



# DEVELOPER GUIDE

# FOXIT PDF

# CONVERSION SDK

For Node.js

## TABLE OF CONTENTS

<b>1</b>	<b>Introduction to Foxit PDF Conversion SDK.....</b>	<b>1</b>
1.1	Why Choose Foxit PDF Conversion SDK.....	1
1.2	Foxit PDF Conversion SDK for Node.js API.....	1
1.3	Evaluation .....	2
1.4	License .....	2
1.5	About this guide.....	2
<b>2</b>	<b>Getting Started.....</b>	<b>3</b>
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
	<b>Support.....</b>	<b>6</b>

# 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 Node.js API

In this guide, we focus on the introduction of Foxit PDF Conversion SDK for Node.js API on Windows and Linux platforms. Foxit PDF Conversion SDK for Node.js API ships with simple-to-use APIs that can help Node.js 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 trial a 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 Node.js 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 about our SDK package. 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 8, 10 and 11 (32-bit and 64-bit) Windows Server 2012 or later (32-bit and 64-bit)	It only supports for Windows 8/10 classic style, but not for Store App or Universal App. Prerequisites for Node.js 10-20 version.
Linux	64-bit OS <ul style="list-style-type: none"><li>The minimum supported version of the GCC compiler is gcc5.4.</li><li>The minimum supported version of GLIBC is GLIBC_2.17.</li></ul>	Prerequisites for Node.js 10-20 version.

**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.
- For PDF2Office, it supports Windows and Linux platforms. For Office2PDF, it only supports Windows platform before version 3.0. Starting from version 3.0, Office2PDF supports Windows and Linux x86/64 platforms.

### 2.2 What is in the package

Package for Windows Node.js is named "foxitpdfconversionsdk\_3\_0\_win\_nodejs\_example.zip", and package for Linux (x64) Node.js is named "foxitpdfconversionsdk\_3\_0\_linux64\_nodejs\_example.zip". Since they have similar structures, this guide primarily uses "foxitpdfconversionsdk\_3\_0\_win\_nodejs\_example.zip" as an example.

Download the package for Windows Node.js and extract it to a new directory. The release package contains the following folders:

<b>doc:</b>	API references, developer guide
<b>examples:</b>	sample demo
<b>res:</b>	font-related configuration files

**Note:**

- In the "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. Starting from version 2.1, metrics data will be included in the dynamic library, no manual configuration is required.
- In the "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, you can follow the steps below:

- 1) Open a command prompt or a terminal window, navigate to the root directory "foxitpdfconversionsdk\_3\_0\_win\_nodejs\_example" for Windows, or "foxitpdfconversionsdk\_3\_0\_linux\_nodejs\_example" for Linux x64, install Conversion SDK nodejs module using the command below:

```
npm i @foxitsoftware/foxit-pdf-conversion-sdk-node
```

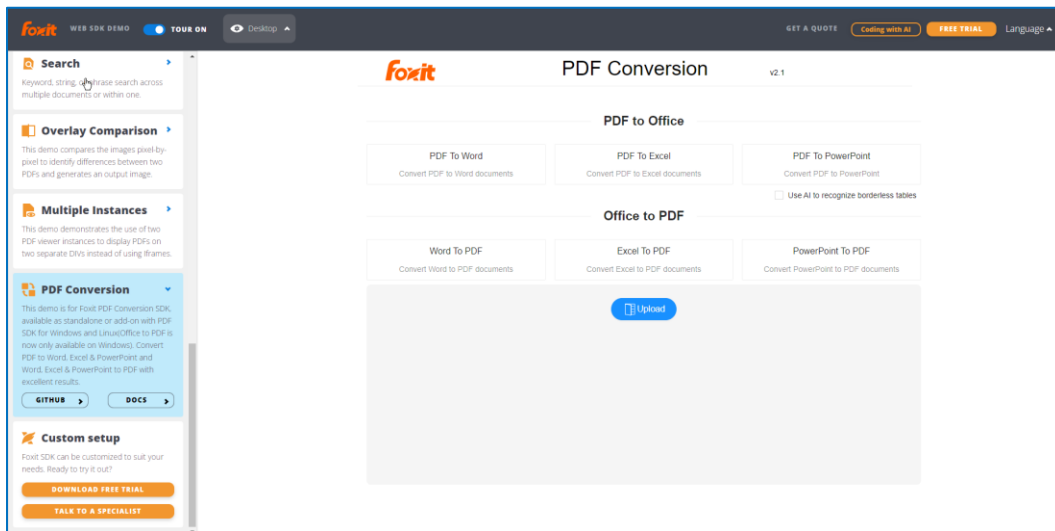
- 2) To run pdf2office demo, go to "examples\simple\_demo\pdf2office" directory, run "node pdf2office.js".

To run office2pdf demo, go to "examples\simple\_demo\office2pdf" directory, run "node office2pdf.js".

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:** Foxit-pdf-sdk-web-demo is an online demo application based on Foxit PDF SDK, which can help users experience the various features and functions of the Foxit PDF SDK. It has added a demo of **pdf2office** function by using Foxit PDF Conversion SDK for Node.js. You can access the foxit-pdf-

sdk-web-demo application at <https://github.com/foxitsoftware/foxit-pdf-sdk-web-demo.git>, and experience the **pdf2office** function at <https://webviewer-demo.foxit.com/conversion>.



## 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 root directory.

```
var sn = " ";
var key = " ";

var error_code = Library.Initialize(sn, key);
if (ErrorCode.e_ErrSuccess != error_code) {
  if (ErrorCode.e_ErrInvalidLicense == error_code)
    console.log("[Failed] Current used Foxit PDF Conversion SDK key information is invalid.");
  else
    console.log("Library Initialize Error: %d", error_code);
  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=").
- When initializing Foxit PDF Conversion SDK for Node.js library with the key, it is recommended to store the key on the backend Node.js service instead of exposing it in the frontend code. Storing the key in the frontend code can lead to security issues that attackers can easily access and misuse it.

## 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