Intel® Graphics Performance Analyzers (Intel® GPA) 2016 R3 Release Notes

Thank you for choosing the Intel® Graphics Performance Analyzers (Intel® GPA), available as a standalone product and as part of Intel® System Studio.

Contents

Introduction

Intel® GPA provides tools for graphics analysis and optimization that can help you make games and other graphics-intensive applications run even faster. The tools support the platforms based on the latest generations of Intel® Core™ and Intel Atom™ processor families, for applications developed for Windows*, Android*, or Ubuntu* OS.

Intel® GPA provides a common and integrated user interface for collecting performance data. Using them, you can quickly see performance opportunities in your application, saving your time and getting your product to market faster.

For detailed information and assistance in using the product, refer to the following online resources:

- Home Page - view detailed information about the tool, including links to training and support resources, as well as videos on the product to help you get started quickly.
- Getting Started - get the main features overview and learn how to start using the tools on different host systems.
- Training and Documentation - learn at your level with Getting Started guides, videos and tutorials.
- Online Help for Windows* OS Host - get details on how to analyze Windows* OS and Android* applications from a Windows* OS system.
- Online Help for OS* X Host - get details on how to analyze Android* applications from an OS* X system.
- Online Help for Ubuntu* OS Host - get details on how to analyze Android* or Ubuntu* applications from an Ubuntu* OS system.
- Knowledge Base articles - explore a list of useful technical articles, including tips & tricks and workarounds to help you get the most out of the product.
- Support Forum - report issues and get help with using Intel® GPA.
- Frequently Asked Questions - see the list of frequently asked questions about Intel® GPA.

What's New

Intel® GPA 2016 R3 offers the following new features:

New Features for Analyzing All Graphics APIs

Graphics Frame Analyzer:

- Enabled API log export into a text file.
- Added Pixel Overdraw view for render targets.
- Updated Metrics Viewer in accordance with the latest performance optimization guide.

New Features for Analyzing Microsoft DirectX* 12 Applications

All Tools:

- Added support for Universal Windows Platform (UWP) apps.
- Added support for Windows 10 Anniversary Update. Make sure to capture frames with Intel® GPA 2016 R3 as previously captured frames are not compatible with this version.
- Added support for the 7th Generation Intel® Core™ i7 Processors.

Graphics Frame Analyzer:

- Implemented extended grouping in the Bar Chart. You can now group by Render targets, Pipeline States and Shader Sets.
- Added Wireframe and Metrics map views into the Render Target viewer.
- Added Command Queue debug region visualization into the API log.
- Improved geometry support.
- Implemented API call parameters visualization.
- Added resource labels to the API log and Resource viewer.

New Features for Analyzing DirectX 9/10/11 Applications

Graphics Frame Analyzer:

- Unified the opening mechanism for all frames. You can now open all frames using the Graphics Frame Analyzer, regardless of the DirectX* version. Intel(R) GPA automatically opens DirectX 9/10/11 frames in the legacy version of Graphics Frame Analyzer.

New Features for Analyzing OpenGL/OpenGL ES* Applications

Graphics Frame Analyzer:

- Added support for glBegin/glEnd.
- Implemented visualization of assembly ARB shaders. Shaders are available on read-only mode.
- Added support for GL_Texture_1D.
- Added support for Multiple RT in highlighting and Simple FS experiments.

Documentation Updates

- Documented all newly added product features.

System Requirements and Supported Platforms

The minimum system requirements are:

- **Host Processor:** Intel® Core™ Processor

- **Target Processor**: see the list of supported Windows and Android devices below

- **System Memory:** 8GB RAM

- **Video Memory:** 512MB RAM

- **Minimum display resolution for client system:** 1280x1024

- **Disk Space:** 300MB for minimal product installation

Direct installation of Intel® GPA on 32-bit Windows* systems is not supported. However, if you need to analyze an application on a 32-bit Windows* target system, you can use the following workaround:

1. Copy the 32-bit *.msi installer distributed with the 64-bit installation from your analysis system to the target system.

2. Run the installer on the target system to install System Analyzer and Graphics Monitor.

3. Start the Graphics Monitor and the target application on the 32-bit system and connect to it from the 64-bit host system.

For details, see the Running System Analyzer on a Windows* 32-bit System article.

The table below shows platforms and applications supported by Intel® GPA 2016 R3.

