Adobe Acrobat X SDK Release Notes

Version 10.0

Readme File

This file contains important information that was not available when the product documentation was written. Review this information before using this version of the Adobe® Acrobat® SDK.

Contents

- About this Release
- What's New
- Known Issues
- Documentation

About this Release

This is Acrobat X SDK Release for Windows and Mac OS.

Back to Contents

What's New

The *What's New* topic in the **Introduction to SDK** topic of the Acrobat SDK Help contains details on the new and changed features in this release.

Back to Contents

Known Issues

Acrobat Distiller restriction on directory access

Beginning with Acrobat 8.1, Distiller® restricts the directories that PostScript® file operators can access. The new default behavior limits directory access to the temp and font cache directories. Earlier versions of Distiller allowed PostScript file operators to have unlimited directory access.

The following Distiller settings enable unlimited directory access. Such unlimited access can pose security problems.

- Microsoft® Windows®: `-F` command line option
- UNIX®: `-allowfileops` command line option
- Apple Mac OS: `AllowPSFileOps` user preference

Common Interface fails to connect to SnippetRunnerServer plug-in

When CI is started for the first time, you will be asked to select the root of the SnippetRunner folder. This selection stays valid each time CI is started. The root SnippetRunner folder should be in the same partition as the boot partition. Otherwise, CI will not work correctly.

Common Interface may fail to connect with the SnippetRunnerServer plug-in, or fail to reflect code snippet changes after rebuilding the SnippetRunnerServer plug-in and restarting Acrobat. This may happen when a PDF document is opened in a browser prior to launching Acrobat on the desktop. CI may be connecting with the SnippetRunnerServer plug-in associated with Acrobat in the browser, not Acrobat on the desktop.

To resolve this problem quit your browser, then restart Acrobat and CI.

Adobe Reader Plug-in Development for Microsoft Windows

To enable your plug-in for Adobe Reader X, you must update `AcroDspOptions.rsp` by removing the `acrocolorHFT` define and adding the READER_PLUGIN define. To define the READER_PLUGIN symbol in a Win32 project in Visual Studio C++, do the following:

1. Locate the `AcroDspOptions.rsp` file in the SDK's `/PluginSupport/Samples` directory.
2. Open `AcroDspOptions.rsp` in a text editor.
3. Change `ACRO_SDK_LEVEL` to `0x000A0000` from `0x00090000`.
4. Remove `the AcroColorHFT` define by removing the following line from the file:
   `/D PI_ACROCOLOR_VERSION=0x00060000`
5. Add the READER_PLUGIN define by adding the following to the file:
   `/D READER_PLUGIN`
6. Save `AcroDspOptions.rsp`.
7. Open and rebuild your Visual C++ plug-in project.

Note: This change affects all SDK plug-ins built after the change is made.

Some Acrobat features are not enabled in external windows unless the AV_EXTERNAL_VIEW flag is set.

One example is Search.

Acrobat and PDF Library API Reference: PDBookmarkGetIndent

The `PDBookmarkGetIndent` method should return the indentation level of `aBookmark` in its containing tree. The root and its direct children have an indentation level of 0.

Windows Vista Issue: Privilege Setup for Application Wizard (Guide to SDK Samples)

If you intend to use Visual Studio App Wizard to create a plug-in that uses privileged operations and you are running on Windows Vista™, you must establish yourself as an administrator. This is required even if your user ID has Admin privileges.

To establish yourself as an administrator, perform these tasks:

1. Select the Windows start icon.
2. From All Programs, navigate to Visual Studio C++.

3. Right click Visual Studio C++.
4. Select "Run as ...."
5. Select "Run as administrator", which is the only selection that elevates your privilege level to administrator.
6. Proceed with instructions in the *Guide to SDK Samples*

SnippetRunner on Mac OS X and Windows Vista

The SnippetRunner Common Interface (based on Adobe Flash® technology) will not function as intended with the Safari browser in Mac OS X 10.4.8. It is recommended that you use the FireFox browser freely available from the Mozilla Foundation (http://www.mozilla.org) if you want to exercise this sample.

