Intel® Media Software Development Kit 2012 R2 Release Notes

(Version 3.5.811.41085)

Overview

The **Intel® Media Software Development Kit** (Intel® Media SDK) is a software development library that exposes the media acceleration capabilities of Intel® platforms for decoding, encoding and video preprocessing. The API library covers a wide range of Intel platforms. The Intel Media SDK targets general application developers who want to integrate encoding and decoding into their applications.

Please see the "`<install-folder>\doc\MediaSDK Sample Guide.pdf`" for information on sample source code provided with the Intel Media SDK.

New Features

The Intel® Media SDK 2012 R2 introduces API version 1.4. This version is backwards compatible with API version 1.3. API version 1.4 introduces specific Multi-view Video Coding (MVC) output mode which facilitates producing MVC streams compliant with: System Description Blu-ray Disc Read-Only Format, Part 3, Audio Visual Basic Specifications, Version 2.41, August 2010 and AVCHD Format, Version 2.0, Book 1: Playback System Basic Specifications .

This release also includes the following changes to API version 1.3 released in Intel® Media SDK 2012:

- Three new H.264 profiles and 6 (all defined in standard) constrained flags were defined. This change should simplify Media SDK usage by providing direct access to constrained flag configuration.
  `MFX_PROFILE_AVC_CONSTRAINT_SET0/1/2/3/4/5`
  `MFX_PROFILE_AVC_CONSTRAINED_BASELINE`
  `MFX_PROFILE_AVC_CONSTRAINED_HIGH`
  `MFX_PROFILE_AVC_PROGRESSIVE_HIGH`

- View output mode was added for MVC encoder `mfxExtCodingOption::ViewOutput`. This flag instructs encoder to output each view in separate bitstream buffer and format them according to Blu-ray* and AVCHD* Format requirements.
- MVC encode now supports interlace coding mode.

Please see the Intel Media SDK Reference Manual for details "`<install-folder>\doc\mediasdk-man.pdf`"

For information on the USER class please see "`<install-folder>\doc\mediasdkusr-man.pdf`"

For information on Multi-view Video Coding support please see "`<install-folder>\doc\mediasdkmvc-man.pdf`"

For information on JPEG/Motion JPEG support please see "`<install-folder>\doc\mediasdkjpeg-man.pdf`"

Additionally, the Intel Media SDK package now contains a standalone library which exposes an Application Programming Interface (API) for Stereoscopic 3D (S3D) rendering with Intel® HD Graphics 3000/2000. Please see Package Contents section for locations of header `igfx_s3dcontrol.h` and library `igfx_s3dcontrol.lib` files.

This library can be used either independently or along with any version of the Intel Media SDK library to utilize, for instance, decoding functionality.

System Requirements

Hardware

- IA-32 or Intel® 64 architecture processors with the Intel® Core™ processor or later is required for this Developer's release.
- 200 MB free hard disk space for this release.
- The software implementation DLLs, `libmfxsw32.dll` and `libmfxsw64.dll`, requires compatible IA-32 or Intel® 64 architecture processor with support for Intel® Streaming SIMD Extensions 2 instructions.

Software

- Microsoft* Windows Vista* with Service Pack 2, or Microsoft Windows* 7 Operating System
- Microsoft Visual C++* 2005 with Service Pack 1, or later version of Microsoft Visual C++
- For the Microsoft DirectShow* samples: Microsoft Windows SDK Update 6.1 for Windows Vista, or later
- For the Microsoft Media Foundation* samples: Microsoft Windows SDK for Windows 7

- For the Intel® OpenCL™ User Plug-in sample: Intel® OpenCL™ SDK 1.5 or later

Package Contents

Note: The suffix `<arch>` indicates 32- or 64-bit Microsoft* Windows* (either "`win32`" or "`x64`"). Both are always installed to allow for cross-platform development.

`<install-folder>`	Intel® Media SDK Release Notes (this file), End User License Agreement (EULA) "`Intel Media SDK EULA.rtf`" and license file "`license.txt`"
`<install-folder>\bin\<arch>`	Intel® Media SDK Dynamic Library, software implementation: `libmfxsw32.dll` for IA-32 architecture `libmfxsw64.dll` for Intel® 64 architecture
`<install-folder>\doc`	Intel® Media SDK documentation: • Intel® Media SDK Reference Manual `mediasdk-man.pdf` • Intel® Media SDK Extensions for User-Defined Functions `mediasdkusr-man.pdf` • Intel® Media SDK Extensions for Multi-view Video Coding `mediasdkmvc-man.pdf` • Intel® Media SDK Extensions for JPEG/Motion JPEG decoding `mediasdkjpeg-man.pdf` • Microsoft DirectShow* Filter Interfaces and Specifications `MediaSDK Filters Specifications.pdf` • Samples Overview `MediaSDK Sample Guide.pdf` • Intel® Media Developer's Guide `Intel_Media_Developers_Guide.pdf` • Intel® Media SDK Library Distribution and Dispatching Process `mediasdk-distrib.pdf`
`<install-folder>\include`	External Intel® Media SDK headers: • Type definitions in `mfxdefs.h` • Structure definitions in `mfxstructures.h`

