# Intel® Media SDK for Linux*
# Getting Started Guide

## Overview

**Intel® Media SDK*** provides software development tools and libraries needed to develop enterprise grade media solutions on Intel® Xeon® and Core™ processor-based platforms.

This document covers installing the package components, which includes source code, libraries, user mode graphics stack components, and kernel module patches.

A set of simplified examples which can be used to validate the install can be found under Intel® Media Server Studio Product Page-> Support -> Code Samples -> Tutorials Package.

As multiple installation layouts are possible, we provide file paths relative to the folder where `MediaStack.tar.gz` package is unpacked.

## Installation Procedure

IMPORTANT NOTE: **The installation procedure for this release is different than previous installations.** This product is a combination of driver, library, and graphics stack components requiring specific hardware, Linux* distributions, and kernel versions.

Intel® Media SDK unlike Intel® Media Server Studio 2018 R2 supports only Generic installation.

Double check the processor on your system with "cat /proc/cpuinfo" before starting.

Intel® Media SDK supports the following platforms with the integrated graphics:

- Intel® Celeron® and Pentuim® processors codenamed "Apollo Lake I"
- Intel® Xeon® E3-1200 v4 Family with C226 chipset
- Intel® Xeon® E3-1200 and E3-1500 v5 Family with C236 chipset
- 5th Generation Intel® Core™
- 6th Generation Intel® Core™
- 7th Generation Intel® Core™
- 8th Generation Intel® Core™
- 9th Generation Intel® Core™

- Note: chipset must have processor graphics enabled. Make sure to check the datasheet for your system.

- Having a C226/C236 chipset is necessary but not sufficient. Make sure to consult with specific platform or board vendor regarding processor graphics being supported. Check Media Server Studio website for the list of "Known OEM/ODM Functional Platforms": https://software.intel.com/en-us/intel-media-server-studio/details

This article describes the processor and OS support matrix: https://software.intel.com/en-us/articles/driver-support-matrix-for-media-sdk-and-opencl

Please consult this article and the release notes for more background and details.

**This release is validated with Ubuntu 18.04** using 4.19.5 kernel from www.kernel.org.

## Prerequisite Steps

Add the user(s) who will run Intel® Media SDK applications to the video group

```
$ usermod -a -G video [LOGIN]
```

Check that an Intel VGA adapter can be found with `lspci`:

```
$ lspci -nn -s 0:02.0

00:02.0 VGA compatible controller [0300]: Intel Corporation Broadwell-U
Integrated Graphics [8086:193b] (rev 09)
```

The command output above shows 193b as the graphics device ID. The ID reported by lspci may be different for your machine. The main thing to look for is that an Intel graphics adapter is available. If not, you may need to check your BIOS settings and hardware configuration.

For Media SDK hardware access:

1. The chipset must support integrated graphics and the motherboard must be wired for display from the processor. This is generally always the case for Intel® Core™-based systems, but may need to be checked for systems using other product lines such as Intel® Xeon® processors. The best place to start is your hardware documentation.
2. Intel integrated graphics must be enabled in the BIOS. For many server machines there is also a small graphics adapter on the motherboard which can be configured separately. Usually integrated graphics should be enabled and the motherboard adapter disabled. Some experimentation may be required to find the best configuration for your system if your hardware documentation does not indicate the settings to use.
3. You must have a processor with Intel integrated graphics (see product release notes for specific processor details.)

If all of these criteria are met and lspci shows an Intel graphics adapter you are ready to proceed with Media SDK installation.

Make sure that proxies, network connections, firewalls, etc. are set up to allow yum and wget to download packages.

### "Generic" Steps

The commands below show the steps to install using the Generic approach in the Ubuntu 18.04 environment, assuming kernel 4.19.5 is already installed. These steps may need to be modified for other configurations.

```
#!/bin/bash

echo "install user mode components"
#unpack the generic package
tar -xvzf MediaStack.tar.gz
cd MediaStack
sudo ./install_media.sh

echo "Install finished, please "
echo "1. add user to video group: usermod -a -G video user"
echo "2. reboot"
```

**Note**: Make sure you boot the right kernel. You may need to edit grub.cfg manually to automatically load the 4.19.5 kernel. 6th Generation Intel® Core™ and 7th Generation Intel® Core™ require kernel parameter i915.alpha_support to be set in order to load i915 kernel mode driver.

## Verifying correct installation

The /opt/intel/mediasdk directory should be populated

```
$ ls /opt/intel/mediasdk/
bin doc  include  lib64  opensource  plugins share
```

The /dev/dri directory should have a renderD interface.

