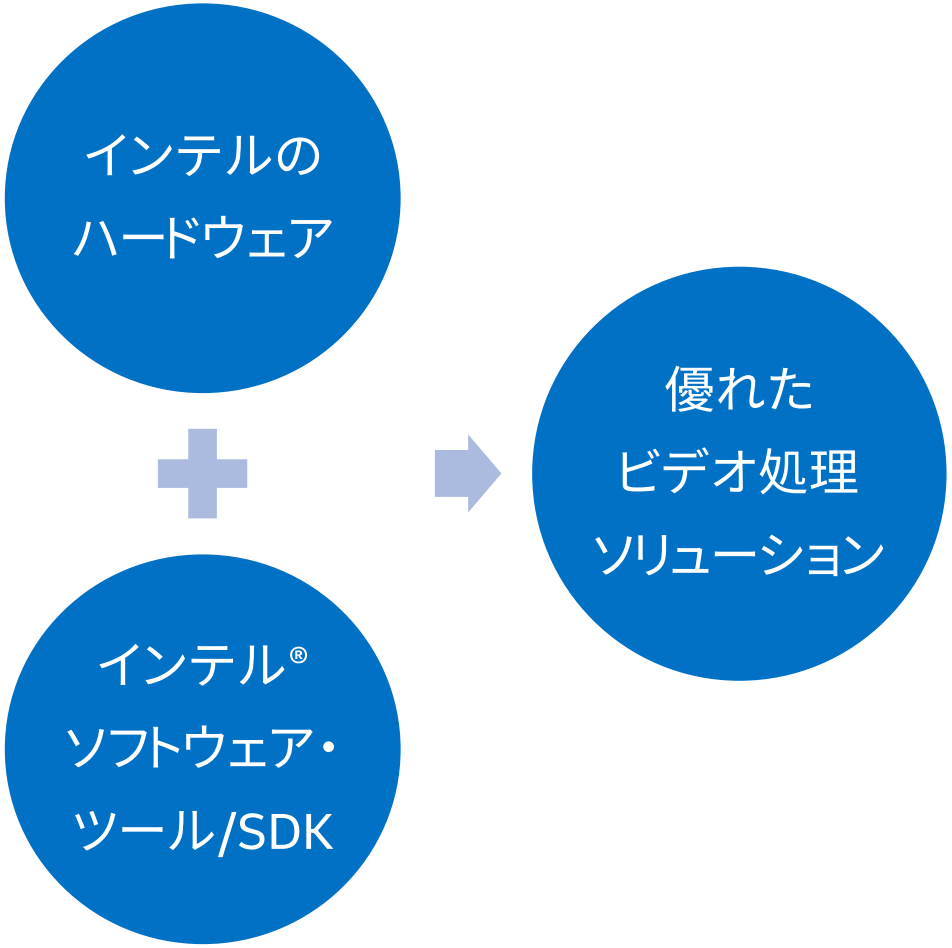
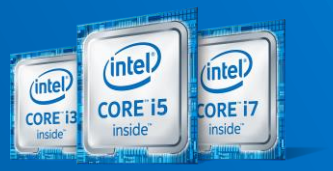
Linux* 向けインテル® Media Server Studio のご紹介

本日の内容



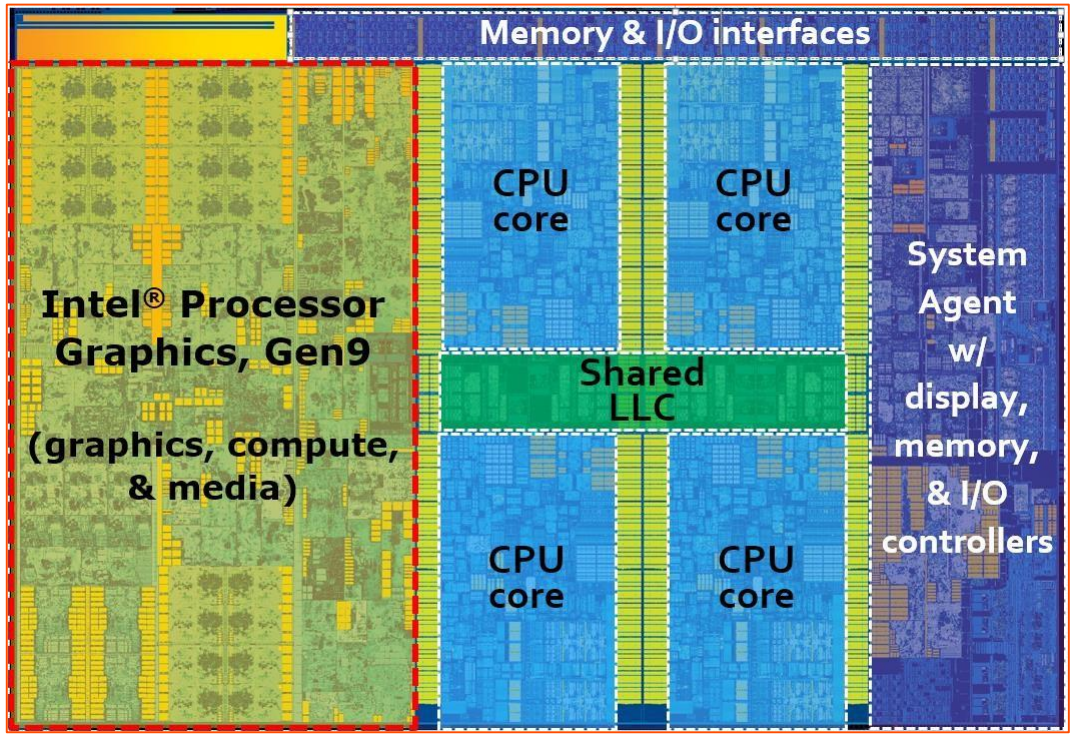
第 6 世代インテル® Xeon® プロセッサー/インテル® Core™ プロセッサーのヘテロジニアス・ハードウェアの概要

