

# Adobe Acrobat 7.0.5



## Developer FAQ

July 27, 2005



Adobe Solutions Network — <http://partners.adobe.com>



© 2005 Adobe Systems Incorporated. All rights reserved.

#### Acrobat® Developer FAQ

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 in sample templates are for demonstration purposes only and are not intended to refer to any actual organization.

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

Apple, Macintosh, and Power Macintosh are trademarks of Apple Computer, Inc., registered in the United States and other countries. PowerPC is a trademark of International Business Machines Corporation in the United States and/or other countries. ActiveX, Microsoft, Windows, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. UNIX is a trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd. Solaris is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. Linux is a registered trademark of Linus Torvalds. AIX is a trademark of International Business Machines Corporation in the United States and/or other countries. All other trademarks are the property of their respective owners.

This software is based in part on the work of the Independent JPEG Group.

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.



# Contents

Basic Acrobat SDK Questions . . . . .	5
What Is Acrobat? . . . . .	5
What Is the Acrobat SDK? . . . . .	6
What Is a Plug-in? . . . . .	7
What Is a Certified Plug-in? . . . . .	8
How Does a Script Written in JavaScript Differ From a Plug-in? . . . . .	8
What Are Some Common Uses of the Acrobat SDK? . . . . .	9
Where Do I Start with the Acrobat SDK? . . . . .	11
Where Can I Find Guidelines On Developing Adobe Reader Plug-ins? . . . . .	12
What Are the Requirements For Using Acrobat Forms? . . . . .	12
What Are the Requirements For Using XML Forms? . . . . .	13
What is the FDF Toolkit? . . . . .	13
Licensing and Distribution . . . . .	13
What Should I Know About Acrobat Licensing and Distribution? . . . . .	13
What Technical and Licensing Issues are Restricted in the Acrobat SDK? . . . . .	15
What ActiveX Solutions Are Provided By the Acrobat SDK? . . . . .	17
User and Developer Resources . . . . .	17
What Do I Need to Download from the Web to Get the Acrobat SDK? . . . . .	17
Where Can I Get Help With Acrobat Product Issues? . . . . .	17
Acrobat SDK Development Support Resources . . . . .	18
How Can I Contact Adobe For Technical Support Issues? . . . . .	18
What Is the ASN Developer Program and How Do I Join? . . . . .	18
What Adobe Resources Are Recommended to Acrobat Developers? . . . . .	19
What Developer Resources Are Provided by Outside Companies? . . . . .	19
Acrobat and Adobe Reader Comparison . . . . .	20
What Are the API Differences Between Acrobat and Adobe Reader? . . . . .	20
Developing for Adobe Reader . . . . .	20
What Is a Rights-enabled PDF Document? . . . . .	20
Visual Basic .NET and Visual C# .NET . . . . .	21
How Can I Use Visual Basic .NET or Visual C# .NET With the Acrobat SDK? . . . . .	21
What Are the Best Resources For Getting Started Using Visual Basic .NET or Visual C# .NET With the Acrobat SDK? . . . . .	21
Enterprise Installation . . . . .	22
How Can I Customize the Acrobat Installer? . . . . .	22
PDF Documents. . . . .	23

What API Methods Are Available to Modify PDF Documents? . . . . .	23
What Options Do I Have to Modify PDF Documents Without C Programming Background?. . .	23
How Can I Extract Text From PDF Documents Using the Acrobat SDK? . . . . .	23
How Can I Display a PDF File in an External Application Window? . . . . .	24
Are Multibyte Font PDF Documents Supported by the Acrobat SDK? . . . . .	24
How Are Security and Encryption Provided in PDF Documents? . . . . .	25
Full-Text Search . . . . .	25
What Tools Are Available with Acrobat For Full-Text Search?. . . . .	25
What Tools Are Available for Extracting and Highlighting Text?. . . . .	25
How Do I Communicate with the Acrobat Search Plug-in? . . . . .	26
How Do I Create Custom DocInfo Fields For Searching?. . . . .	26
How Do I Use Command Lines with Acrobat and Adobe Reader on Windows? . . . . .	27



# Acrobat Developer FAQ

---

## Basic Acrobat SDK Questions

### What Is Acrobat?

Adobe® Acrobat® 7.0 consists of a family of products for creating, modifying, indexing, searching, displaying, and manipulating PDF (Portable Document Format) files. For further information about the Acrobat family of products, see the [Acrobat Main Page](#).

### What is the Acrobat Family?

- Adobe Reader® for viewing, navigating, and printing PDF documents. Adobe Reader is free software that lets anyone view and print Adobe PDF files on all major computer platforms, as well as fill in and submit Adobe PDF forms. Adobe has distributed more than 500 million copies of the software worldwide.
- Acrobat for adding navigational links, annotations, and security options, in addition to the functionality provided by Adobe Reader. Acrobat 7.0 is offered in two configurations:
  - Acrobat 7.0 Standard — Offers all of the Adobe PDF creation and distribution tools in Acrobat Elements® (see below) and also allows users to more effectively manage document review cycles. Acrobat Standard's intuitive review and commenting tools help streamline collaboration and reduce the costs and inefficiencies of paper-based review processes.
  - Acrobat 7.0 Professional — Contains all of the functionality of Acrobat 7.0 Standard and Acrobat Elements® (see below). In addition, it offers the most advanced control over document creation, exchange, review, and output. It is intended for users who need to perform sophisticated tasks, such as preflighting print files, distributing layered CAD drawings, and creating interactive forms.
- Acrobat Elements for organizations requiring widespread PDF creation capability without the need to install the full Acrobat Distiller® product on every desktop. Acrobat Elements allows organizations to extend their investments in Microsoft® Office by standardizing on Adobe PDF for document distribution. Acrobat Elements is available through licensing with minimum orders of 1,000 seats. There is no public API for Acrobat Elements.