	• Function definitions in C in `mfxvideo.h` • C++ wrapper of the SDK functions in `mfxvideo++.h` • Extensions for Multi-view Video Coding options `mfxmvc.h` • Extensions for User-Defined Functions `mfxplugin.h` • C++ wrapper for User-Defined Functions `mfxplugin++.h` • Extensions for JPEG*/Motion JPEG Video Decoding options `mfxjpeg.h`
`<install-folder>\lib\<arch>`	• Static Dispatcher Library `libmfx.lib` • Static Dispatcher Library `libmfxmd.lib` build with /MD compiler option to compatibility with applications using the multithread- and DLL-specific version of the run-time library
`<install-folder>\igfx_s3dcontrol\include`	• S3D API definitions `igfx_s3dcontrol.h`
`<install-folder>\igfx_s3dcontrol\lib\<arch>`	• Static S3D Control Library `igfx_s3dcontrol.lib`
`<install-folder>\samples\_bin\<arch>`	Contains the following source code samples: • Intel® Media SDK Encoding Sample in folder `sample_encode` • Intel® Media SDK Decoding and S3D Rendering Sample in folder `sample_decode`. A script for building a video wall application using this sample is provided. • Intel® Media SDK Transcoding Sample in folder `sample_multi_transcode` • Intel® Media SDK Video Processing Sample in folder `sample_vpp` • Intel® Media SDK Rotation Plug-in Sample in folder `sample_user_modules\rotate_cpu` • Intel® Media SDK OpenCL™ Plug-in Sample in folder `sample_user_modules\rotate_opencl` • Intel® Media SDK VPPPlugin Utility Class in folder `sample_utilities\vpp_plugin` • Intel® Media SDK Application Sample using Microsoft DirectShow in folder `sample_dshow_player` • Intel® Media SDK Plug-Ins Sample using Microsoft

	DirectShow in folder `sample_dshow_plugins` • Intel® Media SDK Application Sample using Microsoft Multimedia Framework Plug-ins in folder `sample_studio` • Intel® Media SDK Video Conferencing Sample in folder `sample_videoconf` For use only with Microsoft Windows 7 Operating System: • Intel® Media SDK Plug-Ins Sample using Microsoft Media Foundation* in folder `sample_mfoundation_plugins`

`<install-folder>\samples\`	Pre-built binaries of installed sample applications
	• If installed, console sample application binaries:
	`sample_encode.exe`
	`sample_decode.exe (with sample_video_wall.bat)`
	`sample_vpp.exe`
	`sample_multi_transcode.exe`
	`sample_videoconf.exe`
	• If installed, user plug-in sample binaries:
	`sample_rotate_plugin.dll`
	`sample_plugin_opencl.dll (with ocl_rotate.cl)`
	• If installed, Microsoft DirectShow sample binary application
	`sample_dshow.exe`
	• If installed, Microsoft DirectShow filters
	`h264_dec_filter.dll`
	`h264_enc_filter.dll`
	`mpeg2_dec_filter.dll`
	`mpeg2_enc_filter.dll`
	`vc1_dec_filter.dll`
	`mvc_dec_filter.dll`
	`jpeg_dec_filter.dll`
	`imc_*.dll`
	• If installed, Microsoft Multimedia Framework sample binary application
	`sample_studio.exe`
	`sample_studio_builder.exe`
	`sample_studio_profiles.txt`

`<install-folder>\`	Contains the following tools in binary form:

tools\	• Intel® Media SDK Tracer in folder mediasdk_tracer. This utility performs runtime recording of Intel Media SDK API calls and parameters to a log file. • Intel® Media SDK System Analyzer in folder mediasdk_sys_analyzer. This utility analyzes the system and reports back Intel Media SDK related capabilities, graphics driver and components status.

Installation

1. Installation requires full administrative rights.
2. Extract files from the .ZIP file to the target hard drive.
3. Select the installer that matches the target system architecture:

File Name for Installer	Target System Architecture
Intel(R)_Media_SDK_win32.msi	IA-32 architecture running Microsoft* Windows Vista* or Windows* 7 Operating System
Intel(R)_Media_SDK_x64.msi	Intel® 64 architecture running Microsoft Windows Vista or Windows 7 64-bit Operating System

Note: If win32 is installed, the x64 lib and bin files are also present to enable cross-platform development and vice versa.

Known Limitations

The Intel® Media SDK library has the following known limitations:

- The Intel Media SDK dispatcher libmfx.lib is best used with a standard DLL entry point (as recommended by Microsoft*) when used in a DLL application such as a Microsoft DirectShow* filter. The DLL entry point setting can be found under the Link > Advanced compiler options. Non-standard entry points can be used, but are not recommended.
- Loading of library libmfxsw32.dll/libmfxsw64.dll not through the dispatcher is unsafe.
- Using the Intel Media SDK in parallel with Intel® Threading Building Blocks could impact performance.
- Frames for different views in single AU in MVC encoder must be provided to encoder in order specified by mfxMVCViewDependency.

