Release Notes v. 25.0

This guide describes how to install the Adobe PDF Library SDK, as well as general information regarding the release. In addition to this document, please read the PDF Library documentation for information on developing with and using the PDF Library.

Tip

This document is available in HTML.

Version 25.0, January 22, 2026

Changes for this release

- Compiler changes as described in System Requirements (below)
- Support for MacExpertEncoding is added to PDFL
- Zlib upgraded to version 1.3.1
- PDFL upgraded to OpenSSL version 3.0.16
- ICU libraries are upgraded to version 74.2

Note

In our packaging, the updated OpenSSL library is included across all platforms as it is dynamically linked at runtime.

Fixed bugs

- 4176738: Vertical aligned text shifts after processing PDPageSetPDEContent()
- 4176715: Crash in Bib.dll observed while using PDPageDrawContentsToMemoryEx()
- 4176733: Text disappears after saving with PDSaveLinearized flag
- Updated Base Image Address for AXE8SharedExpat.dll and XPS2PDF.ppi to conform with effective ASLR standards.
- 4176713: Flattener plugin resolution parameters restricted incorrectly. The limits for 2 flattener parameters are changed to correspond to Acrobat's behaviour.
  - internalDPI: (equivalent to "Gradient and Mesh resolution" in Acrobat) is now limited to 1200. The previous value was 9600.
  - externalDPI: (equivalent to "Line Art and Text resolution" in Acrobat) is now limited to 9600. The previous value was 1200.

Getting started

System requirements

The following operating systems and compilers are supported by PDFL 25.0:

The product dropped support for Windows 32-bit with this release.

V. 25.x Compiler Support

Platform	Processor	Compiler
Windows 11 64-bit	Intel	Visual Studio 2022 (17.14.7)
Linux 64-bit, Ubuntu 22.04	Intel	Clang 18.1.8
Mac 64-bit (OS 15.2)	Intel and ARM	Xcode 15.2

Installation overview

All installations generally involve the following:

1. Setting up your environment as described below.
2. Reading the PDF Overview which provides links to related documentation, including the PDF Reference and Snippet Runner Cookbook.
3. Examine, compile, and run the relevant code samples or snippets for your platform (located in the AdobePDFLSDK25.0.0.1/SAMPLES/ directory).
4. On all platforms, read the source code comments before compiling and running the sample so that you understand the demonstrated functionality.

Directory structure

The following is the directory structure for all platforms:

```
/AdobePDFLSDK<version>
    /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
```

Path modification

Because some samples require font, CMap, unicode, or color profile files, modify the MyPDFLibUtils.cpp file with the folder paths on your system.

For example, `PDFLInitHFT()`'s function calls `PDFLGetDirList`, `PDFLGetCMapDir`, `PDFLGetUnicodeDir`, and `PDFLGetColorProfileDirList` to locate the font, CMap ,unicode, color profile files on your system when they pass PDFLDataRec's `dirList`, `cMapDirectory`, `unicodeDirectory`, and `colorProfileDirList` members.

Best practices

General

- Examine, compile, and run the relevant code samples or snippets for your platform. The code samples and the SnippetRunner reside in the AdobePDFLSDK25.0/samples directory.
- On all platforms, read the comments in the source code before compiling and running the sample to understand the functionality demonstrated by the sample.

Required best practices

1. 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_ASEXCEPTIONS 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 PDFLDataRecstructure. 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.

Windows installation

The following two archives are shipped for Microsoft Windows:

- AdobePDFLSDKMinSize25.0.0.1.zip - For standard version of the PDF Library SDK.
- AdobePDFLSDKMaxSpeed25.0.0.1.zip - For the speed-optimized version of the PDF Library SDK.

1. Verify the signature of the ZIP file:
   - Install JDK (preferably JDK 1.6 or higher) on your machine.
   - From the bin folder of the JDK installation, execute the following command:

     ```
     jarsigner.exe -verify [-verbose -certs] %ZIP_File_Path%
     ```

The following parameters are optional: `-verbose`, `-certs`

2. Unzip the preferred ZIP file: Important: If both the above variants need to exist on the disk, extract them to separate locations.
   - For standard version of the PDF Library SDK, use AdobePDFLSDKMinSize25.0.0.1.zip.
   - For the speed-optimized version of the PDF Library SDK, use AdobePDFLSDKMaxSpeed25.0.0.1.zip.
3. To ensure that applications run successfully outside the Microsoft Visual Studio C++ .NET IDE, verify the executable can 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.
4. 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.
5. Include the PDFLInitCommon.cpp file available in Include/Source folder to enable C++ Exception Handling across multiple HFT calls.

Note

Executing SnippetRunner in UI mode (64-bit): You must explicitly set the `64BitMode` parameter to 1 in the `pdflsdk.config` file in your home directory. This variable is initialized to 0 (zero) by default.

Mac installation

The DMG file, AdobePDFLSDK25.0.0.1.dmg, is shipped for the Mac platform. To install: 1. Verify the AdobePDFLSDK25.0.0.1.dmg. From 18.0.2 onwards, the signing process on Mac platform has been changed. Now, the SIG file is not required. To view the signature of the DMG file, you can use the following codesign command:

```
codesign -dvv AdobePDFLSDK25.0.0.1.dmg
```

Note

The signature information appears. If it is unsigned, "code object is not signed at all" appears.

2. Mount the DMG image for AdobePDFLSDK25.0.0.1.dmg.
3. Copy the AdobePDFLSDK25.0.0.1 folder to a local drive.

By default, all libs are present under the AdobePDFLSDK25.0.0.1/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.

Linux installation

1. Copy the file AdobePDFLSDK25.0.0.1.tar.gz to a directory.
2. Run the following command in the directory that you want to unpack the SDK:

```
tar -xvf AdobePDFLSDK25.0.0.1.tar.gz
```

3. Change directories to AdobePDFLSDK25.0.0.1/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 the AdobePDFLSDK25.0.0.1/Libs/linux_x64/ folder

Before you can build your product on a Linux platform running the Ubuntu 22.04 operating system, install the Clang compiler. We recommend using version Clang 18.1.8 and keeping the compiler in the directory: /opt/llvm/

Additional documentation

- PDF Reference, sixth edition, version 1.7
- PDF Redaction: Addendum to the PDF Reference, sixth edition, version 1.7
- Errata for the PDF Reference, sixth edition, version
- Adobe Supplement to the ISO 32000, BaseVersion 1.7, ExtensionLevel 3 (for Acrobat XI)
- PDF Library API Reference. The API Reference is the reference manual for all APIs exposed by the PDF Library.
- PDF Library Overview: Provides an introduction to development using the Adobe PDF Library.

3rd party software notices and TOU

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.