Foxit PDF SDK

DEVELOPER GUIDE
FOXIT PDF CONVERSION SDK

For .NET

©Foxit Software Incorporated. All Rights Reserved.

TABLE OF CONTENTS

1 Introduction to Foxit PDF Conversion SDK 1

1.1 Why Choose Foxit PDF Conversion SDK 1

1.2 Foxit PDF Conversion SDK for .NET 1

1.3 Evaluation 2

1.4 License 2

1.5 About this guide 2

2 Getting Started 3

2.1 System Requirements 3

2.2 What is in the package 3

2.3 How to run a demo 4

2.4 How to initialize Foxit PDF Conversion SDK 5

FAQ 6

Support 7

1 INTRODUCTION TO FOXIT PDF CONVERSION SDK

Foxit PDF Conversion SDK is a flexible high-performance library for conversion between PDF files and MS Office files while maintaining the layout and format of your original documents. It offers advanced conversion capabilities as well as the same quality and user experience you can expect from Foxit. The engines for creating MS office files or PDF files are developed independently by Foxit and no 3rd party office engine is required to be pre-installed on your machine. Foxit PDF Conversion SDK can be used as a standalone library or as an add-on with Foxit PDF SDK.

1.1 Why Choose Foxit PDF Conversion SDK

Foxit is a leading software provider of solutions for reading, editing, creating, organizing, and securing PDF documents. Foxit PDF SDK libraries have been used in many of today's leading apps, and are proven, robust, and battle-tested to provide the quality, performance, and features that the industry's largest apps demand. Foxit PDF Conversion SDK is a new SDK product independently developed by Foxit. Customers choose Foxit PDF Conversion SDK for the following reasons:

- **Easy to integrate**

Developers can seamlessly integrate Foxit PDF Conversion SDK into their own applications.

- **Lightweight footprint**

Does not exhaust system resource and deploys quickly.

- **Independently-developed engine**

The core technology of the Conversion SDK is developed independently by Foxit.

- **Premium World-side Support**

Foxit offers premium support for its developer products because when you are developing mission critical products you need the best support. Foxit has one of the PDF industry's largest team of support engineers. Updates are released on a regular basis to improve user experience by adding new features and enhancements.

1.2 Foxit PDF Conversion SDK for .NET

In this guide, we focus on the introduction of Foxit PDF Conversion SDK for .NET API on Windows platform. Foxit PDF Conversion SDK for .NET API ships with simple-to-use APIs that can help .NET developers do conversion between PDFs and MS Office files (only support docx, pptx and xlsx format files).

1.3 Evaluation

Foxit PDF Conversion SDK allows users to download a trial version to evaluate the SDK. The trial version has no difference from a standard version except for the 14-day limitation trial period. After the evaluation period expires, customers should contact Foxit sales team and purchase licenses to continue using Foxit PDF Conversion SDK.

1.4 License

Developers should purchase licenses to use Foxit PDF Conversion SDK in their solutions. Licenses grant users permissions to release their applications based on PDF SDK libraries. However, users are prohibited to distribute any documents, sample codes, or source codes in the SDK released package to any third party without the permission from Foxit Software Incorporated.

1.5 About this guide

This guide is intended for developers who need to integrate Foxit PDF Conversion SDK for .NET into their own applications. It aims at introducing the installation package, and the usage of SDK.

2 GETTING STARTED

It's very easy to setup Foxit PDF Conversion SDK and see it in action! This guide will provide you with a brief introduction on how to integrate Foxit PDF Conversion SDK into the projects based on .NET Framework 4.0 or higher. The following sections introduce the contents of system requirements, the installation package as well as how to run a demo, and how to initialize Foxit PDF Conversion SDK.

2.1 System Requirements

Platform	System Requirement	Note
Windows	Windows Vista, 7, 8 and 10 (32-bit and 64-bit) Windows Server 2003, 2008 and 2012 (32-bit and 64-bit)	Visual Studio 2010/2015/2017 installed with .NET Framework 4.0 or higher. It only supports for Windows 8/10 classic style, but not for Store App or Universal App.

Note: Starting from version 1.1, Foxit PDF Conversion SDK supports machine learning-based recognition functionality which can identify borderless tables in PDF documents. By default, it is disabled, if you want to enable it, please make sure the network is connected first.

2.2 What is in the package

In this guide, the highlighted rectangle in the figure is the version of the SDK. Here the SDK version is 2.0, so it shows 2_0.

Download the Foxit PDF Conversion SDK zip for .NET package and extract it to a new directory "foxitpdfconversionsdk_2_0_win_dotnet", which is shown in Figure 2-1.

Note: You can also get this package from NuGet.