#### What Are the Tools For Creating PDF Files?

- The Adobe Acrobat family. Adobe Acrobat 7.0 Standard and Professional both include Acrobat Distiller for creating PDF files from PostScript files created with your favorite applications (including desktop publishing software).
- Acrobat Capture<sup>®</sup> with OCR (Optical Character Recognition) for creating text-searchable PDF documents from scanned paper originals.
- The Adobe LiveCycle<sup>™</sup> products, which provide core document services designed to align with your organization's services-oriented architecture and leverage open standards including J2EE and XML. They provide key functions such as document generation, document control and security, document collaboration, and process management for electronic forms. These products include LiveCycle Forms<sup>®</sup>.
- Adobe Document Server<sup>®</sup>. This software can be used to dynamically generate customized PDF documents. See [User and Developer Resources](#) for more information.

#### Related Adobe Technologies

- Adobe LiveCycle Reader Extensions<sup>®</sup> provides advanced services and functionality for documents and forms accessed by Adobe Reader clients, extending users' capabilities for participating in form applications or workflows without requiring the full Acrobat product.
- Adobe LiveCycle Policy Server<sup>®</sup> enables document authors and IT administrators to create usage policies that dynamically control who can view a PDF document, determine whether the recipient can modify, copy, print, or forward the document, and establish document expiration dates.
- Adobe LiveCycle Document Security<sup>®</sup> provides digital signature and encryption capabilities in a server environment, enabling your organization to securely automate essential business processes.

#### What Is the Acrobat SDK?

The Acrobat Software Development Kit contains samples (example source code), header files (to compile the samples), type libraries, simple utilities, and documentation.

These tools help you design and develop:

- External applications that communicate with and control Acrobat and Adobe Reader.
- Scripts written in JavaScript and used to control Acrobat and Adobe Reader.
- Acrobat plug-ins that extend the functionality of Acrobat Professional, Acrobat Standard, and Adobe Reader.

## Where Can I Get the Acrobat SDK?

There are two ways to obtain the complete Acrobat SDK:

- You may join the ASN Developer Program to receive the full SDK, developer technical support, marketing assistance, and other benefits.
- You may simply purchase an Acrobat SDK subscription for only US\$99.

The Acrobat SDK is available on the Adobe Solutions Network (ASN) Web site. Some portions are available for free, while others require either an ASN Developer membership or an Acrobat SDK subscription.

To access the Acrobat SDK, see the [Acrobat SDK Main Page](#).

## What Is a Plug-in?

Plug-ins are dynamically-linked extensions to Acrobat or Adobe Reader that enhance or customize its functionality. They can appear in the user interface in a variety of ways, and can respond to events that occur in the application.

A plug-in is a program written in ANSI C/C++ that uses the Acrobat public API. Its purpose is to add functionality to Acrobat Professional, Acrobat Standard, or Adobe Reader. A plug-in program file must be stored in the `Plug_ins` folder in the user's Acrobat or Adobe Reader installation, and its functionality becomes available when the Acrobat or Adobe Reader application is started.

There are three types of plug-ins:

1. Regular plug-ins. The samples in the SDK are of this type.
2. Reader-enabled plug-ins. These are plug-ins that are developed with permission from Adobe, and require special processing to load under Adobe Reader. For more information, see *Adobe Reader-enabled Plug-ins*.
3. Certified plug-ins. For more information, see [What Is a Certified Plug-in?](#)

Plug-ins are:

- DLLs on Windows. Note, however, that plug-in names must end in `.API`, not `.DLL`.
- Code fragments on Macintosh PowerPC.
- Shared libraries on UNIX®.

## What Is a Certified Plug-in?

A certified plug-in is considered the most secure type of plug-in available for usage with Acrobat or Adobe Reader. Certified plug-ins are reserved for Adobe only, and are tested extensively to ensure that they are in full compliance with Acrobat's security model. The Acrobat and Adobe Reader user interfaces provide a checkbox that may be used to ensure that only certified plug-ins load. In Acrobat or Adobe Reader, select **Edit > Preferences... > Startup** and check **Use only certified plug-ins**.

**NOTE:** There is currently no way for third-party plug-ins to be certified by Adobe.

## How Does a Script Written in JavaScript Differ From a Plug-in?

Acrobat JavaScript was originally developed for Acrobat Forms, and has been greatly expanded. You can use Acrobat JavaScript to:

- develop and process PDF forms
- customize the behavior and appearance of the PDF document
- facilitate online team review, implement security policies
- interact with databases and Web services
- customize the Acrobat application itself through access to its user interface components

While developers writing plug-ins have direct access to the Acrobat Core API, JavaScripts tend to be easier to write and implement, since they are developed using the editor and debugger that are included in Acrobat Professional. JavaScripts are also easier to distribute since they can be included directly within a PDF file, whereas plug-ins must be placed in the `Plug_ins` folder by either an installer or the user. Acrobat JavaScript can be used across multiple platforms, while plug-ins must be built on multiple platforms in order to handle certain platform-specific issues.

It is also worth noting the following comparisons between Acrobat JavaScript and plug-ins:

- Since it is interpreted rather than compiled, execution of Acrobat JavaScript code is slower than plug-in code. However, the difference tends to be noticeable only in computationally intensive applications, such as a full text search in a large PDF file.
- Acrobat JavaScript cannot be used to add content to a PDF file via a content stream.
- Acrobat JavaScript is well-suited to quickly adding user interface capabilities.