If you are using SnippetRunner on Windows Vista, the directory in which you place the Trust Policy file differs from the corresponding directory on Windows XP. Specifically, if you are using Windows Vista, use these instructions to place the Trust Policy file:

1. In the directory `[UserHome]\AppData\Roaming\Macromedia\Flash Player`, manually create the subdirectory `#Security\FlashPlayerTrust`.
2. Place an ASCII text Trust Policy file (`CommonInterfaceFX.cfg`) in the Flash Player Trust directory you created:
   `[UserHome]\AppData\Roaming\Macromedia\Flash Player\#Security\FlashPlayerTrust`

PDF Support for U3D

U3D supports only the latest IETF Unicode, specified at http://tools.ietf.org/html/rfc3629. Therefore, PDF support for U3D is also restricted to that version of Unicode

Change to ASFileSysPathFromDIPathProc Function

If you implement a file system, you should change the signature of the `pathFromDIPath` callback to add the `const` keyword to the `diPath` parameter, as in the following code:

```
typedef ACCBPROTO1 ASPathName (ACCBPROTO2 *ASFileSysPathFromDIPathProc)
(
const char * diPath, //Add const here
ASPathName relativeToThisPath
    );
```

Deprecated Help APIs

The following APIs related to help have been deprecated: `AVAppHelpSearch`, `AVAppHelpShowContents`, and `AVAppHelpShowIndex`. The corresponding parameters for opening PDF files (`help=index|contents|search`) are also no longer available.

Acrobat 8.0 Plug-in Modeless Dialog Box Does Not Receive Focus

If you have created an Acrobat or Reader plug-in that pops open a dialog box with which your users will interact, the dialog box may not receive focus and may require a double click to get focus. You can work around this issue by modifying the plug-in code with an appropriate call to `AVAppBeginModal` before the dialog box pops up. Then call `AVAppEndModal` immediately thereafter.

Unable to Reader-enable a Plug-in Using the Acrobat 8.x SDK

If you are attempting to Reader-enable a plug-in for use with Adobe Reader 8.x and the plug-in fails to load without an error message, the `#define` directives are incorrect in the following two SDK header files:

PIHeaders.h and PIHeaders++.pch. For replacement headers containing the correct `#define` directives for Reader-enabling, see the SDK zip folder located within …/PluginSupport/Tools/Reader-enabling Tools/.

Using AVDoc.OpenInWindowEx Causes Search Functions to Stop Working

If you are using OLE automation and invoke the `AVDoc.OpenInWindowEx` method specifying `AV_DOC_VIEW` for the `openFlags` parameter, the search function may stop working. If this occurs, specify `AV_EXTERNAL_VIEW` for the `openFlags` parameter and use code such as the following to hide the toolbar:

```
myPDDoc = AcroExchAVDoc(fIndex).GetPDDoc();

jso = myPDDoc.GetJSObject();

jso.app.toolbar = false;

MakeKey Reader-enabling Tool
```

At this time, the MakeKey tool for Reader-enabling does not generate keys if you are using Mac OS 10.x on MacIntel or Windows Vista on 64-bit machines. Please check www.adobe.com/go/acrobat_developer periodically for future updates to this tool.

Back to Contents

Documentation

The Acrobat SDK includes a complete set of documentation in the **Documentation** folder of the Acrobat SDK installation. The *Acrobat SDK Help* provides access to this documentation set. To view the documentation, you must have an Internet browser and Adobe Reader® installed on your computer.

Acrobat and PDF Library API Reference: PDWordFinderAcquireWordList sort

The description of the function `PDWordFinderAcquireWordList` contains an over-simplification of the sorted results returned through the parameter `xySortTable`. That is, this description fails to mention how reading order affects sort order. When the functions `PDDocCreateWordFinder` or `PDDocCreateWordFinderUCS` are called with the `WXE_XY_SORT` flag set, these functions return an array of PDWords, sorted in x-y order.

The API determines sort order by splitting the input PDF text content into multiple regions (which are basically vertical bounded areas in the input PDF text content) based on the text-blocks present and then determining the sort order within those regions. The order in which the different regions will actually be processed in the reading order depends on the kind of script being used because that determines whether to read from right to left (for example Arabic) or from top to bottom or any other order.

Acrobat Distiller API Reference

In its description of Distiller command line options for use on Windows, the *Distiller API Reference* erroneously describes the `-F` as restricting directory access. In fact, that option enables unlimited directory access.

Acrobat and PDF Library API Reference: PDFL enumeration

The enumerations section for the `PDFL` object has an error in the value it associates with the `TextServer` enumerated name `kPDPageDrawSmoothImage`. This name has the value 0x00000010, as shown in the following declaration.

```
enum TextServer{ kPDPrefAASmoothText = 0x00000001, kPDPrefAASmoothLineArt =
0x00000002, kPDPrefAASmoothImage = 0x00000010 }
```

JavaScript for Acrobat API Reference: exportAsFDF, exportAsXFDF, importDataObject

The quick bars for these entries should be as follows:

exportAsFDF

4.0		S	F C

exportAsXFDF

5.0		S	F C

importDataObject

5.0	D	S	D

Back to Contents