

foxit PDF SDK

开发者指南
FOXIT PDF
CONVERSION SDK

For JAVA

TABLE OF CONTENTS

1	Foxit PDF Conversion SDK 简介.....	1
1.1	为什么选择 Foxit PDF Conversion SDK.....	1
1.2	Foxit PDF Conversion SDK for Java.....	1
1.3	评估.....	1
1.4	授权.....	2
1.5	关于此文档.....	2
2	入门指南.....	3
2.1	系统要求.....	3
2.2	包结构说明.....	3
2.3	运行 demo.....	4
2.3.1	使用批处理脚本运行 demo.....	4
2.3.2	使用 Eclipse IDE 运行 demo.....	5
2.4	初始化 Foxit PDF Conversion SDK.....	5
	技术支持.....	7

1 FOXIT PDF CONVERSION SDK 简介

Foxit PDF Conversion SDK 是一个灵活的高性能库，用于在 PDF 文件和 MS Office 文件之间进行转换，同时保持原始文档的布局和格式。其提供了先进的转换功能，以及与福昕其他 SDK 产品同等的质量和用户体验。用于创建 MS Office 文件或者 PDF 文件的引擎由福昕独立开发，无需安装任何第三方引擎。Foxit PDF Conversion SDK 可以作为独立的库使用，也可以作为 Foxit PDF SDK 的插件使用。

1.1 为什么选择 Foxit PDF Conversion SDK

福昕是领先的 PDF 软件解决方案供应商，专注于 PDF 显示、编辑、创建、管理以及安全方面。Foxit PDF SDK 开发库已在当今许多知名的应用程序中使用，并且经过长期的测试证明 Foxit PDF SDK 的质量、性能和功能正是业界大部分应用程序所需要的。Foxit PDF Conversion SDK 是一款由福昕自主研发的全新 SDK 产品。选择 Foxit PDF Conversion SDK 产品的几大理由：

- **易于集成**

开发人员可以将 Foxit PDF Conversion SDK 无缝集成到他们自己的应用程序中。

- **轻量级**

部署简单快速，占用系统资源少。

- **引擎自主研发**

Foxit PDF Conversion SDK 的核心技术由福昕独立自主研发。

- **优秀的技术支持**

福昕对自己的开发产品提供了优秀的技术支持，当您在开发关键重要的产品时，可以提供高效的帮助和支持。福昕拥有一支 PDF 行业优秀的技术支持工程师团队，同时将定期地进行版本更新发布，通过添加新的功能和增强已有的功能来提升用户体验。

1.2 Foxit PDF Conversion SDK for Java

本指南将主要介绍 Windows 和 Linux 平台上的 Foxit PDF Conversion SDK for Java API。Foxit PDF Conversion SDK for Java API 提供简单易用的 API，帮助 Java 开发人员在 PDF 和 MS Office 文件 (只支持 docx、pptx 和 xlsx 的文件格式) 之间进行转换。

1.3 评估

用户可申请下载 Foxit PDF Conversion SDK 的试用版本进行试用评估。试用版除了有试用期 14 天时间的限制以外，其他都和标准认证版一样。当试用期到期后，用户需联系福昕销售团队并购买 licenses 以便继续使用 Foxit PDF Conversion SDK。

1.4 授权

程序开发人员需购买 licenses 授权才能在其解决方案中使用 Foxit PDF Conversion SDK。Licenses 授予用户发布基于 Foxit PDF Conversion SDK 开发的应用程序的权限。然而，在未经福昕软件公司授权下，用户不能将 Foxit PDF Conversion SDK 包中的任何文档、示例代码以及源代码分发给任何第三方机构。

1.5 关于此文档

此文档适用于需要使用 Java 开发语言将 Foxit PDF Conversion SDK 集成到自己的应用程序中的开发人员。它旨在介绍 SDK 包结构和 SDK 的用法。

2 入门指南

安装并集成 Foxit PDF Conversion SDK 非常简单。本手册将提供 SDK 包的简要介绍。本章的主要内容是介绍系统要求、SDK 包结构、以及如何运行 demo 和初始化 Foxit PDF Conversion SDK。