Finally, while there is a certain amount of overlap between what can be done in JavaScript and what can be done in a plug-in, some capabilities are only available in JavaScript while others are only available with a plug-in, as summarized in the table below. For more information, see the *Acrobat SDK User's Guide*.

Capability	JavaScript	Plug-in
SOAP and Web services	Yes	No

Capability	JavaScript	Plug-in
Multimedia	Yes	No
Set up automated email review workflow	Yes	No
Search Acrobat Help	Yes	No
Use Acrobat security policies	Yes	No
Content stream and other low-layer access	No	Yes
Create new menus or toolbars	No	Yes
Create new annotation or action types	No	Yes
Modify the scaling factor for large PDF file sizes	No	Yes
Access platform-specific services or events	No	Yes
Access wireframe drawing mode	No	Yes

## What Are Some Common Uses of the Acrobat SDK?

The Acrobat SDK enables you to customize or modify how Acrobat looks and operates, and contains the information necessary to extend Acrobat's functionality to meet your needs. It documents and demonstrates the Acrobat core API capabilities, how to use the API, and how to use and debug Acrobat JavaScript. Some common uses of the Acrobat SDK are described below. To understand the types of applications you can implement with the Acrobat SDK, see the *Acrobat SDK User's Guide*. To see examples of the types of applications that can be created with the Acrobat SDK, including ones specifically intended for Windows and Macintosh platforms, see the *Guide to SDK Samples*.

### PDF File Viewing

You can view a PDF file in your own application's window using Acrobat. For guidelines on how this may be done, see [How Can I Display a PDF File in an External Application Window?](#) You can also display a PDF as an ActiveX document in applications using simplified browser controls, by using an automation interface available in both Adobe Reader and Acrobat. For details, see the *Acrobat Interapplication Communication Overview* and *Acrobat Interapplication Communication Reference*.

#### PDF File Printing

You can customize how various portions and layers of a PDF file are printed using Acrobat JavaScript, Interapplication Communication (IAC), or plug-ins. For more information, see the *Acrobat SDK User's Guide*.

#### Customizing the User Interface

You can customize the user interface for specific workflows by writing a plug-in or JavaScript to add or change menu bars, menu items, or tool buttons that appear in the user interface. For more information, see the *Acrobat SDK User's Guide*.

#### Supporting Workflows

You can support custom workflows for PDF documents by:

- adding elements such as watermarks, bookmarks, and links
- creating custom templates
- automatically filling form fields
- adding metadata
- setting up custom searches and building catalogs
- managing page orientation and fonts  
setting up batch jobs
- automating PDF creation through control of Distiller
- merging or splitting documents
- directly accessing the file system to save and open PDF files to and from directory structures
- saving in various formats such as XML
- optimizing performance of PDF viewing
- applying stamps
- using security settings
- exchanging information with databases
- collaboration servers or Web services
- manipulating engineering drawings

For more information, see the *Acrobat SDK User's Guide*.

#### Controlling an Acrobat or Adobe Reader Session

You can remotely control an Acrobat Professional, Acrobat Standard, or Adobe Reader session. See the **DDEServer** sample plug-in for an example of how to communicate with an Acrobat plug-in from an external application using DDE, and the **RemoteControlAcrobat** Interapplication Communication sample. For more information, see the *Guide To SDK Samples*.

## Where Do I Start with the Acrobat SDK?

To help you get started with the Acrobat SDK, the following documents are included within the **Documentation** directory in the SDK installation:

- The *Acrobat SDK User's Guide* describes the capabilities of the Acrobat SDK, and provides a general overview of its usage.
- The *Acrobat SDK Documentation Roadmap* will help you find all the documentation for the Acrobat SDK.
- The *Acrobat SDK Release Notes* contain information relating to the latest release of the Acrobat SDK.
- *Guide To SDK Samples* provides a large number of sample applications, scripts, code snippets, and plug-ins to demonstrate how to use the APIs in the Acrobat SDK. Use this guide to locate appropriate samples.
- The *Acrobat JavaScript Scripting Guide* provides an overview of how to write Acrobat JavaScripts and how to use the JavaScript scripting environment.
- The *Acrobat Plug-In Guide* describes basic Acrobat plug-in development concepts, and shows how Acrobat interacts with plug-ins. It demonstrates how your plug-ins can manipulate and enhance the Acrobat and Adobe Reader user interface as well as the contents of underlying PDF documents.
- The *Acrobat Interapplication Communication Overview* describes Acrobat's facilities for interapplication communication (DDE and OLE automation on Windows, and AppleScript and Apple events on the Macintosh).
- The *Acrobat and PDF Library API Overview* provides a summary of the core API.
- *Developing for Adobe Reader* provides an introduction to those portions of the Adobe Acrobat Software Development Kit (SDK) that pertain to your development efforts for Adobe Reader.

## Where Can I Find Guidelines On Developing Adobe Reader Plug-ins?

If you are considering plug-ins for Adobe Reader, bear in mind the following:

- You may not develop an Adobe Reader plug-in without approval from Adobe. There is a Web-based application where you describe your plug-in and submit the information to Adobe; Adobe will then review it and let you know whether your application has been approved. The application and the Adobe Reader Integration Key Licensing Agreement can only be submitted as a Web form and are found at:  
<http://partners.adobe.com/asn/acrobat/index.jsp>
- There is a fee to obtain the enabling key.
- There is a restricted set of APIs available in Adobe Reader. For technical limitations, see the following documents:
  - *Acrobat and PDF Library API Overview*
  - *Acrobat Interapplication Communication Overview*
- For information on how to access the Host Function Tables (HFTs) available to Adobe Reader plug-ins, see the *Acrobat SDK Plug-in Guide*.
- For instructions on how to make your plug-in Adobe Reader-enabled, see *Adobe Reader-enabled Plug-ins*.