**Note:** vainfo and other libva tools are part of "libva-utils" package. You may download source code of package from https://github.com/01org/libva-utils.

The **vainfo** utility should show the current driver, Media SDK's iHD (from /opt/intel/mediasdk) and several codec entry points.

```
$ /opt/intel/mediasdk/bin/vainfo

libva info: VA-API version 1.4.0
libva info: va_getDriverName() returns 0
libva info: User requested driver 'iHD'
libva info: Trying to open /opt/intel/mediasdk/lib64/iHD_drv_video.so
libva info: Found init function __vaDriverInit_1_4
libva info: va_openDriver() returns 0
vainfo: VA-API version: 1.4 (libva 2.4.1)
vainfo: Driver version: Intel iHD driver - 19.1.git_42f6f231_2019-04-17
vainfo: Supported profile and entrypoints


      VAProfileNone                   : VAEntrypointVideoProc
      VAProfileNone                   : VAEntrypointStats
      VAProfileMPEG2Simple            : VAEntrypointVLD
      VAProfileMPEG2Simple            : VAEntrypointEncSlice
      VAProfileMPEG2Main              : VAEntrypointVLD
      VAProfileMPEG2Main              : VAEntrypointEncSlice
      VAProfileH264Main               : VAEntrypointVLD
      VAProfileH264Main               : VAEntrypointEncSlice
      VAProfileH264Main               : VAEntrypointFEI
      ...
```

If you see an error that i965_drv_video cannot be found, verify and update your environment settings:

```
$ export LIBVA_DRIVERS_PATH=/opt/intel/mediasdk/lib64
$ export LIBVA_DRIVER_NAME=iHD
```

## Samples and Tutorials

Media SDK package includes Samples with released samples binaries for testing after install. The package with latest samples binaries will not auto installed by installation script. Samples and tutorials are available at https://software.intel.com/en-us/intel-media-server-studio-support/code-samples

Samples are longer, more complex, and their main purpose is to show a broad range of the capabilities in Media SDK. Tutorials are shorter, clearer, and intended to be starting points for understanding how to develop code with the SDK.

Example code can be downloaded from https://software.intel.com/sites/default/files/managed/2b/60/mediasdk-tutorials-0.0.4.tar.gz

**Building Intel® Media Server Studio Tutorials:**

The tutorials are intended to be simple and clear starting points. They are built with standard makefiles. By default, executables can be found in the _build directory.

To get more info on tutorial parameters use

[executable name] --help

**Note:** example programs in the tutorials package are suitable for SDK API study only. It has limited functionality and were not intended for any kind of SDK and media stack validation.

**Building Intel® Media SDK Samples:**

The Intel® Media SDK samples are built with a recent version of CMake*. This can be downloaded from www.cmake.org or installed via standard package management.

To build, navigate to samples directory (by default /opt/intel/mediasdk/share/mfx/samples) then type

```
export PKG_CONFIG_PATH=/opt/intel/mediasdk/lib64/pkgconfig:$PKG_CONFIG_PATH
mkdir build && cd build
cmake ../
make
```

The build script will only build samples if the prerequisites can be found. For most cases only libdrm is needed. If X11 is not installed the X11 support in samples will be disabled.

The transcode sample is a great starting point to check your system:

```
sample_multi_transcode -i::h264 test_in.h264 -o::h264 test_out.h264 -hw
```

**To test OpenCL:**

Download the OpenCL Linux samples package from https://software.intel.com/en-us/intel-opencl-support/code-samples

The CapsBasic sample prints OpenCL query results for your system, and makes an excellent smoke test.

```
cd CapsBasic; make; ./CapsBasic
```

This should show a functioning GPU device.

The matrix multiply sample tests launching kernels:

```
cd GEMM; make;./GEMM –t gpu
```

More samples are available from https://software.intel.com/en-us/intel-opencl-support/code-samples. Use OpenCL 1.2 Samples for Linux.

# Legal Information

THIS DOCUMENT CONTAINS INFORMATION ON PRODUCTS IN THE DESIGN PHASE OF DEVELOPMENT.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this

document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

MPEG is an international standard for video compression/decompression promoted by ISO. Implementations of MPEG CODECs, or MPEG enabled platforms may require licenses from various entities, including Intel Corporation.

VP8 video codec is a high quality royalty free, open source codec deployed on millions of computers and devices worldwide. Implementations of VP8 CODECs, or VP8 enabled platforms may require licenses from various entities, including Intel Corporation.

Intel, the Intel logo, Intel Core, Iris, Intel HD Graphics, Intel Xeon are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others

© Intel Corporation.

**Optimization Notice**

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel.

Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Notice revision #20110804