2.1 系统要求

平台	系统要求	JDK 版本	备注
Windows	Windows Vista, 7, 8 and 10 (32-bit and 64-bit) Windows Server 2003, 2008 and 2012 (32-bit and 64-bit)	1.8 或更高版本	仅支持 Windows 8/10 经典样式，不支持 Store APP 和 Universal App。
Linux	x86/x64 (32-bit 和 64-bit OS) armv7/armv8 Foxit PDF Conversion SDK for Linux (x86 和 x64) 支持的最低 GCC 编译器版本为 gcc4.9.4。	1.8 或更高版本	所有的 Linux (x86/x64) 示例在 Ubuntu14.0 32/64 bit 上进行过测试。 所有的 Linux (armv7/armv8) 示例都在 armv7 或者 armv8 系统中进行过测试。

备注:

- 从 1.1 版本开始，Foxit PDF Conversion SDK 支持基于机器学习的识别功能，可以识别 PDF 文档中的无边框表格。默认情况下，该功能是禁用的，如果想要启用它，请首先确保网络是连接的。
- 当前，PDF2Office 支持 Windows 和 Linux 平台，Office2PDF 只支持 Windows 平台。

2.2 包结构说明

在本手册中，图片上面高亮的矩形区域指的是 SDK 的版本号，当前 SDK 的版本是 2.0，则其代表 2_0。

Windows Java 包命名为 "foxitpdfconversionsdk_2_0_win_java.zip"，Linux (x86/x64) Java 包命名为 "foxitpdfconversionsdk_2_0_linux_java.zip"，Linux (armv7/armv8) Java 包命名为 "foxitpdfconversionsdk_2_0_linux_arm_java.zip"。它们具有相同的包结构，因此在本手册中主要以 "foxitpdfconversionsdk_2_0_win_java.zip" 为例来进行介绍。

下载 Foxit PDF Conversion SDK zip for Windows Java API 包，解压到一个新的目录如 "foxitpdfconversionsdk_2_0_win_java"，如 Figure 2-1 所示。其中解压包中包括如下的内容：

doc: API 手册，开发者指南

examples: 示例 demo

- lib:** SDK 库和授权文件
- res:** 与字体相关的配置文件

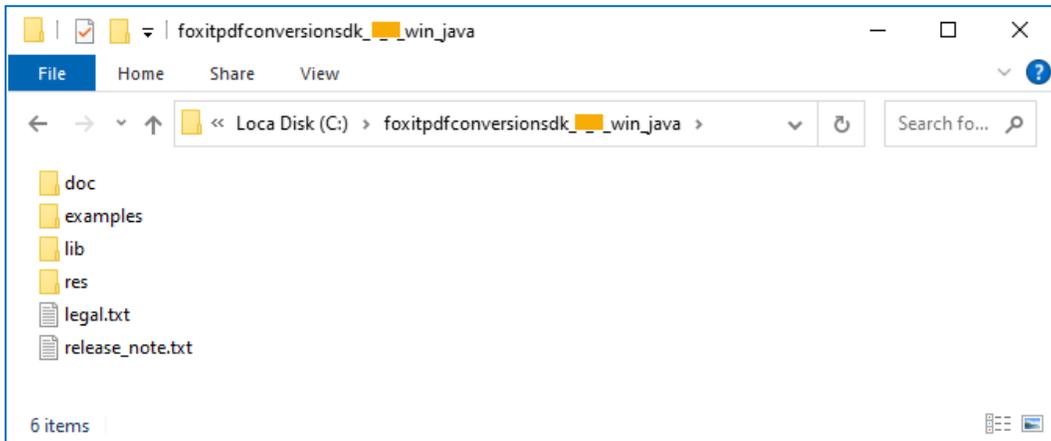


Figure 2-1

备注:

- "res\metrics_data" 包含了用于在将 PDF 转换为 Office 文档时模拟 Office 格式排版过程所需的文件。
- "res\office2pdf" 包含了用于在将 Office 转换为 PDF 文档时模拟 Office 格式排版过程所需的文件。