チップの別の部分



動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

インテルのハードウェアはヘテロジニアス



CPU

- 優れた汎用パフォーマンス
- 大規模なソフトウェア・エコシステム

プログラマブルなインテルのハードウェア

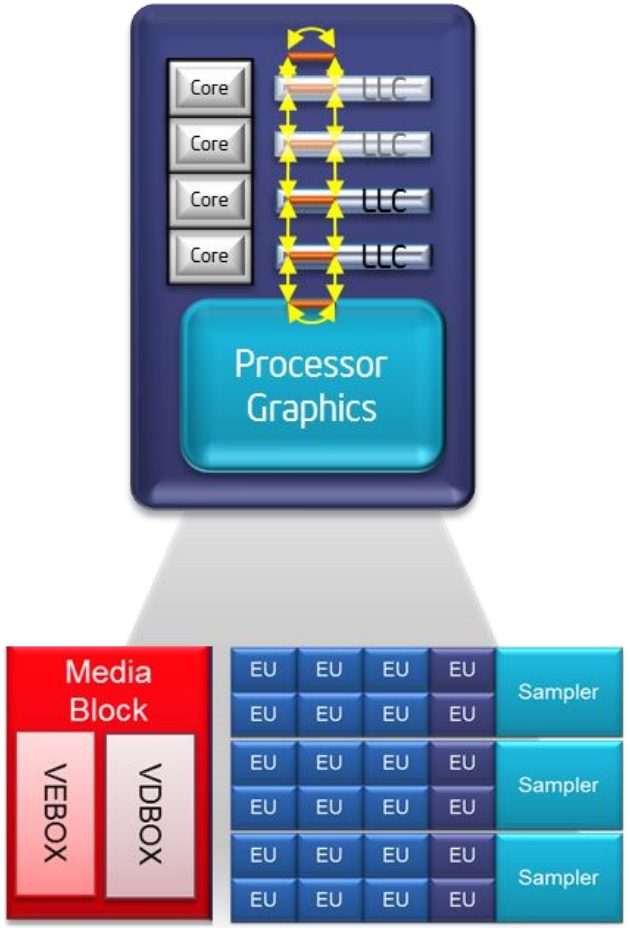
- GPU (ここで説明)
- IPU
- インテル® FPGA (旧 Altera® FPGA)

動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

メディア機能

第 9 世代インテル® プロセッサー・グラフィックス GPU

- 14nm プロセス・テクノロジー
- プロセッサーに統合
- より高いパフォーマンス
  - GT2 (24 実行ユニット)
  - GT4e (72 実行ユニット/128MB eDRAM)
  - CPU + GPU で 1TFLOPS を超える処理能力を提供
- 最新の API 機能をサポート
  - DirectX® 3D 2015、OGL 4.4、OpenGL* ES 3.0、OpenCL* 2.1
  - 共有仮想メモリー + OpenCL* を使用した密結合の CPU/GPU プログラミング
- メディア機能向けに拡張されたハードウェア・アクセラレーション
  - 省電力/フル固定機能 AVC エンコード
  - HEVC エンコード/デコード
  - MJPEG エンコード



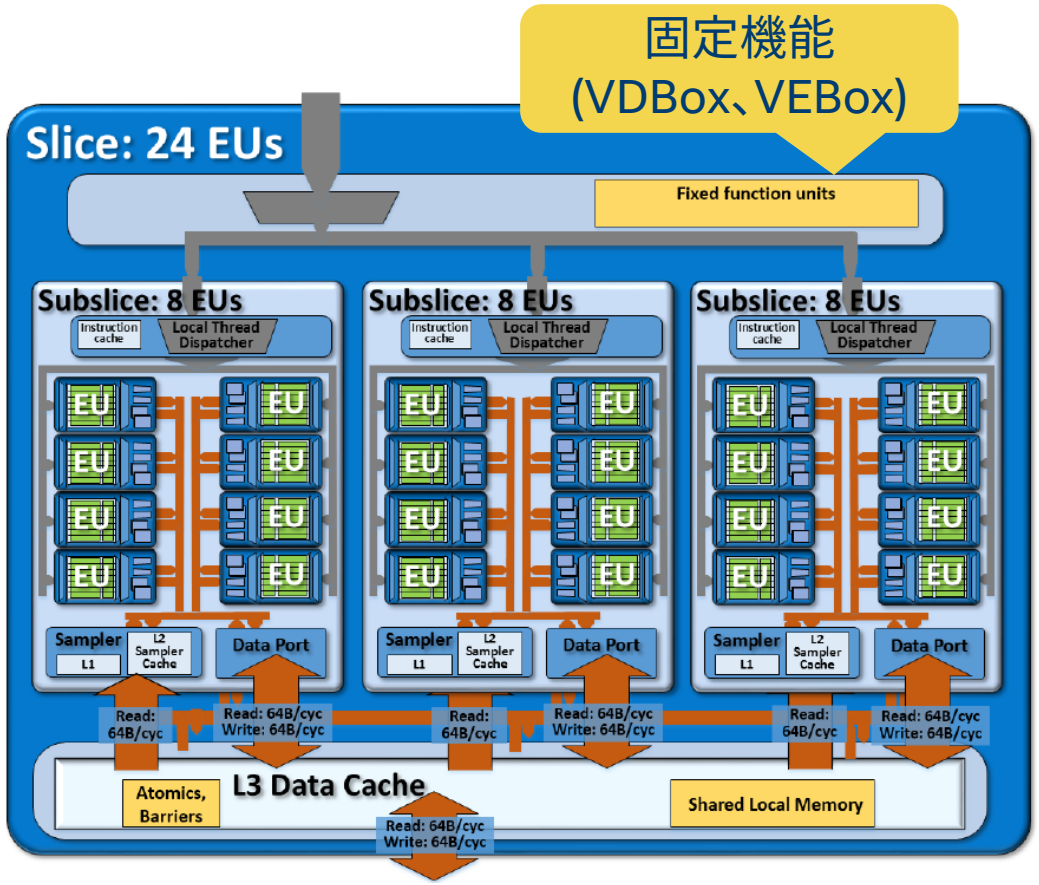
Processor Graphics Components

動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

主なグラフィックス・テクノロジー

用語集

- 実行ユニット (EU) = 汎用コア
- “スライス” には EU、サンプラー、キャッシュなどが含まれる
- 固定機能は “アンスライス” に含まれる
- eDRAM はキャッシュを追加し、帯域幅を増加