- MFX_EXTBUFF_AVC_REFLIST_CTRL and MFX_EXTBUFF_CODING_OPTION_SPSPPS external buffers are not supported by MVC encoder.
- MVC encoder supports MFX_PROFILE_AVC_STEREO_HIGH only.
- H.264 encoder doesn't support processing of mfxExtPictureTimingSEI template. During initialization 0xFFFF values will be reset to default values. In runtime 0xFFFF values will be put to bitstream as is.
- Known limitations for H.264 Multiple-Segment Encoding:
    1. Hardcoded HRD parameters: bit_rate_scale = 0, cpb_size_scale = 3
    2. Encoded bit_rate_value_minus1, bit_rate_scale represent BitRate from original SPS within precision of kbps (maximum supported BitRate is 2^16 – 1 kbps).
    3. Encoded cpb_size_value_minus1, cpb_size_scale represent CpbSize from original SPS within precision of Kb (maximum supported CpbSize is 2^16 – 1 Kb).
    4. Encoded time_scale, num_units_in_tick could be both multiplied by 2 if the time_scale from original SPS is odd.
    5. Conflicts between SPS/PPS and mfxVideoParam for parameters that are not covered by SPS/PPS could lead to change of parameters in SPS/PPS.
- RefPicMarkRepSEI syntax is not supported by MVC encoder.
- H.264 encoder is known to be slower on x64 platform than on win32 platform.
- If the MPEG-2 Video encoder mfxVideoParam::mfxInfoMFX::CodecProfile is initialized to 0, then the stream will be encoded as MFX_PROFILE_MPEG2_MAIN. Additionally if the MPEG-2 Video encoder mfxVideoParam::mfxInfoMFX::CodecLevel is initialized to 0, then the stream will be encoded as MFX_LEVEL_MPEG2_MAIN.
- MFX_FRCALGM_DISTRIBUTED_TIMESTAMP is unsupported by InverseTelecine and Deinterlace (60i->60p) VPP filters.
- Target usages 4 and 5 of H.264/MVC encoders are known to be non-monotonic in terms of encoding FPS versus encoding quality.
- Parameter BRCParamMultiplier is not supported by MVC video encoder.
- RGB32 as output color format for video pre-processing is allowed only if resize filter is enabled.
- H.264 decoder may consume more than 1 frame from the input bitstream and then propagate same timestamp to all of the consumed frames. If accurate time stamp handling is required the application has to make sure that it doesn't store more than one-frame wise data in the input bitstream.
- Target usage 7 of H.264/MVC encoders is known to have a non-monotonic quality vs. bitrate dependency.
- MPEG2 Video, VC-1 and MVC decoders are not optimized for low delay of output frames.
- Maximum supported resolution for the MPEG2 Video encoder is 4080x4080.

- The output MVC streams contain SPS and PPS headers before IDR frames only.
- MVC encoder produces different buffering period SEI messages for base and dependent views.
- MVC encoder ignores any user SEI messages for dependant view.

Other Limitations

- Limitations related to source code samples are discussed in their corresponding readme files. See "`<install-folder>\doc\MediaSDK Sample Guide.pdf`" for an overview of the samples and additional documentation.

Legal Information

Optimization Notice

Intel® compilers, associated libraries and associated development tools may include or utilize options that optimize for instruction sets that are available in both Intel® and non-Intel microprocessors (for example SIMD instruction sets), but do not optimize equally for non-Intel microprocessors. In addition, certain compiler options for Intel compilers, including some that are not specific to Intel micro-architecture, are reserved for Intel microprocessors. For a detailed description of Intel compiler options, including the instruction sets and specific microprocessors they implicate, please refer to the "Intel® Compiler User and Reference Guides" under "Compiler Options." Many library routines that are part of Intel® compiler products are more highly optimized for Intel microprocessors than for other microprocessors. While the compilers and libraries in Intel® compiler products offer optimizations for both Intel and Intel-compatible microprocessors, depending on the options you select, your code and other factors, you likely will get extra performance on Intel microprocessors.

Intel® compilers, associated libraries and associated development tools 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 Intel® Streaming SIMD Extensions 2 (Intel® SSE2), Intel® Streaming SIMD Extensions 3 (Intel® SSE3), and Supplemental Streaming SIMD Extensions 3 (Intel® 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.

While Intel believes our compilers and libraries are excellent choices to assist in obtaining the best performance on Intel® and non-Intel microprocessors, Intel recommends that you evaluate other compilers and libraries to determine which best meet your requirements. We hope to win your business by striving to offer the best performance of any compiler or library; please let us know if you find we do not.

Notice revision #20101101