## What Are the Requirements For Using Acrobat Forms?

You may create and fill in Acrobat forms in PDF files, and submit or import form data. These forms are typically used for viewing or printing information, filling in information, selecting choices, digitally signing documents, and exchanging information with databases. Note that users of Adobe Reader can fill in and submit forms, and can save the form data to disk if they have LiveCycle Reader Extensions. Once a user fills out an Acrobat form, the data can be submitted to a server for processing. Data can be exported from a PDF form into Forms Data Format (FDF) or XML-based FDF (XFDF). Data can only be imported into a PDF form if it is in FDF or XFDF format. Acrobat forms support the following formats: FDF, XFDF, tab-delimited text, and XML.

Every FDF file contains a reference to a PDF file for which the data is intended, designated with the **/F** key inside the FDF file (unless the FDF file is for the same PDF from which you submitted your data). When Acrobat or Adobe Reader receives an FDF file, it opens the corresponding PDF file, and fills the form fields with the data from the FDF file. If the PDF file is referenced by a URL (for example, **http://yourcompany.com/file.pdf**), the FDF file must be sent to the server in response to a **submit** action from a PDF form. For more information on the FDF format, which is based on PDF, see the *PDF Reference*.

Acrobat plug-ins can programmatically import FDF data into a PDF file from a local file system using the HFT made available by the Forms plug-in. OLE Automation can be used to programmatically add, modify, or delete form fields, import or export FDF data, execute scripts written in JavaScript, and much more. For more information, see the *Acrobat and PDF Library API Reference*.

## What Are the Requirements For Using XML Forms?

The XML Forms Architecture (XFA) supports the production of XML-based business form documents known as dynamic XML forms. Dynamic XML forms are suited for exchanging information with Web services and enterprise infrastructures. Dynamic XML forms include the following features:

- fields that may be moved and dynamically resized even to the point of spanning multiple pages
- repeating subforms that can be spawned as needed with page contents shifting accordingly
- page elements that can be shown or hidden as needed

Use LiveCycle Designer<sup>®</sup> to create or edit templates. For more information, see <http://www.adobe.com/products/server/>.

Dynamic XML forms support both the XML and XDP formats. For more information, see the *Acrobat JavaScript Scripting Guide* and the *Acrobat JavaScript Scripting Reference*.

## What is the FDF Toolkit?

The FDF (Forms Data Format) Toolkit is a thread-safe API for writing a server application that generates FDF data or parses FDF data from a form created by the Acrobat Forms plug-in. If you need to parse FDF files submitted from a PDF form, or generate FDF files to be submitted to a PDF form, you can use the FDF Toolkit. The FDF Toolkit supports COM on Windows, C and Perl on Windows, Solaris<sup>™</sup>, AIX<sup>®</sup> or Linux<sup>®</sup>, and Java VMs compatible with versions 1.2 or higher.

For detailed information, see the documentation included with the FDF Toolkit.

---

## Licensing and Distribution

### What Should I Know About Acrobat Licensing and Distribution?

Some of the Acrobat products may be licensed for commercial redistribution.

For smaller quantities, the Adobe Acrobat Publisher Kit includes Acrobat and the right to reproduce and distribute up to 100 copies of the software. A flexible installer allows custom installation options on Windows systems and pre-installation of serial numbers. There is no minimum requirement for number of copies. If you would like to license a Publisher Kit, contact your Adobe reseller.

For larger quantities, licensing agreements are available from Adobe by special arrangement.

Products built around Acrobat and Adobe Reader must comply with the Acrobat End-User License Agreement (EULA). For example, customers who purchase third-party developer products that use Acrobat still must comply with the Acrobat EULA.

#### **Adobe Reader**

You may distribute Adobe Reader, subject to the terms set forth in the license agreement. However, you must use the installer that comes with Adobe Reader. You may invoke the Adobe Reader installer from your installer, but the Adobe Reader installer must remain intact and the End User License Agreement (EULA) must be displayed at first use. For information on customizing the Adobe Reader installer, see [Enterprise Installation](#).

If you would like to distribute Adobe Reader, see the Adobe Reader distribution page given below in [Additional Resources](#).

#### **Acrobat Capture (fee-based, license required)**

With Acrobat Capture, you can submit an image file to a Capture workflow, where it will be “captured,” converted to PDF, and made available to the user. Capture can also convert paper to PDF and includes OCR to recognize text; it preserves the visual layout of the document. Capture is a stand-alone product that is licensed separately. It is not included as part of the Acrobat product. Low-volume Capture requires a hardware dongle that counts the pages processed; dongles are available with various page limits. For OEM's and integrators, Adobe offers special pricing for Capture.

#### **Acrobat Toolkits**

- FDF Toolkit:  
Visit the Acrobat Family Developer Center at <http://partners.adobe.com/public/developer/acrobat/fdf/topic.html> to obtain the FDF Toolkit.
- Adobe PDF Library (fee-based, license required):  
The Adobe PDF Library is an object code library that can be linked to your application. The Adobe PDF Library is not free and is not available for free download from the Adobe public Web site. However, the following URL provides information about the product: <http://partners.adobe.com/asn/pdf/>.