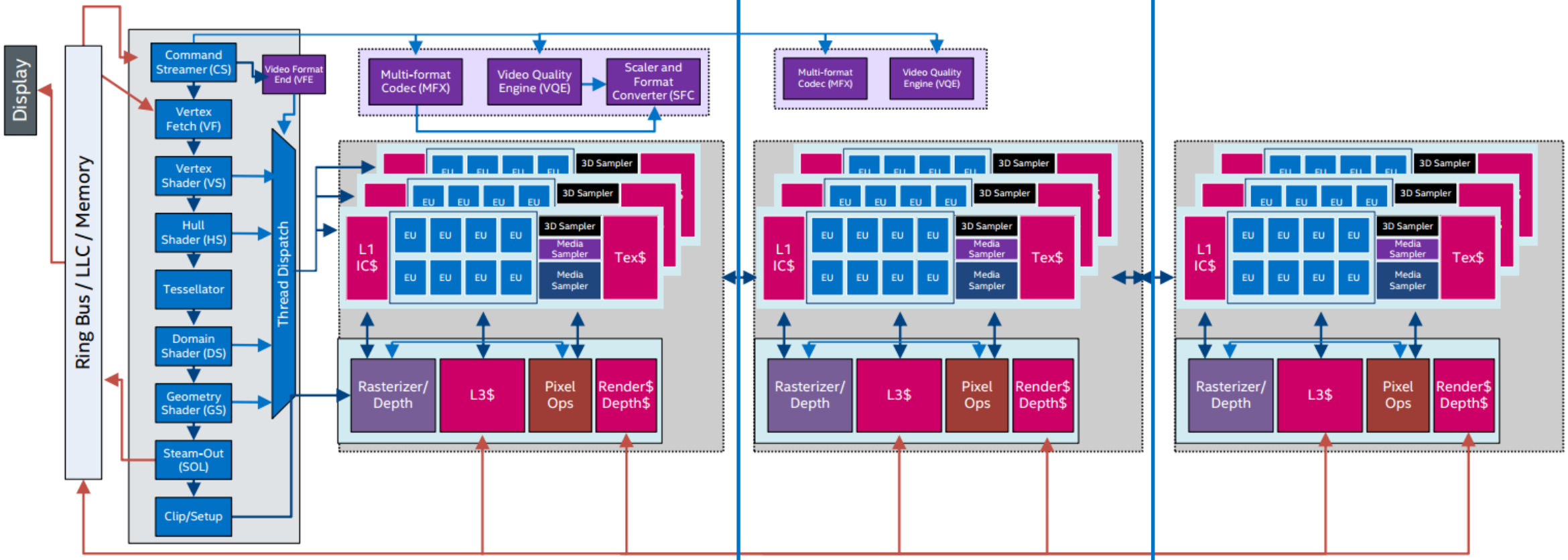


各プロセッサー・グラフィックス

	拡張	別名	要約
インテル® HD グラフィックス		GT2 “4+2”	良い
インテル® Iris™ グラフィックス	+ スライス + eDRAM	GT3 “2+3e”	さらに良い
インテル® Iris™ Pro グラフィックス	+ スライス + eDRAM	GT3e、GT4e “4+4e”	最良

intel.co.jp で “インテル® クイック・シンク・ビデオ” を検索してみてください

インテル® プロセッサー・グラフィックス/GPU の概要



GT2
インテル® HD グラフィックス
24 EU、1 MFX

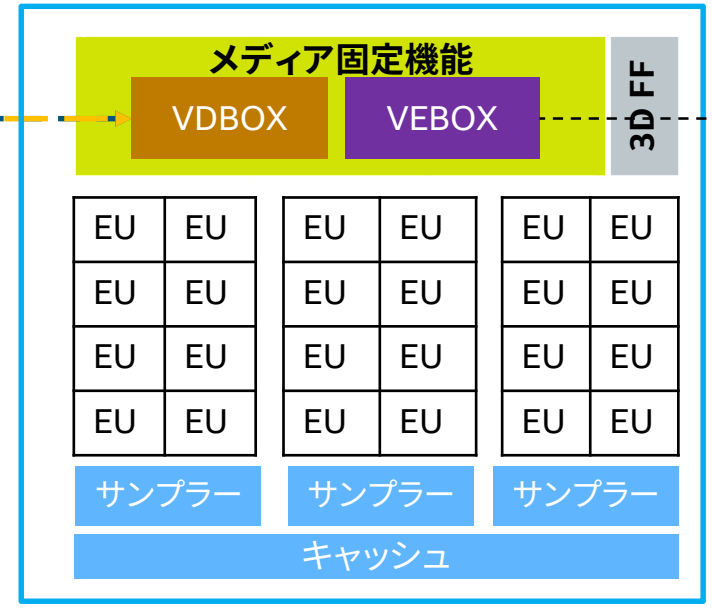
GT3
インテル® Iris™ グラフィックス
48 EU、2 MFX

GT4
インテル® Iris™ Pro グラフィックス
72 EU、2 MFX

コーデックとフレーム処理は固定機能と実行ユニットを使用

ビデオ・エンコーディング
ENC = EU + VDBox VME (MB タイプ、動きベクトル、ビット配分/BRC)
PAK = VDBox (残差パッキングとエントロピー・コーディング)
VDENC = 省電力エンコード (第 6 世代インテル® Core™ プロセッサー・ファミリー以降)

ビデオ・デコーディング
BSD = VDBox デコード



VPP

VPHal
ビデオ処理ハードウェア・
アクセラレーション・レイヤー

VEBox
- デインターレース
- ノイズ除去 (Luma/Chroma)
- フレームレート変換
- 色空間変換
- 合成/アルファ・ブレンディング
- スケーリング

インテル® Media Server Studio のインストール

Linux*

動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

インストール前の準備

- インテル® Media Server Studio を使用するユーザーを video グループに登録

$ usermod -a -G video [LOGIN]

