

# **SENSOR MANAGER PLUG-IN SDK**

## **OVERVIEW GUIDE**

**April 2018**



© 2018 Adobe. All rights reserved.

*Sensor Manager Plugin SDK, Overview Guide .*

NOTICE: All information contained herein is the property of Adobe Systems Incorporated. No part of this publication (whether in hardcopy or electronic form) may be reproduced or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Adobe Systems Incorporated. Adobe, the Adobe logo are registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

This publication and the information herein is furnished AS IS, 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, makes no warranty of any kind (express, implied, or statutory) with respect to this publication, and expressly disclaims any and all warranties of merchantability, fitness for particular purposes, and noninfringement of third party rights.

# Contents

<b>Overview .....</b>	<b>4</b>
SDK directory structure .....	4
Supported platforms .....	5
Plug-in template project .....	5
Build PluginTemplate .....	5
Sample converter .....	6
Build the JsonConverter sample .....	6

# Overview

This Overview Guide introduces you to the Sensor Manager Plug-in SDK for Motion Graphics JSON (MGJSON). The Sensor Manager Plug-in SDK enables developers to write plug-ins for converting their file formats to MGJSON.

Adobe Motion Graphics JSON (MGJSON) is a structured data format that is specific to motion graphics. It is flexible enough to be able to represent data that represents motion. It differs from standard JSON in that it has a defined self-referential structure. The structure allows for the definition of streams of data that can represent almost any kind of data.

MGJSON is supported natively in Adobe Video Products - it can be imported just as you would any other type of footage and added to a Composition timeline. The data streams are then accessible and the values can be accessed and used to drive the values for other elements in the Composition.

**NOTE:** An understanding of JSON and JSON schemas is assumed and necessary in understanding how data is represented with MGJSON.

## SDK directory structure

The following directories are created when you unzip the *SensorManager-PluginSDK-CC201804* SDK file:

Directory	Purpose/Contents
/	It contains <i>SensorManager-Plugin-overview-guide.pdf</i> file.
public/	It contains docs, include, and source sub-directories.
public/docs/	It contains <i>Manifest_Schema.json</i> - schema for the manifest file, <i>MGJSON_Schema2.0.0.json</i> - schema for the MGJSON file, and <i>SensorManager-PluginSDKGuide</i> .
public/include/	It contains public header files to be included in the plug-in.
public/include/mgjsonWriter	It contains the header files for the MGJSON Writer Interfaces.
public/source/	It contains public source files to be included in the plug-in
public/source/mgjsonWriter	It contains client glue source files for the MGJSON Writer Interfaces.
third_party/ /boost /rapidjson	A placeholder for the third-party source files which are required by the Plug-in SDK. Refer to <i>ReadMe.txt</i> files within the respective folders for information on how to obtain and install the tools.  boost is required for all plug-ins. rapidjson is required for the sample <i>JsonConverter</i> .
PluginTemplate/	It contains the plug-in template. A new plug-in can be created using the <i>PluginTemplate</i> as a reference.
JsonConverter/	It contains the sample plug-in demonstrating the conversion of JSON to MGJSON file format.

## Supported platforms

Ensure that the machine on which you are running the Sensor Manager Plug-in SDK meets the following hardware and software requirements:

- ▶ Windows 7 and above (64 bit) using Visual C++ 2015 (Visual Studio Version 14)
- ▶ Mac OS X 10.12 using Xcode 8.2.1 for Intel processors

## Plug-in template project

The SDK includes ready-to-use template that you can use to create your own plug-ins. The `PluginTemplate` directory contains the following resources:

Directory	Purpose/Contents
<code>build/[vc14 xcode8]</code>	It contains projects for the respective platform.
<code>source/</code>	It contains the minimum required template files and the required headers files for the converter. The other required files are referenced from <code>public/source</code> and <code>public/include</code> directories.
<code>resource/txt</code>	It contains <code>MODULE_IDENTIFIER.txt</code> file. The plug-in's ID must be specified in the <code>MODULE_IDENTIFIER.txt</code> file. The other file - <code>SMPLUGINUIDS.txt</code> is the manifest file for the plug-in.

## Build PluginTemplate

Perform the following steps to build `PluginTemplate`:

1. Add the required files in the `third_party/boost` and other dependencies in `third_party`, if required.
2. Implement the required source files and modify the resources files.
3. Build the `PluginTemplate` project present at `PluginTemplate/build/[vc14|xcode8]/[PluginTemplate.sln|PluginTemplate.xcodeproj]` on `[win|mac]`.
4. The resulting `.mgx` plug-in is created at the following location:  
`public/libraries/[windows|macintosh]/[x64|intel_64_libcpp]/[Debug|Release]/PluginTemplate.mgx`

This plug-in can be used with Adobe After Effects 15.1.

## Sample converter

The SDK includes a sample `JsonConverter` that you can use to convert JSON file to MGJSON file. The `JSONConverter` directory contains the following resources:

Directory	Purpose/Contents
<code>build/[vc14 xcode8]</code>	It contains projects for the respective platform.
<code>include/</code>	It contains the header files used by the converter. The other required header files are referenced from the <code>public/include</code> folder.
<code>source/</code>	It contains the source files required by the converter. The other required source files are referenced from the <code>public/source</code> folder.
<code>resource/txt</code>	It contains <code>MODULE_IDENTIFIER.txt</code> file. The plug-in's ID must be specified in the <code>MODULE_IDENTIFIER.txt</code> file. The other file - <code>SMPLUGINUIDS.txt</code> is the manifest file for the plug-in.

### Build the JsonConverter sample

1. Add the source files in the `third_party/boost` and `third_party/rapidjson` directories.
2. Build the `JsonConverter` project present at the following location:  
`JsonConverter/build/[vc14|xcode8]/[JsonConverter.sln|JsonConverter.xcodeproj]` on `[win|mac]`.
3. The resulting `.mgx` plug-in is created at the following location:  
`public/libraries/[windows|macintosh]/[x64|intel_64_libcpp]/[Debug|Release]/JsonConverter.mgx`.