#### **Additional Resources**

Adobe Reader distribution:

<http://www.adobe.com/products/acrobat/distribute.html?readstep>

Volume Software Licensing: <http://www.adobe.com/aboutadobe/openoptions/main.html>

## What Technical and Licensing Issues are Restricted in the Acrobat SDK?

The following is a list of Acrobat Software Development Kit uses that are not supported. Most activities are identified as *Technically Infeasible* and/or *Contrary to Licensing*.

- Use of any Acrobat product in a multithreaded way (*Technically Infeasible*).

Any multithreaded access to the Acrobat core API is likely to crash or hang the application. Acrobat makes the following methods available that can help a plug-in or application manage its thread's access to the Acrobat core API, as documented in the *Acrobat and PDF Library API Reference*:

### **AVAppRegisterNotification**

### **AVAppRegisterIdleProc**

When registering for a notification, the method is passed a function to be called by the Acrobat viewer when the event occurs. Registering for an **IdleProc** calls the function when nothing else is occurring. All notifications and **IdleProcs** are queued and called in order. Registering for notifications of events or **IdleProcs** can help a multithreaded application ensure that Acrobat is not accessed by multiple threads simultaneously; however, it is still the application's responsibility to manage its threads that access Acrobat.

- Use of any Acrobat product as a server process accessed by multiple clients, unless it is specifically stated in product documentation as designed and licensed for such purpose (*Technically Infeasible* | *Contrary to Licensing*).

Unless specifically stated in the product documentation and licensing agreement, Acrobat products are not designed, tested, or licensed for this purpose. This use is in violation of the licensing restrictions, as described in the End User License Agreement displayed during Acrobat installation and not supported by Acrobat Developer Support.

- Use of Distiller as a Windows NT service. (*Technically Infeasible*).

Acrobat Distiller requires the ability to open a window on the desktop to run. It is not possible to use Distiller as a Windows NT service.

**NOTE:** For customers needing this capability, the Acrobat 7.0 Professional product provides for *watched folders*.

- Use of **MenuItemExecute** to bring up Acrobat dialog boxes when a PDF file is displayed in an external window using OLE Automation (*Technically Infeasible*).

Due to problems of managing window focus, using the Acrobat dialog boxes (using **MenuItemExecute**) when a PDF file is displayed in an external window using OLE Automation is not supported. The actions executed by the dialog box may or may not affect the intended PDF file and there can be problems of windows not redrawing properly. The only supported workaround is to use the OLE Automation methods or to develop a plug-in to achieve any functionality not available in the OLE API to Acrobat. The [Adobe Reader Integration Key License Agreement](#) only permits displaying in an external window when Acrobat is used, not Adobe Reader.

- Use of the *PDF Reference* to develop a third-party application that writes PDF files without the use of Acrobat products.

**NOTE:** The Adobe PDF Library, available under license, can be used to simplify development of these types of applications.

We do not provide support to developers who are creating their own PDF generation capabilities without the use of Adobe products. The *PDF Reference* is the best resource for this kind of development. The use of Adobe products to create PDF files as a benchmark for your own development is recommended. Acrobat Developer Support will not debug PDF files created with non-Adobe products. Questions regarding the completeness or accuracy of the *PDF Reference* will be answered. The *PDF Reference* is available for free download from the [Adobe Public Web Site](#).

- Interapplication communication (IAC) between a plug-in and a third-party application. Interapplication communication between a plug-in and a third-party application does not differ significantly from interapplication communication between two stand-alone applications. Documentation for your development platform's API and development environment are the best resources for this type of development. The Acrobat SDK contains two samples that can serve as examples.
  - **DDEServer** demonstrates DDE communication between a stand-alone application and a plug-in.
  - **ExternalWindow** uses Windows messaging to communicate with a stand-alone application.

One typical difficulty for developers occurs when a multithreaded stand-alone application communicates with a plug-in. See [User and Developer Resources](#) for more information.

- Use of platform API methods or API methods of applications other than Acrobat. Acrobat Developer Support supports the API to the Acrobat family of products. Questions regarding the use of platform API methods should be directed to the manufacturer of your operating system.
- Use of the ActiveX<sup>®</sup> Control or Netscape plug-in to display a PDF file in an external application besides Internet Explorer or Netscape. The methods used by Acrobat to display a PDF file in Netscape and Internet Explorer are intended only for use with these browsers. Use of the ActiveX Control and Netscape plug-in installed by Adobe Reader is not licensed to other applications. Development with these interfaces is not supported and no documentation is available.
- Automating importing image files using the Import plug-in to Acrobat (*Technically Infeasible*).

The Import plug-in to Acrobat does not provide an API that allows it to be called from a plug-in or another application. Executing the **Import Image** menu item with **MenuItemExecute** brings up a dialog box requiring user input.

For information about other developer resources, see [User and Developer Resources](#).

## What ActiveX Solutions Are Provided By the Acrobat SDK?

The PDF browser control automation object, described in the *Acrobat Interapplication Communication Overview* and *Acrobat Interapplication Communication Reference*, supports the loading and displaying of PDF documents as ActiveX documents, and its interface is available in both Acrobat and Adobe Reader.

---

## User and Developer Resources

### What Do I Need to Download from the Web to Get the Acrobat SDK?

The Acrobat SDK is made up of many different technical documents and samples for different operating systems. You can download the installer from the Web site to obtain the entire Acrobat SDK, or you can download each piece separately using the “exploded” version. You will need to determine what you would like to do with the SDK, and then check if there is a sample that gets you started in the right direction.

See the [Acrobat SDK Main Page](#) to access the Acrobat SDK. The Web site uses lock icons next to individual portions of the SDK to indicate that ASN membership or a subscription is required for access.