- lspci コマンドによるグラフィックス・コントローラーの確認

$ lspci -nn -s 0:02.0

(出力例) 00:02.0 VGA compatible controller [0300]: Intel Corporation Sky Lake Integrated Graphics [8086:193b] (rev 09)

- インテル® Media Server Studio を使用したハードウェアへのアクセス:
  - インテル® インテグレーテッド・グラフィックスをサポートしているシステムであることを確認
  - 通常は、インテル® Core™ プロセッサーか一部のインテル® Xeon® プロセッサーに実装されているが、BIOS で Enable になっていることを確認

インストール前の準備 (CentOS* の場合)

- CentOS* をインストールする際には、カーネルのバージョンをチェック
  - インテル® Media Server Studio 2017 (R1/R2) は CentOS* 7.2.1511 kernel (3.10.0327.13.1.x86_64) のみを全機能検証
    - CentOS* でも、それ以外のバージョンについては “Generic install” を参照
  - CentOS* の最新版はカーネル 7.3 (http://www.centos.org (英語)) ← このバージョンはインストーラー・スクリプトでエラーになる
    - “yum -releaserver=7.2.1511” コマンドでチェック
    - 解決策: http://vault.centos.org/7.2.1511/isos/x86_64/ (英語) から ISO ファイルをダウンロードしてインストール
  - CentOS* のインストール時にはソフトウェアは “Development and Creative Workstation” を選択
  - ミラーリストに関するエラー: /etc/yum.repos.d/CentOS-Base.repo ファイルを下記のように修正

#mirrorlist=http://mirrorlist.centos.org/?release=¥$releasever&ar ch=¥$basearch&repo=os&infra=¥$infra

baseurl=http://mirror.centos.org/centos/¥$releasever/os/¥$basearch/

デフォルトは逆

サポートしている Linux* カーネルバージョンを必ずチェック!!

インストール・スクリプト (CentOS*)

- インテル® Media Server Studio のインストール

```
$ tar -xzf MediaServerStudio*.tar.gz
$ cd MediaServerStudio*
$ tar -xzf SDK2017*.tar.gz
$ cd SDK2017*/CentOS
$ tar -xzf install_scripts_*.tar.gz
$ su
# ./install_sdk_CentOS.sh
# reboot
```

- インテルの OpenCL* r4.0 ドライバーのインストール

```
$ sudo rpm -ivh intel-opencl-r4.0-BUILD_ID.x86_64.rpm
$ sudo rpm -ivh intel-opencl-devel-r4.0-BUILD_ID.x86_64.rpm
$ sudo rpm -ivh intel-opencl-cpu-r4.0-BUILD_ID.x86_64.rpm
```

インストール・スクリプト (CentOS* 以外)

- インストールに必要なソフトウェア
  - glibc >= 2.12
  - gcc >=4.7
- インストール・スクリプトの詳細例

https://software.intel.com/en-us/articles/mss-2017-generic-install (英語)

  - Linux* カーネル 4.4 + インテル® Media Server Studio 2017
  - Linux* カーネル 3.14.5 + インテル® Media Server Studio 2016

```
#!/usr/bin/bash
# install prerequisite packages yum -y -t groupinstall "Development Tools"
yum -y -t install kernel-headers kernel-devel bc wget bison ncursesdevel hmaccalc zlib-devel binutils-devel elfutils-libelf-devel rpmbuild redhat-rpm-config asciidoc hmaccalc perl-ExtUtils-Embed pesign xmlto audit-libs-devel binutils-devel elfutils-devel elfutils-libelfdevel newt-devel numactl-devel pciutils-devel python-devel zlib-devel mesa-dri-drivers openssl-devel
GENERIC_KERNEL_SRC=linux-4.4.tar.xz
GENERIC_KERNEL_WEB_PATH=http://www.kernel.org/pub/linux/kernel/v4.x
#install Media Server Studio packages
find . -name 'libdrm*' -exec rm {} ¥;
find . -name 'libkms*' -exec rm {} ¥;
find . -name 'libva*' -exec rm {} ¥;
/bin/cp -r etc/* /etc
/bin/cp -r opt/* /opt
/bin/cp -r lib/* /lib
/bin/cp -r usr/* /usr
#get generic kernel source
if [ ! -f ./$GENERIC_KERNEL_SRC ]; then
    wget $GENERIC_KERNEL_WEB_PATH/$GENERIC_KERNEL_SRC
fi
if [ ! -f ./$GENERIC_KERNEL_SRC ]; then
   echo -e "Failed to get $GENERIC_KERNEL_SRC, please try download it manually from $GENERIC_KERNEL_WEB_PATH, put it same folder as this script, then run this script again." 1>&2
   exit 1
fi
tar -xJf $GENERIC_KERNEL_SRC
 cp /opt/intel/mediasdk/opensource/patches/kmd/4.4/intel-kernelpatches.tar.bz2 .

tar -xjf intel-kernel-patches.tar.bz2
 cd linux-4.4
for i in ../intel-kernel-patches/*.patch; do patch -p1 < $i; done
make olddefconfig
make -j 8
make modules_install
make install
```

インストール・スクリプト (CentOS* 以外)

https://software.intel.com/en-us/articles/mss-2017-generic-install (英語)

- **Intel-opencl-r4.0 ドライバーのインストール**
  - **インテル® Media Server Studio 2017 R2 では intel-opencl-r4.0 ドライバーをインストール**

```
$ mkdir intel-opencl

$ tar -C intel-opencl -Jxf intel-opencl-r4.0-BUILD_ID.x86_64.tar.xz

$ tar -C intel-opencl -Jxf intel-opencl-devel-r4.0-BUILD_ID.x86_64.tar.xz

$ tar -C intel-opencl -Jxf intel-opencl-cpu-r4.0-BUILD_ID.x86_64.tar.xz

$ sudo cp -R intel-opencl/* /

$ sudo ldconfig
```

インストールの確認 (重要!)

