

# Intel<sup>®</sup> RealSense<sup>™</sup> SDK

## Getting Started Tutorial Using Unity\* Software

With the Intel<sup>®</sup> RealSense<sup>™</sup> SDK, you have access to robust, natural human-computer interaction (HCI) algorithms such as face tracking, finger tracking, gesture recognition, speech recognition and synthesis, fully textured 3D scanning and enhanced depth augmented reality.

Using the SDK and Unity\* software you can create Windows\* applications and games that offer innovative user experiences.

After performing the steps in this tutorial, you'll be ready to work through the [Capturing Raw Stream Tutorial](#) and start using the SDK in Unity software with the C# programming language.

# Contents

- **Quick Start**
- **The Intel® RealSense™ SDK Architecture**
- **Hardware and Software Requirements and Tools**
- **Install the SDK**
  - Install the SDK
  - Set up the Intel® RealSense™ Developer kit Camera
  - Installing Software Updates
- **Configure Unity Application Development Environment**
  - 1<sup>st</sup> Approach: Import the entire Unity Toolkit
  - 2<sup>nd</sup> Approach: Import only the Required DLLs
- **Run the Tutorial Code Samples**
- **To learn more**

# Quick Start

To get started quickly, follow these steps:

1	2	3
<a href="#">Download and install the SDK</a>	<a href="#">Configure Unity Application Development Environment</a>	Read the next tutorial, <a href="#">Unity Capturing Raw Streams</a> .

To learn more about the SDK, read the other sections in this document.

# The Intel® RealSense™ SDK Architecture

As you can see in Figure 1, applications that integrate the SDK sit on multi-layer software stack consisting of an SDK application layer, SDK Wrappers layer, SDK interfaces layer, SDK Core layer, I/O and capability modules. The SDK layer has direct access to the SDK interface when programming in the native SDK language (C++).

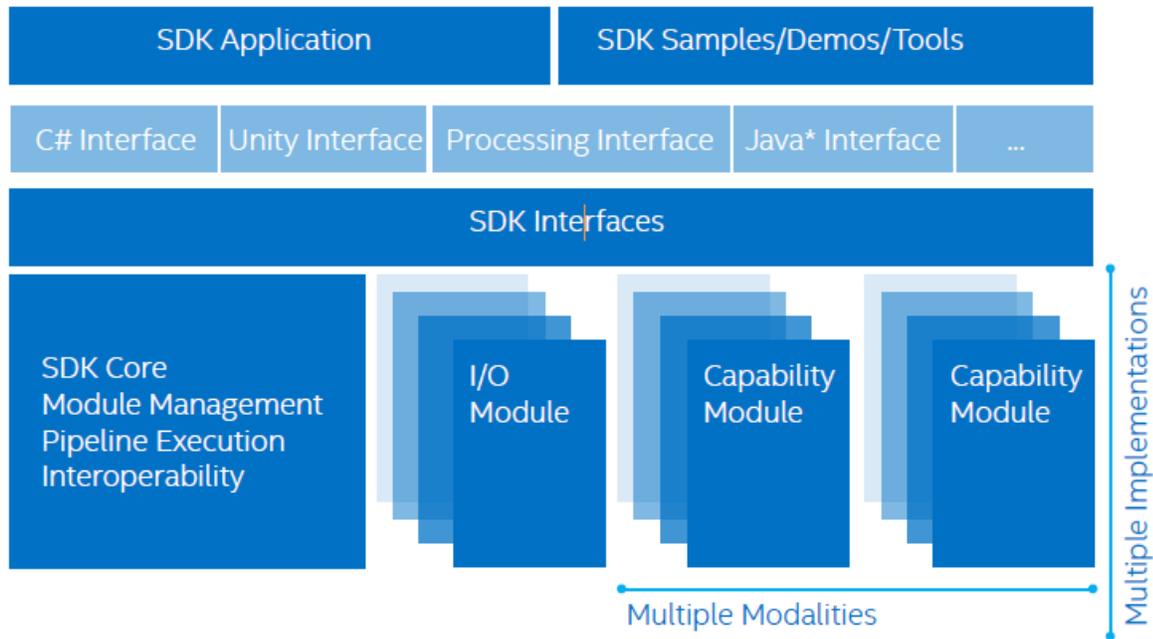


Figure 1. The Intel® RealSense™ SDK Architecture

The SDK wrapper layer exposes the SDK interface in a variety of languages i.e. C#, Processing, Java, Unity etc. This gives the user the flexibility to use any language of choice.