Target System (the system where your game runs)	Host System (your development system where you run the analysis)	Target Application (types of supported applications running on the target system)
Windows* 7 SP1 OS	Windows* 7 SP1/8/8.1/10 OS	Microsoft* DirectX* 9/9Ex, 10.0/10.1, 11.0/11.1/11.2
Windows* 8/8.1/10 OS	Windows* 7 SP1/8/8.1/10 OS	Microsoft* DirectX* 9/9Ex, 10.0/10.1, 11.0/11.1/11.2
Windows* 10 OS	Windows* 10 OS	Microsoft* DirectX* 12 For details, see the What's New section above.
Google* Android* 4.1, 4.2, 4.3, 4.4, 5.x, 6.0 The specific version depends on the officially released OS for commercial version of Android* phones and	Windows* 7 SP1/8/8.1/10 OS or OS X* 10.10, 10.11 or Ubuntu* OS 14.04, 15.04, 16.04	OpenGL* ES 1.0, 1.1, 2.0, 3.0, 3.1, 3.2

tablets. See the list of supported devices below. NOTE: Graphics Frame Analyzer for OpenGL* does not currently support GPU metrics for the Intel® processor codenamed Clover Trail+.		
Ubuntu* OS 14.04, 15.04, 16.04	Ubuntu* OS 14.04, 15.04, 16.04	OpenGL* 3.2, 3.3, 4.0, 4.1 Core Profile

Intel® GPA does not support the following Windows* OS configurations: all server editions, Windows* 8 RT, or Windows* 7 starter kit.

For details about the tools availability for your host/target system configuration, please see this article.

Supported Windows* OS Graphics Devices

Intel® GPA supports the following graphics devices as targets for analyzing Windows* OS workloads. All these targets have enhanced metric support:

- Intel HD Graphics 2500/4000 (3rd Generation Intel® Core™ Processors)

- Intel HD Graphics 4200/4400/4600/5000 (4th Generation Intel® Core™ Processors)

- Intel Iris™ Graphics 5100 (4th Generation Intel® Core™ Processors)

- Intel Iris™ Pro Graphics 5200 (4th Generation Intel® Core™ Processors)

- Intel HD Graphics 5300 for Intel® Core™ M Processors

- Intel HD Graphics 5500 for 5th Generation Intel® Core™ Processors

- Intel HD Graphics 6000 for 5th Generation Intel® Core™ Processors

- Intel Iris Graphics 6100 for 5th Generation Intel® Core™ Processors

- Intel HD Graphics for Intel® Pentium® Processor N3700 and Intel® Celeron® Processors N3000/N3050/N3150

- Intel HD Graphics 530 for 6th Generation Intel® Core™ Processors

- Intel HD Graphics 515 for Intel® Core™ M Processors

- Intel HD Graphics 620 for 7th Generation Intel® Core™ i7 Processors

Although the tools may appear to work with other graphics devices, these devices are unsupported. Some features and metrics may not be available on unsupported platforms. If you run into in an issue when using the tools with any supported configuration, please report this issue through the Support Forum.

Driver Requirements for Intel HD Graphics

When running Intel® GPA on platforms with supported Intel HD Graphics, the tools require the latest graphics drivers for proper operation. You may download and install the latest graphics drivers from http://downloadcenter.intel.com/.

Intel® GPA inspects your current driver version and notify you if your driver is out-of-date.

Supported Devices Based on Intel® Atom™ Processor

Intel® GPA supports the following devices based on Intel Atom processor:

Processor Model	GPU	Android* Version	Supported Tools
Intel® Atom™ Z35XX (based on Intel® microarchitecture codenamed Moorefield)	Imagination Technologies* PowerVR G6430	Android* 4.4 (KitKat), Android* 5.x (Lollipop)	System Analyzer Graphics Frame Analyzer Platform Analyzer
Intel® Atom™ Z36XXX/Z37XXX (based on Intel® microarchitecture codenamed Baytrail)	Intel® HD Graphics	Android* 4.2.2 (Jelly Bean MR1) Android* 4.4 (KitKat) Android* 5.x (Lollipop)	System Analyzer Graphics Frame Analyzer Platform Analyzer
Intel® Atom™ Z25XX (based	Imagination Technologies*	Android* 4.2.2 (Jelly Bean	System Analyzer

| on Intel® microarchitecture codenamed Clovertrail+) | PowerVR SGX544MP2 | MR1) Android* 4.4 (KitKat) | Graphics Frame Analyzer Platform Analyzer |
| Intel Intel® Atom™ x7-Z8700, x5-Z8500, and x5-Z8300 (based on Intel® microarchitecture codenamed CherryTrail) | Intel® HD Graphics | Android* 5.x (Lollipop), Android* 6.0 (Marshmallow) | System Analyzer Graphics Frame Analyzer Platform Analyzer |