- **インストール場所の確認**

  $ ls /opt/intel/mediasdk/

  builder doc include lib lib64 opensource plugins samples tools

- **/dev/dri renderD インターフェイスの確認**
- **vainfo コマンドによるコーデックの確認**

  $ vainfo | grep -v 'unknown'

出力例

```
libva info: VA-API version 0.99.0
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_0_32
libva info: va_openDriver() returns 0
vainfo: VA-API version: 0.99 (libva 1.67.0.pre1)
vainfo: Driver version: 16.5.1.59511-ubit
vainfo: Supported profile and entrypoints
Supported profile and entrypoints
    VAProfileH264ConstrainedBaseline:        VAEntrypointVLD
    VAProfileH264ConstrainedBaseline:        VAEntrypointEncSlice
    VAProfileH264Main          :    VAEntrypointVLD
    VAProfileH264Main          :    VAEntrypointEncSlice
    VAProfileH264High          :    VAEntrypointVLD
    VAProfileH264High          :    VAEntrypointEncSlice
```

インストールの確認 (重要その 2)

- ランタイム・ライブラリーやドライバーの動作を確認

インテル® Media Server Studio のパッケージに同梱

```
$ tar -xzf MediaSamples_Linux_16.5.1-59511.tar.gz

$ cd MediaSamples_Linux_bin

$ ./sample_multi_transcode -i::h264 ../content/test_stream.264 -o::h264 test_out.h264 -hw –la
```

Pre-build されたバイナリー

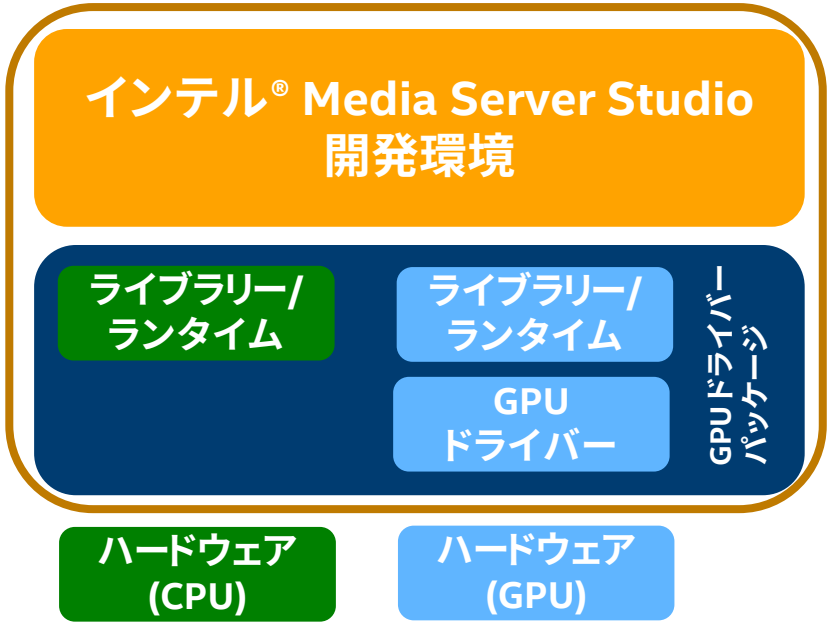
-hwオプションでは
第4世代インテル® Core™
プロセッサーのみ有効

ハードウェア (GPU) の
使用を指定

```
Multi Transcoding Sample Version 7.0.16053633
 libva info: VA-API version 0.99.0 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_0_32
libva info: va_openDriver() returns 0 Pipeline surfaces number: 120
MFX HARDWARE Session 0 API ver 1.21 parameters:
Input  video: AVC
Output video: AVC

Session 0 was NOT joined with other sessions

Transcoding started

Transcoding finished

Common transcoding time is  0.18 sec  MFX session 0 transcoding PASSED:
Processing time: 0.18 sec  Number of processed frames: 101

The test PASSED
```

出力例

インテル® Media Server Studio のアーキテクチャー

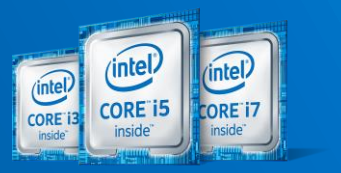


intel-opencl-<id>.el7.x86_64.rpm, intel-opencl-devel-<id>.el7.x86_64.rpm	インテル® Media Server Studio – OpenCL* ドライバー パッケージ
intel-linux-media-<id>.el7.x86_64.rpm	インテル® Media Server Studio –ドライバー & SDK ランタイムパッケージ
intel-linux-media-devel-<id>.el7.x86 _64.rpm	インテル® Media Server Studio –ドライバー & SDK 開発パッケージ
kmod-ukmd-<id>.el7.x86_64.rpm, ukmd-kmod-<id>.el7.src.rpm	ukmd カーネルモジュール
intel-i915-firmware-<id>.el7.x86_64.rpm	Skylake† サポート用ファームウェア・バイナリー
libdrm*-<id>.el7.x86_64.rpm, drm-utils*-<id>.el7.x86_64.rpm	ダイレクト・レンダリング・マネージャーのランタイム、ライブラリー、開発など
libva*-<id>.el7.x86_64.rpm, libva-utils*-<id>.el7.x86_64.rpm	ビデオ・アクセラレーション (VA) API ランタイム、開発など
install_scripts_centos_<id>.tar.gz	インテル® Media Server Studio –ドライバー & SDK インストール・スクリプト
MediaSamples_Linux_bin-<id>.tar.gz	インテル® Media Server Studio – サンプルパッケージ (バイナリーのみ)
Generic/intel-linux-media_generic_<id>_64bit.tar.gz, intel-opencl-<id>.tar.xz, intel-opencl-devel-<id>.tar.xz	インテル® Media Server Studio – 一般 OS 用ドライバー & SDK & OpenCL* パッケージ