### Where Can I Get Help With Acrobat Product Issues?

For information and support for the Acrobat and LiveCycle products, you can visit the following sites:

Acrobat Product Information: <http://www.adobe.com/products/acrobat/>.

Acrobat Product Support: <http://www.adobe.com/support/products/acrobat.html>.

Adobe LiveCycle Product Information: <http://www.adobe.com/products/server/>.

For information on using or installing Adobe Acrobat products, start at the [Acrobat Product Support Page](#), which contains free downloads of the latest Acrobat product updates, answers to top product support issues, a searchable support database, user forums, and product support telephone numbers.

For International customers, see the [Adobe International Technical Support and Service Page](#).

## Acrobat SDK Development Support Resources

In general, Acrobat Developer Support supports software development with the Acrobat core application programming interface (API), as documented in the Acrobat SDK. Non-ASN members can buy support cases, except in Japan. ASN Premium members receive some free cases with their membership. For more information, see the [ASN Developer Program Home Page](#). For Adobe Developer Support information, see <http://partners.adobe.com/asn/support/index.jsp>

Acrobat Developer Support does not support use of the product that does not involve the Acrobat core API. Any non-programmatic issues, such as questions about installing, using, customizing, or deploying Acrobat, should be directed to Acrobat Technical Support rather than Adobe Developer Support. The supported development activities using the Acrobat SDK include those for which the product is designed, tested, and licensed.

**NOTE:** Each SDK version will continue to be supported for one major release after the next SDK version is released.

## How Can I Contact Adobe For Technical Support Issues?

If you have a technical support issue and do not find your answer online, you can contact Adobe Acrobat Technical Support at the telephone numbers listed on the Technical Support Phone Numbers Page:

<http://www.adobe.com/support/phonenumbers/main.html>.

Technical support for the Acrobat SDK (“Acrobat Developer Support”) is included with an ASN membership. It is available to Adobe Solutions Network (ASN) Developer Program Premium members or may be purchased (except in Japan). For more information, see the [ASN Developer Program Home Page](#).

**NOTE:** The UNIX version of the Acrobat SDK is available from <http://partners.adobe.com/asn/acrobat/index.jsp> (the Acrobat SDK Main Page). It has release notes containing UNIX-specific information.

You will find answers to many common technical issues in the Acrobat Developer Knowledge Base at <http://support.adobe.com/devsup/devsup.nsf/main.html>.

## What Is the ASN Developer Program and How Do I Join?

The Adobe Solutions Network Developer Program includes marketing tools and technical tools. For complete information on benefits the ASN Developer Program, visit the Adobe Solutions Network Page at <http://partners.adobe.com/>.

For developer information related to Adobe LiveCycle products, see <http://partners.adobe.com/public/developer/livecycle/devcenter.html>.

### Contact Information

To contact Adobe Systems, visit the ASN Developer Program contact page at <http://partners.adobe.com/asn/programs/developer/contactus.jsp>.

### **Technical Support**

Support for the Adobe Acrobat SDK is available to Adobe Solutions Network (ASN) Developer Program Premium members or may be purchased (except in Japan).

### **Technologies Supported**

The Acrobat core API; Interapplication Communication (IAC), including Apple events; DDE; Adobe-generated Portable Document Format (PDF); **pdfmark** operator; command-line and initialization file interfaces; installer customization.

### **Regular Notification of Updates and Upgrades**

Members receive e-mail regarding updates to technical notes, development tools, and development software.

### **Acrobat Software Development Kit (SDK)**

See [What Is the Acrobat SDK?](#)

## **What Adobe Resources Are Recommended to Acrobat Developers?**

### [Information on Partner Resources](#)

Provides plug-ins and applications available from Adobe or third parties, Acrobat/PDF service providers, or links to pages on how to use the available plug-ins.

### [Acrobat Main Page](#).

Provides white papers and case studies of what can be done and what has been done with Acrobat.

## **What Developer Resources Are Provided by Outside Companies?**

[blueworld ListSearch](#) provides search capabilities to Acrobat discussions.

[PDFZone](#) provides resources that may be of help with developing and using the Acrobat products. These resources include forums, mailing lists, and an archive of previous mail.

---

## Acrobat and Adobe Reader Comparison

### What Are the API Differences Between Acrobat and Adobe Reader?

Acrobat provides a “full-featured” development environment that includes the entire Acrobat core API. There are some small differences between the public APIs available in Acrobat Professional and Acrobat Standard. These are documented in the *Acrobat SDK Plug-in Guide* and the *Acrobat and PDF Library Reference*.

The APIs that may be used for Adobe Reader are limited technically and legally. Technical limitations are documented in the *Acrobat and PDF Library API Overview* and the *Acrobat Interapplication Communication Overview*.

Both Acrobat and Adobe Reader accept plug-ins. The primary difference between the two is that Adobe Reader can neither make changes to a file nor save a file. API methods that change a file in such a way that a save would be required are not available in Adobe Reader.

### Developing for Adobe Reader

Adobe Reader only accepts “Reader-enabled” plug-ins, which can only access a limited set of APIs. For information on enabling plug-ins for Adobe Reader, see *Adobe Reader-enabled Plug-ins* and *Developing for Adobe Reader*. Certain APIs are available to a Reader-enabled plug-in if the PDF document has been assigned additional usage rights, which is possible using LiveCycle Reader Extensions. For more information on this, see the *Acrobat SDK User’s Guide*.

For additional legal restrictions, see the [Adobe Reader Integration Key License Agreement](#).