2.3 运行 demo

以下列举了当前 Foxit PDF Conversion SDK for Linux 发布版本所支持的最低版本列表:

OS	Tool chain	GLIBC
Linux x86/x64	gcc 4.9.4	GLIBC_2.17
Linux armv7	gcc-arm-8.3-2019.03-x86_64-arm-linux-gnueabi	GLIBC_2.28
Linux armv8	gcc-arm-8.3-2019.03-x86_64-aarch64-linux-gnu	GLIBC_2.27

2.3.1 使用批处理脚本运行 demo

在 "examples" 文件夹下, Foxit PDF Conversion SDK 提供了一个 pdf2office 和 office2pdf demo。

运行 pdf2office demo, 导航到 "\examples\simple_demo\pdf2office" 目录, 然后, Windows 平台运行 "RunDemo.bat", Linux 平台运行 "RunDemo.sh"。

运行 office2pdf demo，首先需要将 "res\office2pdf" 文件夹下的所有文件拷贝到 java.exe 所在的目录。然后，导航到 "\examples\simple_demo\office2pdf" 目录，Windows 平台运行 "RunDemo.bat"。

"\examples\simple_demo\input_files" 文件夹下包含了 demo 使用的输入文件。在 "\examples\simple_demo\output_files" 文件夹下将生成以 demo 名称命名的文件夹，输出文件 (Word, Excel, PowerPoint 或者 PDF) 将会在该文件夹下生成。

2.3.2 使用 Eclipse IDE 运行 demo

为了减少发布包的大小，福昕没有提供 demo 的 Eclipse 工程。因此，如果您需要在 Eclipse IDE 中运行 demo，请按照如下的步骤操作：

- 1) 在 Eclipse 中创建一个 Java 工程，然后将 SDK 库集成到工程中。
- 2) 将您需要运行的 demo 的 java 文件拷贝到工程中。例如，对于 pdf2office demo，将 "\examples\simple_demo\pdf2office" 目录下的 "pdf2office.java" 文件拷贝到工程的 "src" 文件夹下。
- 3) 打开 Java 文件，根据您的需要修改 *output_path* 和 *input_path* 路径。例如，如果您需要使用代码中默认测试文件，则需要使用绝对路径，该路径指向 "examples\simple_demo" 目录下的 "output_files" 和 "input_files" 文件夹。
- 4) 运行 demo。右击工程，选择 **Run As -> Java Application** 运行该工程。

2.4 初始化 Foxit PDF Conversion SDK

在调用任何 APIs 之前，应用程序必须使用 license 授权码初始化 Foxit PDF Conversion SDK 库。试用 license 文件在 "lib" 文件夹下。

```
// Load the library.  
// You can also use System.load("filename") instead. The filename argument must be an absolute path name.  
static {  
    String os = System.getProperty("os.name").toLowerCase();  
    String lib = "fpdfconversionsdk_java_";  
    if (os.startsWith("win")) {  
        lib += "win";  
    } else {  
        lib += "linux";  
    }  
    if (System.getProperty("sun.arch.data.model").equals("64")) {  
        lib += "64";  
    } else {  
        lib += "32";  
    }  
    System.loadLibrary(lib);  
}
```

```
// Initialize the library.  
int error_code = Library.initialize("sn", "key");  
if (error_code != e_ErrSuccess) {  
    return;  
}
```

备注：参数 "sn" 的值在 "**conversionsdk_sn.txt**" 中 ("SN=" 后面的字符串)，"key" 的值在 "**conversionsdk_key.txt**" 中 ("Sign=" 后面的字符串)。

技术支持

Foxit 支持

为了提供更具个性化的技术支持，您可以登陆您的 [Foxit 帐户](#)来提交问题报告，以便我们收集问题的详细信息。我们的技术支持团队在收到问题报告后，会第一时间来帮助您解决问题。

您也可以查看我们的[支持中心](#)，选择 Foxit PDF SDK 产品，里面有很多有用的文章可能有助于您解决问题。

联系电话

电话: 1-866-MYFOXIT 或者 1-866-693-6948