† 開発コード名

最小システムのインストール

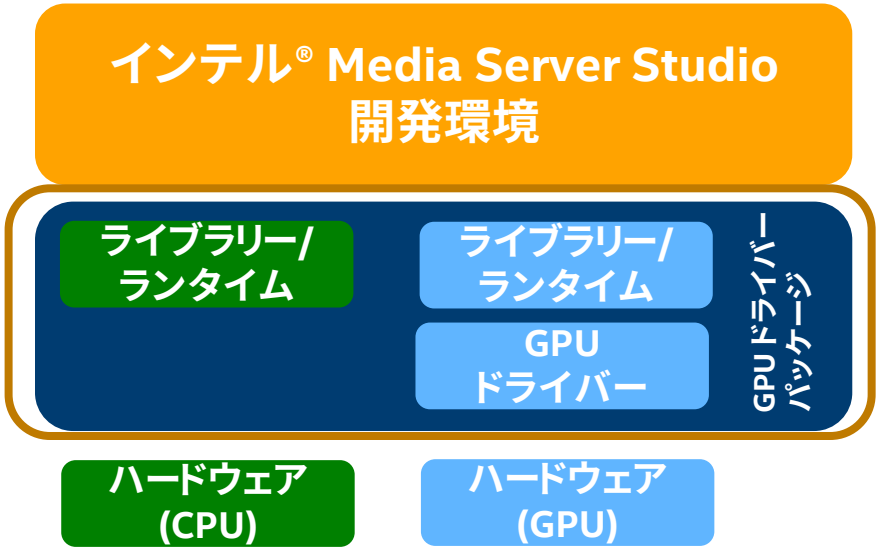
(実行環境のみ)



動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

実行環境のみのインストール (開発システム)

- 開発システムへのインストール
  - 前述の開発環境をすべてインストール
- libva を DRM 依存のみで再構築




```
#!/bin/bash
cd /opt/intel/mediasdk/opensource/libva/1.67.0.pre1-*/
tar -xjvf libva*
cd libva-*
./configure --enable-x11=no --enable-glx=no --enable-egl=no --enablewayland=no --disable-static --prefix=/usr --libdir=/usr/lib64/
make clean
make
cd va/.libs/
tar -czvf /tmp/libva_1.67.0.pre1.tar.gz libva*.so*
```

ビルドされた libva

実行環境のみのインストール (ターゲットシステム)

- ターゲットシステムに CentOS* 7.2 'minimal' ソフトウェア・パッケージをインストール
- 以下のファイルをターゲットシステムにコピー

intel-i915-firmware-16.5.1-*.el7.x86_64.rpm,

intel-linux-media-16.5.1-*.el7.x86_64.rpm,

libdrm-2.4.66-*.el7.x86_64.rpm,

intel-opencl-r4.0-*.x86_64.rpm,

intel-opencl-cpu-r4.0-*.x86_64.rpm,

kmod-ukmd-16.5.1-*.el7.x86_64.rpm (MediaServerStudio*/ SDK2017* ディレクトリー)

libva_1.67.0.pre1.tar.gz (/tmp ディレクトリー)

- ターゲットシステムでインストール

```
#!/usr/bin/bash
#install Media Server Studio packages
rpm -Uvh ¥
intel-i915-firmware-16.5.1-*.el7.x86_64.rpm ¥
intel-linux-media-16.5.1-*.el7.x86_64.rpm ¥
libdrm-2.4.66-*.el7.x86_64.rpm ¥
intel-opencl-r4.0-*.x86_64.rpm ¥
intel-opencl-cpu-r4.0-*.x86_64.rpm
tar -xvf libva_1.67.0.pre1.tar.gz -C /usr/lib64

#remove all other kmod-ukmd installation
for installed_kmod in `rpm -qa | grep kmod-ukmd`
do
echo "Removing previous version $installed_kmod...“
rpm -e $installed_kmod
done

#install kmod-ukmd
echo "Installing kmod-ukmd-$MILESTONE_VER-
$BUILD_ID..."
rpm -ivh kmod-ukmd-16.5.1-*.el7.x86_64.rpm

echo "Installation done."
```

インストール・スクリプトの例

ライブラリーの依存

インテル® Media Server Studio のライブラリーは 3 つに分けられ、ほかのライブラリーに依存

- **Runtime – メディア・トランスコーディング・アプリ実行中にランタイムで必要になるライブラリー**
- **Devel – アプリケーションや libva/libdrm のビルドの際に必要なライブラリー**
- **Utils – vainfo のようなツールに必要**

Runtime	内部依存*	外部依存**
libdrm	N/A	libpciaccess
libva	N/A	mesa-libGL libX11 libXext libXfixes mesa-dri-drivers
intel-linux-media	libdrm	N/A
kmod-ukmd	N/A	centos official kernels

Devel	内部依存*	外部依存 **
libdrm-devel	libdrm	kernel-headers
libva-devel	libva	N/A
intel-linux-mediadevel	intel-linux-media	N/A

チュートリアル・サンプルの実行

チュートリアル

https://softwre.intel.cm/en-us/intel-media-server-studio/code-samples#tutorials (英語)

- **インテル® Media Server Studio の基本機能を説明**
- **各セクションごとの Makefile でビルドが可能だが、API実装解説用なので、メディアスタックの検証には向かない**
- **ダウンロード・サイト https://software.intel.com/sites/default/files/mediasdk-tutorials-0.0.3.tar.gz (英語)**

Hello World	Hello World
Decode	Simple_2_decode
	Simple_2_decode_vmem
Encode	Simple_3_encode
	Simple_3_encode_vmem
	Simple_3_encode_vmem_async
VPP	Simple_4_vpp_resize_denoise
	Simple_4_cpp_resize_denoise_vmem
Transcode	Simple_5_transcode
	Simple_5_transcode_opaque
	Simple_5_transcode_opaque_async
	Simple_5_transcode_opaque_async_vppresize
	Simple_5_transcode_vmem
Advance	Simple_6_vpp_vmem_lowlatency*
	Simple_6_transcode_opaque_lowlatency*
	Simple_6_decode_vpp_postproc
	Simple_6_encode_vmem_vpp_preproc

* https://software.intel.com/en-us/articles/video-conferencing-features-of-intel-media-software-development-kit (英語)

サンプルプログラム

https://software.intel.com/en-us/intel-media-server-studio/code-samples#code-samples (英語)

- **インテル® Media Server Studio の最新の API における重要な機能を紹介**
- **サンプルプログラムの詳細:**

https://software.intel.com/sites/default/files/managed/83/95/Media_Samples_Guide_linux.pdf (英語) (サンプルガイド)
https://software.intel.com/sites/default/files/managed/1a/5a/MediaSamples_Linux_2017R2.tar.gz (英語) (ダウンロード・サイト)