### What Is a Rights-enabled PDF Document?

A rights-enabled PDF document is used by people with the free and ubiquitous Adobe Reader to download, save, fill in, digitally sign, and submit electronic PDF forms that have been “enabled” using LiveCycle Reader Extensions. With a rights-enabled PDF document, developers are able to use the digital signature APIs for signing and signature verification. One major advantage of LiveCycle Reader Extensions is that it supports data automation through XML-based representation and transfer using SOAP, thus ensuring seamless integration with business logic and advanced data transport capabilities available within enterprise applications. For more information, see the *Acrobat SDK User’s Guide*.

---

## Visual Basic .NET and Visual C# .NET

### How Can I Use Visual Basic .NET or Visual C# .NET With the Acrobat SDK?

The Microsoft Windows version of Acrobat is an OLE Automation server. In order to use the OLE objects made available by Acrobat, you must have the full Acrobat product installed on your system and the `Acrobat.tlb` file included in the project references for your Visual Basic project. This allows you to use the Visual Studio .NET 2003 object browser to browse the OLE objects.

For Acrobat versions 5.0 and higher, it is possible to access Acrobat's JavaScript object model through the OLE Automation `JSObject` interface. See the `JSObjectAccess` sample. For more information, see *Programming Acrobat JavaScript Using Visual Basic* and the *Acrobat Interapplication Communication Overview*.

### What Are the Best Resources For Getting Started Using Visual Basic .NET or Visual C# .NET With the Acrobat SDK?

The *Acrobat Interapplication Communication Overview* and *Acrobat Interapplication Communication Reference* describe the objects and methods available in these languages, as well as guidelines for usage. These documents (as well as the API) were designed with C programming in mind, and programming with the API requires some familiarity with C concepts such as `enum`.

Besides the object browser, the best resources for programming in these languages are the sample projects. The samples demonstrate use of the Acrobat OLE objects and contain comments describing the parameters for the more complicated methods.

The `Iac.bas` file in the `InterAppCommunicationSupport\Headers` folder contains the Visual Basic counterparts to the C `#define` statements and enumerations passed to many of the API methods.

---

## Enterprise Installation

### How Can I Customize the Acrobat Installer?

Adobe provides various ways in which you can deploy Acrobat to a large number of systems.

You can find documentation about Enterprise Installation, including information about tools for customization and installation as well as guidelines for extending enterprise applications, at: <http://www.adobe.com/products/acrobat/planning.html>.

For technical notes to deploy Acrobat and Adobe Reader throughout your entire enterprise, see <http://partners.adobe.com/asn/acrobat/technotes.jsp>.

Solutions for enterprise deployment are provided at <http://www.adobe.com/products/acrobat/deployment.html>, where you will find a variety of technical information on enterprise deployment, including the following:

- Citrix User Document — An explanation of how to use the Citrix Independent Computing Architecture (ICA) protocol (via the Citrix MetaFrame 1.8 add-on) with Acrobat.
- SMS User Document — Information on deploying Acrobat to the enterprise via SMS.
- WTS User Document — Information on deploying Acrobat to the enterprise via Windows Terminal Service.
- InstallShield Tuner for Acrobat — Enterprise installation support for Acrobat.
- Tivoli User Document — Support for enterprise deployment of Acrobat Professional, Acrobat Standard, Adobe Reader, and Acrobat Elements using Tivoli.

---

## PDF Documents

### What API Methods Are Available to Modify PDF Documents?

Modifying the page contents of a PDF file is primarily accomplished using the Acrobat core API.

Using the Acrobat core API greatly simplifies modifying and creating PDF page contents. To demonstrate this, there are several “snippets” available from the `SnippetRunner` sample plug-in that show you how to add data to the contents of a page, while ensuring that the PDF file is still readable after modification. To attempt to do this without using the core API would be significantly more difficult and could result in an unreadable PDF file.

Acrobat’s automation interfaces are limited mainly to what a user can do through the user interface and cannot modify the contents of a page.

For more information on the Acrobat core API, see the *Acrobat and PDF Library API Overview* and *Acrobat and PDF Library API Reference*.

### What Options Do I Have to Modify PDF Documents Without C Programming Background?

You can make some modifications to PDF files through Acrobat JavaScript. The *Acrobat JavaScript Scripting Guide* and *Acrobat JavaScript Scripting Reference*, freely available for download from the Adobe Web site, provide all the information you need to get started using JavaScript. They serve as a reference to Acrobat’s JavaScript objects, properties, and methods.

### How Can I Extract Text From PDF Documents Using the Acrobat SDK?

You can extract text with the Acrobat SDK in two ways:

- Use the Acrobat core API
- Use Acrobat’s automation API

Through the core API, you can extract ASCII text from a PDF file using Acrobat and a plug-in developed in C or C++. The `TextExtraction` and `WordFinder` sample plug-ins demonstrate text extraction and can be used as starting points for your own plug-in.

**AVConversion** methods also can be used to save PDF as text or rich text. In addition, the **SDKTextExtraction** sample in the first level of the **SaveAsXML** directory provides a good starting point for creating richer extraction tables. For more information, see the *Guide To SDK Samples*.

## How Can I Display a PDF File in an External Application Window?

There are several ways to have the Acrobat program display a PDF file in an external application's window. Acrobat must be installed on the system to view a PDF file in your own application's window.

There is no best way that we suggest to display PDF files in your application; you should examine the following list for the most appropriate method for your situation. For more information, see *Acrobat Interapplication Communication Overview* and *Acrobat Interapplication Communication Reference*.

