static library directories.
4. Set the environment variable LD_LIBRARY_PATH to point to AdobePDFLSDK18.0.4/Libs/linux_x64/ folder

Before you can build your product on a Linux platform running the CentOS 7.3 operation system, you must procure the Clang compiler. We recommend using version 3.9.1 or later, and keeping the compiler in the directory: /opt/llvm/

Note – For linux platform, new libraries libc++, libc++abi and libunwind are now included along with the existing libraries libgcc and libstdc++ in the packaging and makefiles for samples has been updated for the same. These libraries are released in AdobePDFLSDK18.0.4/Libs/linux_64/ folder.

Documentation

1. **PDF Reference and Related documentation, Sixth Edition** (pdf_reference.pdf) is available in the /AdobePDFLSDK18.0.4/Docs folder: This PDF Portfolio contains the documents that describe the PDF standard supported by Adobe Acrobat®. It includes these documents:

PDF Reference, sixth edition, version 1.7 (Nov.2006)

PDF Redaction: Addendum to the PDF Reference, sixth edition, version1.7

Errata for the PDF Reference, sixth edition, version1.7

Adobe Supplement to the ISO 32000, BaseVersion 1.7, ExtensionLevel 3 (for Acrobat XI)

2. **PDF Library API Reference** is available in HTML form in the PDFL_SDK_HTMLHelp compressed archives. This document is the reference manual for all APIs exposed by the PDF Library. Please visit the Documentation section of the Release website to access the archives.
3. **PDF Library Overview:** Provides an introduction to development using the Adobe PDF Library. Please visit the Documentation section of the Release website to access this document.

Third Party Software Notices and/or Additional Terms and Conditions

Copyright 2020 Adobe Systems Incorporated and its licensors. All rights reserved.

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

This product contains either BSAFE and/or TIPEM software by RSA Security, Inc.

Portions utilize Microsoft Windows Media Technologies. Copyright (c) 2006 Microsoft Corporation.

All Rights Reserved.

Notices, terms and conditions pertaining to other third party software are located at

<https://www.adobe.com/go/thirdparty/> and incorporated herein by reference.