サンプルプログラムのビルド:
$ yum install gcc g++ make cmake perl libX11-devel mesa-libGL-devel
$ export MFX_HOME=/opt/intel/mediasdk (デフォルト)
$ perl build.pl --cmake=intel64.make.release –build

サンプルプログラムの確認:
$ sample_multi_transcode -i::h264 test_in.h264 -o::h264 test_out.h264 -hw

"i965 cannot be found" エラーの場合:
$ cd /opt/intel/mediasdk/lib64
$ ln –s iHD_drv_video.so i965_drv_video.so

Transcoding sample
Encoding Sample
Decoding Sample
Video Processing sample
OpenCL Video Motion Estimation Sample

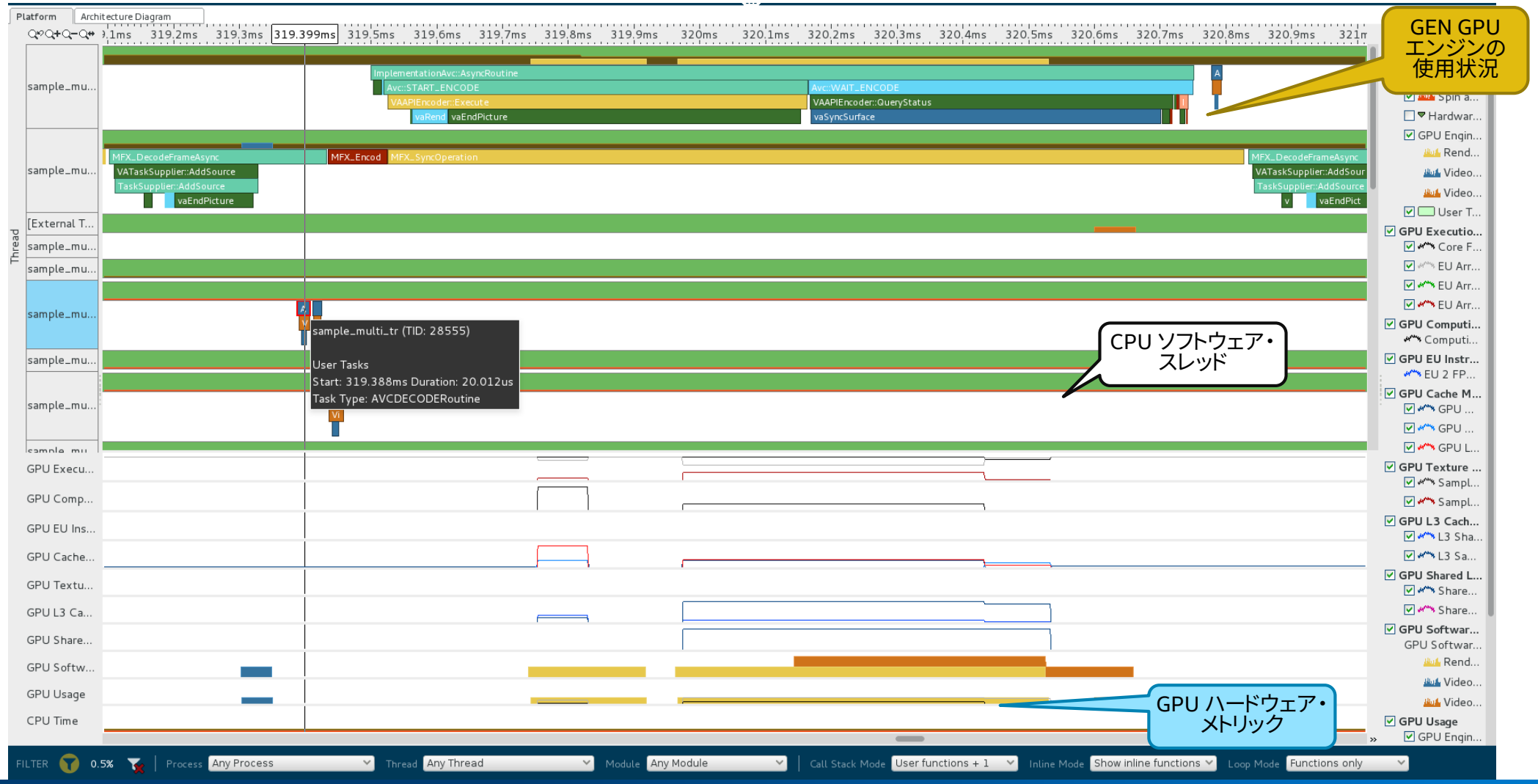
OpenCL* を使用した Transcoding のサンプルプログラム

Linux* における性能解析ツール

動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

インテル® VTune™ Amplifier for Linux*

インテル® VTune™ Amplifier** を使用してインテル® Media SDK と OpenCL* で最適化したメディア・アプリケーションを解析



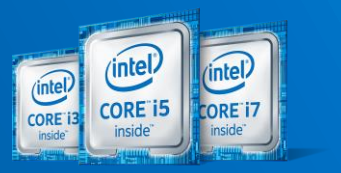
**インテル® Media Server Studio 内またはスタンドアロン・ツールとして利用可能。

FFmpeg* の解析例

(https://software.intel.com/en-us/articles/accessing-intel-media-server-studio-for-linux-codecs-with-ffmpeg (英語))

- FFmpeg* はバージョン 2.8 以降でインテル® クイック・シンク・ビデオをサポート
- https://www.ffmpeg.org/download.html (英語) からソースプログラムをダウンロード
- “--enable-libmfx --enable-nonfree” オプションで configure を実行
  - /opt/intel/mediasdk/include/mfx のコピー
  - libmfx.pc の作成
- “-vcodec h264_qsv” オプションで FFmpeg* を実行

デモ



動作環境は技術仕様を参照 - インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォームの SKU を選択

インテル® Media SDK の入手方法

インテル® Media Server Studio – 3 つのエディション (無料の Community Edition を含む)