### Using Acrobat to View a PDF File in Your Own Application's Window

Windows Only:

- OLE automation, using the **OpenInWindowEx** command. This displays a live view of the PDF file in the OLE application window.  
For samples, see the Visual Basic and C++ **ActiveView** applications.
- OLE automation, using the **DrawEx** command. This displays a bitmap of the current page in the OLE application window.  
For samples, see the Visual Basic and C++ **StaticView** applications.
- Copy to Clipboard. This copies a PDF image to the clipboard without requiring an **hWnd** or **hDC** using OLE Automation.
- If you are using simplified browser controls in your application, you may treat the PDF document as an ActiveX document by using the **AcroPDF** object's **LoadFile** method. This automation interface is available in both Acrobat and Adobe Reader.

Macintosh Only:

- The **AcroAppleEvents** sample application demonstrates rendering a PDF file into another application's window (see **DrawIntoWindowCommand** for details).

Windows and Macintosh:

- The **ExternalWindow** sample plug-in demonstrates a live view of the PDF file in a window created by the plug-in. You can extend this to display PDF files in an external application window.

## Are Multibyte Font PDF Documents Supported by the Acrobat SDK?

Acrobat allows the creation, modification, and use of PDF files containing multibyte fonts. For more information, see the *Acrobat SDK User's Guide*.

## How Are Security and Encryption Provided in PDF Documents?

Adobe provides an password-based and certificate-based encryption schemes with Acrobat. In the password-based scheme, Acrobat queries the user for a password before a file is opened. The authorization procedure to open a file or to set an owner password can be modified by creating a custom security handler. For more information on security handlers and encryption in PDF documents, see the *Acrobat SDK User's Guide*.

---

## Full-Text Search

### What Tools Are Available with Acrobat For Full-Text Search?

Adobe provides a full-text search system based on third-party technology. However, this does not preclude other search systems from integrating with Acrobat. The Acrobat search system was created using only public API and IAC calls, and you can easily remove it from Acrobat and replace it with another search technology. For more information, see the *Acrobat Interapplication Communication Reference*.

The Acrobat interface can be extended with new menus and toolbar icons to allow tight integration with your search plug-in and Acrobat. For example, buttons to invoke a search and to find the next or previous occurrences can be added to the toolbar.

For samples of adding menu items and toolbar buttons, see the related snippets in the `SnippetRunner` sample plug-in.

### What Tools Are Available for Extracting and Highlighting Text?

For indexing and searching PDF files directly, Acrobat provides support through IAC and plug-in calls. For indexing PDF files, Acrobat provides text extraction APIs. For further information on extracting text from PDF, see [How Can I Extract Text From PDF Documents Using the Acrobat SDK?](#)

## How Do I Communicate with the Acrobat Search Plug-in?

You can communicate with the Search plug-in via its plug-in API or via IAC (DDE or Apple events). Using either of these methods, you can control the Acrobat Search plug-in in the following ways:

- Control the list of indexes to search  
The search interface allows for searches to be performed on one or more of the available indexes. You can control the list of active indexes.
- Initiate a search with options  
You can pass query expressions and search settings to the Acrobat search plug-in and initiate the search, and the results will be presented to the user.

For an example, see the **SearchPDF** Visual Basic application. For documentation, see the *Acrobat and PDF Library API Reference*.

## How Do I Create Custom DocInfo Fields For Searching?

The Catalog feature is used to create a search index; its API is available to third parties via the Catalog plug-in provided with the Acrobat SDK. See the *Acrobat and PDF Library API Reference* and the *Acrobat Interapplication Communication Reference* for more information.

Once you have built the search index, you must create the custom **DocInfo** fields, which can be used to search document metadata. You can accomplish this task with either the Acrobat core API or using Interapplication Communication. See the *Acrobat and PDF Library API Reference* for more information.

At this point you may submit custom queries to the Acrobat Search plug-in to search by the custom fields, as demonstrated by the **SearchQuery** sample plug-in. You can include your custom **DocInfo** fields in the search by specifying them in the **sortSpec** parameter of the **SearchExecuteQuery** method.

## How Do I Use Command Lines with Acrobat and Adobe Reader on Windows?

These are unsupported command lines, but have worked for some developers. There is no documentation for these commands other than what is listed below. You can display and print a PDF file with Acrobat and Adobe Reader from the command line.

**NOTE:** All examples below use Adobe Reader, but apply to Acrobat as well. If you are using Acrobat, substitute **Acrobat.exe** in place of **AcroRd32.exe** on the command line.

**AcroRd32.exe *pathname*** — Executes Adobe Reader and displays the file, whose full path must be provided.

Other options for the command line are:

Option	Meaning
<b>/n</b>	Launches a separate instance of Acrobat or Adobe Reader, even if one is currently open.
<b>/s</b>	Opens Acrobat or Adobe Reader, suppressing the splash screen.
<b>/o</b>	Opens Acrobat or Adobe Reader, suppressing the open file dialog.
<b>/h</b>	Opens Acrobat or Adobe Reader in a minimized window.

**AcroRd32.exe /p *pathname*** — Executes Adobe Reader and displays the **Print** dialog box.

**AcroRd32.exe /t *path* "*prntername*" "*drivername*" "*portname*"** — Initiates Adobe Reader and prints a file, whose path must be fully specified, while suppressing the **Print** dialog box.

The four parameters of the **/t** option evaluate to **path**, **prntername**, **drivername**, and **portname** (all strings).

- **prntername** — The name of your printer.
- **drivername** — Your printer driver's name, as it appears in your printer's properties.
- **portname** — The printer's port. **portname** cannot contain any "/" characters; if it does, output is routed to the default port for that printer.

