Mesh Generator 설명서

작성 : 2016.09.16 김미수

최종수정 : 2016.09.18 21:02 김미수

# **SDK 설치**

<https://software.intel.com/en-us/intel-realsense-sdk/download>

요구사항이 USB3.0, 하스웰, 윈8 이상이므로 주의.

# **API**

<https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_devguide_introduction.html> 참고

# **Importing the SDK Property Sheets**

API 문서에 C++를 위한 방법이 2가지가 있는데 아래의 방법 사용하였음. 하다보니 .MD, .MT 다 추가함.

For easy integration into the Microsoft Visual Studio development environment, use the property sheets located under the $(RSSDK\_DIR)/props directory. Table 1 lists available property sheets.

|  |  |
| --- | --- |
| **Property Sheet** | **Description** |
| VS2010-15.Integration.MD.props | Microsoft Visual Studio 2010-2015 property sheet for applications that compile with the dynamic runtime option. |
| VS2010-15.Integration.MT.props | Microsoft Visual Studio 2010-2015 property sheet for applications that compile with the static runtime option. |

Table 1: Visual Studio Integration Property Sheets

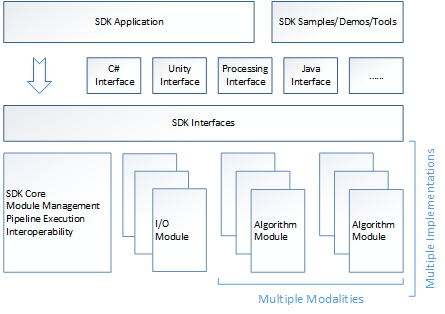
To import the property sheets, complete the following steps:

|  |  |
| --- | --- |
| • | Create a new project or open an existing project. |
| • | Open the property manager by View → Other Windows → Property Manager. |

|  |  |
| --- | --- |
| • | Right click on the project name and choose Add Existing Property Sheet. Add VS2010-15.Integration.MD.props or VS2010-15.Integration.MT.props for the application that requires dynamic or static runtime, respectively. |

원문 – <https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_devguide_property_sheets.html>

# SDK 구조

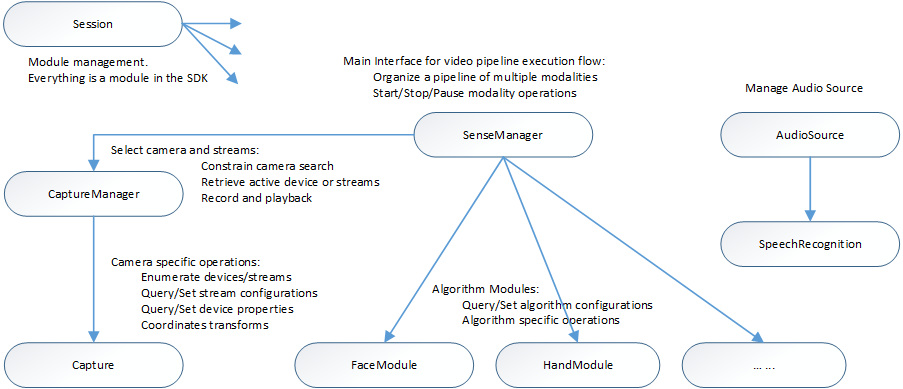


SDK 어플리케이션 계층을 C++로 구현하였음. 네이티브는 C++이고 그 위에 C#, 유니티, 자바 등의 인터페이스를 통해서도 SDK 인터페이스에 접근할 수 있는 듯. C++용 인터페이스는 앞에 PXC이 붙음.

원문 - <https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_essential_programming_guide.html>

원문 - <https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_essential_languages_and_frameworks.html>

## SDK 인터페이스



[Session](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcsession.html) 인터페이스가 모든 모듈을 제어하기 때문에 먼저 [Session](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcsession.html) 객체를 [CreateInstance](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/createinstance_pxcsession.html) 함수를 이용해서 생성해야 함. 다른 모듈은 [Session](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcsession.html) 객체를 통해서 생성.

이미 정의된 Hand tracking, Face tracking등의 인터페이스를 사용하기 위해서 [SenseManager](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcsensemanager.html) 인터페이스를 사용할 수 있으며 [CreateInstance](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/createinstance_pxcsensemanager.html) 함수를 이용해 생성함. 내부적으로 [SenseManager](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcsensemanager.html) 인터페이스는 [CaptureManager](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxccapturemanager.html) 인터페이스를 사용해 RGB, Depth, Audio 등의 스트림을 선택함. 캡쳐된 샘플은 [Image](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcimage.html) 인터페이스를 통해 접근 가능. 알고리즘 모듈에도 접근할 수 있는데 Hand Tracking에는 [HandModule](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxchandmodule.html), Face tracking에는 [FaceModule](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/pxcfacemodule.html)을 사용.

원문 - <https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_essential_sdk_interfaces.html>

# Language-Specific Restrictions for C#

Each C# instance is a C# managed object that internally points to a C++ unmanaged instance. Dispose the C# object so that the unmanaged C++ instance can be destroyed, with exception that internally managed instances (such as those returned by the [QueryInstance](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/queryinstance_pxcbase.html) function) do not have to be explicitly destroyed.

tipReleasing internally managed instances is ignored.

Example 3 shows how to cast a session instance to a meta data instance in C#:

Example 3: Cast a PXCMSession Instance to PXCMMetadata

|  |
| --- |
| PXCMSession session=PXCMSession.CreateInstance();  ...  PXCMMetadata mdata=session.QueryInstance<PXCMMetadata>();  ...  session.Dispose(); |

# 구현

Procedural Call을 이용하는 방법, Event Callback을 이용하는 방법이 있는데 전자로 구현.

3D 스캔 부분을 주로 참고하였음.

[~~https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc\_scan\_scanning\_process.html~~](https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_scan_scanning_process.html)

오브젝트 스캔은 평판 위에 있는 물체만 가능.

메쉬 제네레이팅은 30초 이상 걸림.

페이스 스캐닝, 오브젝트 스캐닝, 헤드 스캐닝, 바디 스캐닝이 가능. 각각 다른 모드.

원문 - <https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?sample_3d_scan_cs.html>

# 참고

## I/O Device Operations

<https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_essential_io_devices_enumeration_and_se.html>

## Calibration:

<https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?pxccalibration.html>

## Coordinate Systems

<https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/index.html?doc_essential_coordinate_systems.html>