プラットフォーム/デバイス

- インテル® Xeon® プロセッサーまたはインテル Core™ プロセッサー・ベースのプラットフォーム
- メディア、通信インフラストラクチャー、ビデオ処理/テレビ会議、デジタル監視、ビデオクラウド、データセンター向けアプリケーション
- HEVC、AVC、MPEG-2、MPEG-Audio をサポート

動作環境は技術仕様を参照

ダウンロード
isus.jp/intel-media-server-studio

インテル® Media SDK - 無料

プラットフォーム/デバイス

- インテル® Core™ プロセッサー・ファミリーまたはインテル® Core™ M プロセッサー・ファミリー
- インテル® クイック・シンク・ビデオをサポートするインテル® HD グラフィックス内蔵のインテル® Celeron® プロセッサー、インテル® Pentium® プロセッサー、インテル® Atom™ プロセッサーの SKU から選択
- クライアント・デバイス – デスクトップ/モバイル・アプリケーション

動作環境は技術仕様を参照

ダウンロード
isus.jp/intel-media-sdk

関連情報

インテル® Media SDK	• isus.jp/intel-media-sdk
インテル® Media Server Studio	• isus.jp/intel-media-server-studio
サンプルとチュートリアル	• github.com/Intel-Media-SDK/samples (英語)
フォーラム	• software.intel.com/forums/intel-media-sdk (英語)

Web セミナーのビデオ (英語)

Q & A

法務上の注意書きと最適化に関する注意事項

インテル® テクノロジーの機能と利点はシステム構成によって異なり、対応するハードウェアやソフトウェア、またはサービスの有効化が必要となる場合があります。実際の性能はシステム構成によって異なります。**絶対的なセキュリティーを提供できるコンピューター・システムはありません。**詳細については、各システムメーカーまたは販売店にお問い合わせいただくか、**http://www.intel.co.jp/** を参照してください。

テストでは、特定のシステムでの個々のテストにおけるコンポーネントの性能を文書化しています。ハードウェア、ソフトウェア、システム構成などの違いにより、実際の性能は掲載された性能テストや評価とは異なる場合があります。購入を検討される場合は、ほかの情報も参考にして、パフォーマンスを総合的に評価することをお勧めします。性能やベンチマーク結果について、さらに詳しい情報をお知りになりたい場合は http://www.intel.com/performance (英語) を参照してください。

ここに記載されているすべての情報は、予告なく変更されることがあります。インテルの最新の製品仕様およびロードマップをご希望の方は、インテルの担当者、営業所、または代理店までお問い合わせください。

性能に関するテストに使用されるソフトウェアとワークロードは、性能がインテル® マイクロプロセッサー用に最適化されていることがあります。SYSmark* や MobileMark* などの性能テストは、特定のコンピューター・システム、コンポーネント、ソフトウェア、操作、機能に基づいて行ったものです。結果はこれらの要因によって異なります。製品の購入を検討される場合は、他の製品と組み合わせた場合の本製品の性能など、ほかの情報や性能テストも参考にして、パフォーマンスを総合的に評価することをお勧めします。

本資料に記載されているコスト削減シナリオは、状況固有の多数の変動要因が加わることで、特定のインテル製品の購入が今後のコストとコスト削減にどのように影響するかについて理解を深めることができるようにするためのものです。本資料の内容は、一定レベルのコストを保証または確約するものではありません。

本資料に掲載されている情報は、インテル製品の概要説明を目的としたものです。本資料は、明示されているか否かにかかわらず、また禁反言によるとよらずにかかわらず、いかなる知的財産権のライセンスも許諾するものではありません。製品に付属の売買契約書『Intel's Terms and Conditions of Sale』に規定されている場合を除き、インテルはいかなる責任を負うものではなく、またインテル製品の販売や使用に関する明示または黙示の保証 (特定目的への適合性、商品適格性、あらゆる特許権、著作権、その他知的財産権の非侵害性への保証を含む) に関してもいかなる責任も負いません。

Intel、インテル、Intel ロゴ、Intel Inside、Intel Inside ロゴ、Intel Atom、Celeron、 Intel Core、 Iris、 Pentium、Xeon、VTune、Altera の名称およびロゴは、アメリカ合衆国および / またはその他の国における Intel Corporation の商標です。
Microsoft、DirectX、および Windows は、米国 Microsoft Corporation の、米国およびその他の国における登録商標または商標です。
OpenCL および OpenCL ロゴは、Apple Inc. の商標であり、Khronos の使用許諾を受けて使用しています。

最適化に関する注意事項

インテル® コンパイラーでは、インテル® マイクロプロセッサーに限定されない最適化に関して、他社製マイクロプロセッサー用に同等の最適化を行えないことがあります。これには、インテル® ストリーミング SIMD 拡張命令 2、インテル® ストリーミング SIMD 拡張命令 3、インテル® ストリーミング SIMD 拡張命令 3 補足命令などの最適化が該当します。インテルは、他社製マイクロプロセッサーに関して、いかなる最適化の利用、機能、または効果も保証いたしません。本製品のマイクロプロセッサー依存の最適化は、インテル® マイクロプロセッサーでの使用を前提としています。インテル® マイクロアーキテクチャーに限定されない最適化のなかにも、インテル® マイクロプロセッサー用のものがあります。この注意事項で言及した命令セットの詳細については、該当する製品のユーザー・リファレンス・ガイドを参照してください。
注意事項の改訂 #20110804

インテル® Media Server Studio のエディション – 概要

プレミアム・コンポーネント	Community Edition	Essentials Edition	Professional Edition
インテル® Media SDK	✓	✓	✓
グラフィックス・ドライバー	✓	✓	✓
コードサンプル	✓	✓	✓
OpenCL* Code Builder およびランタイム	✓	✓	✓
メトリックモニター (Linux* のみ)	✓	✓	✓
Online Service Center		✓	✓
HEVC デコーダー/エンコーダー、GPU アシスト API			✓
オーディオデコーダー/エンコーダー			✓
Video Quality Caliper			✓
インテル® VTune™ Amplifier			✓
高品質の逆テレシネとデインターレース			✓

intel Software

おかげさまで 30 周年

XLSOFT

エクセルソフト株式会社