The SDK core is responsible for organizing the execution pipeline. It is also the base of the components that manages the two types of modules that provide SDK functionalities to your application.

- **I/O modules:** Capture the input data from your device and send that data to an output device or to the capability modules.
- **Capability modules:** Includes various RealSense algorithms such as pattern detection and recognition algorithms, like face tracking and recognition, finger tracking, gesture recognition, and voice recognition and synthesis.

It is possible to have multiple capability modules contained within the pipeline at the same time, so it is essential that the pipeline have an SDK manager. If you want to utilize more than one camera or other input device in your application, you may require multiple pipelines, each with its own manager.

## Hardware and Software Requirements and Tools

Required Hardware	4th generation (code name Haswell) Intel® Core™ processor 8GB free hard disk space Intel® RealSense™ 3D Camera
Required OS	Microsoft Windows* 8.1 OS 64-bit
Supported Languages	C++, JavaScript* C# (Microsoft .Net* 4.0 Framework is required) Java* (JDK 1.7.0_11 or later) Processing* 2.1.2 or later
IDE used for Samples and Tutorials	Microsoft Visual Studio* C++ 2010-2013 with service pack 1 or newer Unity* PRO 4. 1. 0 or later for Unity game development
Supported browsers for JavaScript development	Microsoft Internet Explorer* 10.0.13 Google Chrome* 33.0.1750.146 Mozilla Firefox* 27.0.1

The series of SDK tutorials only cover three (JavaScript, C++, C#) out of the five supported programming languages.

# Install the SDK

## Install the SDK

1. Download and run the SDK installer from <https://software.intel.com/realsense>
2. You will see a welcome screen as illustrated in Figure 2. Follow the instructions to complete the installation process.
  - By default, the SDK installs to the `C:/ProgramFiles(x86)/Intel/RSSDK` directory.
  - If the SDK installer detects any existing SDK versions, the SDK installer will prompt you for an upgrade. It is recommended to always do a clean uninstall and then install any newer SDK versions.

After installation, reboot the system when prompted. (This step is critical to propagate all environmental variables.)



Figure 2. Installer Welcome Screen

## Set up the Intel® RealSense™ Developer kit Camera

1. If the camera is integrated into the computer or laptop, skip to step 4.
  2. Install the camera on top of the computer or laptop lid.
  3. Plug the USB connection into one of the **USB 3.0** ports, as shown in Figure 3.
  4. Position yourself comfortably, with your back supported by your chair in a relaxed position, so your hands can move freely in front of the camera.
- To avoid fatigue, it is critical that users be in a relaxed position.

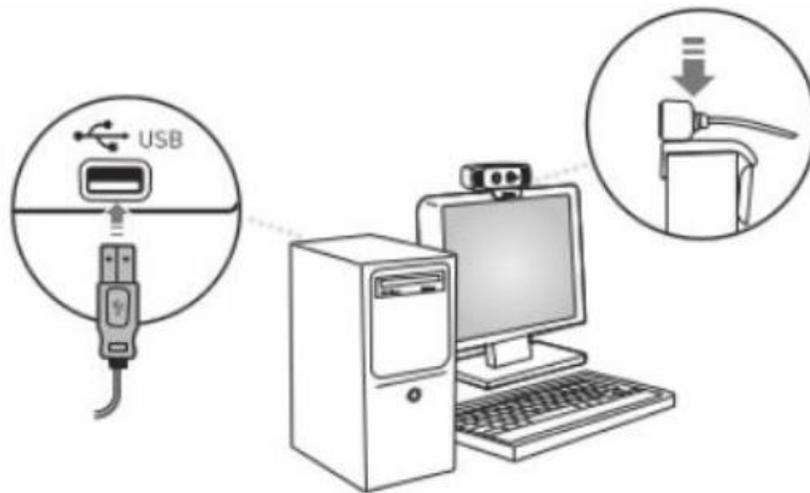


Figure 3. Camera Setup

## Installing Software Updates

All other software updates are at [intel.com/realsense/sdk](http://intel.com/realsense/sdk) - this includes:

1. **Language Packages** other than English if using other languages.
2. **Firmware Update Tool** – New firmware was pre-loaded on peripheral cameras distributed after Nov.17 2014. If your camera hardware stalls or has issues, check for new updates.
3. If you will use the peripheral camera microphone, download the **Audio Driver**.

Firmware and driver updates for systems with integrated 3D cameras will be provided by device manufacturers.

# Configure Unity Application Development Environment

The two approaches to incorporate the Intel RealSense SDK into your Unity application are:

## 1<sup>st</sup> Approach: Import the entire Unity Toolkit

You can import the Unity package, **RSUnityToolkit**, provided with the SDK and located at [C:\Program Files \(x86\)\Intel\RSSDK\framework\Unity](C:\Program Files (x86)\Intel\RSSDK\framework\Unity). The **RSUnityToolkit** is a set of scripts, prefabs, and other utilities along with a Unity Editor extension for using Unity's Editor user interface and Inspector to add Intel RealSense interactions with minimal code writing.

## 2<sup>nd</sup> Approach: Import only the Required DLLs

1. Create a new project or open an existing project in **Unity** software.
2. Create a **Plugins** directory under **Assets**.
3. Create a **Plugins.Managed** directory under **Assets**.
4. Copy the DLL files below from the source directory to the appropriate directory in your project.

File	Source Directory	Destination Directory
<b>libpxccpp2c.dll</b> <b>libpxccpp2c.dll.signature</b>	<a href="C:\Program Files (x86)\Intel\RSSDK\bin\win32">C:\Program Files (x86)\Intel\RSSDK\bin\win32</a> or <a href="C:\Program Files (x86)\Intel\RSSDK\bin\x64">C:\Program Files (x86)\Intel\RSSDK\bin\x64</a>	<b>Plugins</b>
<b>libpxcclr.unity.dll</b>	<a href="C:\Program Files (x86)\Intel\RSSDK\bin\win32">C:\Program Files (x86)\Intel\RSSDK\bin\win32</a> or <a href="C:\Program Files (x86)\Intel\RSSDK\bin\x64">C:\Program Files (x86)\Intel\RSSDK\bin\x64</a>	<b>Plugins.Managed</b>
<b>pxcmdefs.extensions.cs</b>	<a href="C:\Program Files (x86)\Intel\RSSDK\framework\common\pxcclr.cs\src">C:\Program Files (x86)\Intel\RSSDK\framework\common\pxcclr.cs\src</a>	<b>Plugins.Managed</b>

# Run the Tutorial Code Samples

We have created folders with prefabs, materials, and scripts with scenes for each of the following three Intel RealSense SDK tutorials: Capturing Raw Streams, Face Analysis, and Hand Analysis. To successfully use these tutorials and code samples, please make sure to copy these folders into your new project directory in case you want to create a project from scratch.

This step is not needed if you download and run the [Unity\\* tutorial code samples](#) provided online.

## To learn more

- For more information, read the [Architecture](#) and the [Programming Guide](#) sections in the [SDK Reference Manual](#).
- For detailed step-by-step information on using the Unity Toolkit provided in the Intel RealSense SDK, read the [SDK Unity Toolkit](#) section from the SDK Reference Manual.
- Read the [Configuring Application Development Environment](#) section for other languages such as Java, JavaScript, Processing, C#, C++.
- Make sure to read [Application Deployment Guidelines](#) section for information related to deployment of RSSDK applications.
- For situations where multiple SDK applications may run concurrently competing for the physical I/O devices please read the [Interoperating with Other Applications](#) section.