The release package contains the following folders:

doc: API references, developer guide

examples: sample demo

lib: libraries and license files

res: font-related configuration files

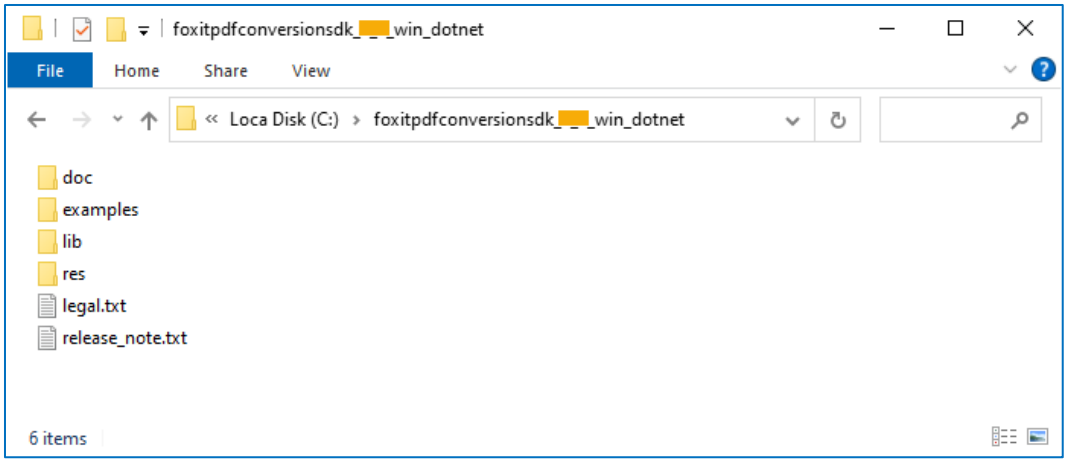


Figure 2-1

Note:

- In "res\metrics_data" folder, it contains the metrics data files which are used to simulate the Office format document typesetting process during PDF to Office conversion.
- In "res\office2pdf" folder, it contains the resource files which are used to simulate the Office format document typesetting process during Office to PDF conversion.

2.3 How to run a demo

Foxit PDF Conversion SDK provides a pdf2office and an office2pdf demo in the "examples" folder. To run the demos in Visual Studio, you can follow the steps below:

1) Load the visual studio solution file "simple_demo_vs2010.sln" or "simple_demo_vs2015.sln" or "simple_demo_vs2017.sln" or "simple_demo_vs2019.sln" or "simple_demo_vs2022.sln" (depending on your Visual Studio version) in the "\examples\simple_demo" folder.

2) Build the two demos by clicking "Build > Build Solution". Alternatively, if you merely want to build a single demo, you can right-click it and then choose "Build" or load the "*.csproj" file in the folder of a specific demo project and then build it.

   After building, the executable file ".exe" will be generated in the "\examples\simple_demo\bin" folder. The names of the executable file depend on the build configurations. For office2pdf demo, please copy all the files under "res\office2pdf" folder to the "\examples\simple_demo\bin" folder.

3) Run a specific executable file by double-clicking it.

The output files (Word, Excel, PowerPoint, or PDF) will be generated in a folder named by the project name under "examples\simple_demo\output_files\" folder.

***Note**: If you want to see the detailed executing processes, you can run it in command line. Start "cmd.exe", navigate to "\examples\simple_demo\bin", and run the executable file.*

2.4 How to initialize Foxit PDF Conversion SDK

It is necessary for apps to initialize Foxit PDF Conversion SDK using a license before calling any APIs. The trial license files can be found in the "lib" folder.

```
string sn = " ";
string key = " ";
ErrorCode error_code = Library.Initialize(sn, key);
if (error_code != ErrorCode.e_ErrSuccess)
{
  return;
}
```

***Note** The value of "sn" can be got from "**conversionsdk_sn.txt**" (the string after "SN=") and the value of "key" can be got from "**conversionsdk_key.txt**" (the string after "Sign=").*

FAQ

1. How to fix the "'xcopy' exited with code 9009" error when building demo in Visual Studio?

   When building demo in Visual Studio, if you encounter the error "'xcopy' exited with code 9009" as follows:

   ```
   'xcopy ..\..\..\..\..\..\lib\conversionsdk_sn.txt..\..\..\ /y > null
   xcopy ..\..\..\..\..\..\lib\conversionsdk_key.txt..\..\..\ /y > null
   xcopy ..\..\..\..\..\..\lib\$(PlatformName)_vc10\fpdfconversionsdk.dll..\..\..\ /y > null
   xcopy ..\..\..\..\..\..\lib\$(PlatformName)_vc10\fpdfconversionsdk_dotnet.dll..\..\..\ /y > null' exited with code 9009
   ```

   Please check the following points:

   1) Check whether the **xcopy.exe** is in the *"%SystemRoot%\System32"* directory, if not, copy one from another machine.

   2) Check whether the system **PATH** environment variables have been set correctly. It should contain *"%SystemRoot%\System32;%SystemRoot%;"*, if the environment variables for **xcopy** is right, but it still reports the error, please put the path of **xcopy** in front of others. Often other environment variables have spelling mistakes which causes the subsequent environment variables to be invalid. Please check it.

   After checking, open a command prompt, type the xcopy command, if it can be recognized, close Visual Studio, and restart the demo. The error should be fixed.

SUPPORT

Foxit Support

In order to provide you with a more personalized support for a resolution, please log in to your Foxit account and submit a ticket so that we can collect details about your issue. We will work to get your problem solved as quickly as we can once your ticket is routed to our support team.

You can also check out our Support Center, choose Foxit PDF SDK which also has a lot of helpful articles that may help with solving your issue.

Phone Support:

Phone: 1-866-MYFOXIT or 1-866-693-6948