Supported ARM*-Based Devices
The following devices are supported with Intel® GPA:

Model	GPU	Android* Version
Samsung* Galaxy S5	Qualcomm* Adreno 330	Android* 4.4
Samsung* Galaxy Nexus (GT-i9250)	Imagination Technologies* PowerVR SGX540	Android* 4.3
Samsung* Galaxy S4 (GT-I9500ZNASER)	Imagination Technologies* PowerVR SGX544	Android* 4.4
Samsung* Galaxy S III (GT-i9300)	ARM* Mali 400MP	Android* 4.3
Google* Nexus 5	Qualcomm* Adreno 330	Android* 5.1 (Lollipop)
Nvidia* Shield tablet	NVIDIA* Tegra* K1 processor	Android* 5.0

Your system configuration should satisfy the following requirements:

- Your ARM*-based device is running Android* 4.0, 4.1, 4.2, 4.3, 4.4, 5.0, or 5.1.
- Your host system is 64-bit and is running Windows* 7 SP1/8/8.1/10 OS
- Your Android* application uses OpenGL* ES 1.0, 1.1, 2.0, 3.0, 3.1, or 3.2

For support level details for ARM*-based devices, see this article.

Supported Ubuntu* OS Graphics Devices
Intel® GPA support the following graphics devices as targets for analyzing Ubuntu* OS workloads. All these targets have enhanced metric support:

- 4th Generation Intel® Core™ Processors
- 5th Generation Intel® Core™ Processors
- 6th Generation Intel® Core™ Processors

Although the tools may appear to work with other graphics devices, these devices are unsupported. Some features and metrics may not be available on unsupported platforms. If you run into in an issue when using the tools with any supported configuration, please report this issue through the Support Forum.

Installation Notes

Installing Intel® GPA
Download and install the Intel® GPA installer from the Intel GPA Home Page.

Installing Intel® GPA on Windows* OS Target and Host Systems
To install the tools on Windows* OS, download the *.msi package from the Intel GPA Home Page and run the installer file.

The following prerequisites should be installed before you run the installer:

- Microsoft DirectX* Runtime June 2010
- Microsoft .NET 4.0 (via redirection to an external web site for download and installation)

If you use the product in a host/target configuration, install Intel® GPA on both systems. For more information on the host/target configuration, refer to Best Practices.

For details on how to set up an Android* device for analysis with Intel® GPA, see Configuring Target and Analysis Systems.

Installing Intel® GPA on Ubuntu* OS Host System

To install Intel® GPA on Ubuntu* OS:

1. Make sure that the `sharutils` package is installed on your host system.

2. Change file permissions by running the command `chmod 755 intel-gpa_16.3_x64.deb.shar`

3. Execute the `.shar` file by running the command `./intel-gpa_16.3_x64.deb.shar`

4. Read the end-user license agreement (EULA), and if you agree to the terms and conditions of the EULA, enter `Y` to accept the agreement.

5. Double-click the extracted `.deb` package to install it using the Ubuntu* OS package manager.

NOTE: You do not need to explicitly install Intel® GPA on the Android* target device since the tools automatically install the necessary files on the target device when you run System Analyzer. For details on how to set up an Android* device for analysis with Intel® GPA, see Configuring Target and Analysis Systems.

Installing Intel® GPA on OS X* Client Platform

To install the tools on OS X*, download the .pkg package and run the installer.

NOTE: You do not need to explicitly install Intel® GPA on the Android* target device since the tools automatically install the necessary files on the target device when you run the System Analyzer. For details on how to set up an Android* device for analysis with Intel® GPA, see Configuring Target and Analysis Systems.

Technical Support and Troubleshooting

For technical support, including answers to questions not addressed in the installed product, visit the Support Forum.

You can find other useful support information in the Frequently Asked Questions and Knowledge Base articles.

Troubleshooting Android* Connection Problems

If the target device does not appear when the "`adb devices`" command is executed on the client system, do the following:

1. Disconnect the device

2. Execute "`adb kill-server`"

3. Reconnect the device

4. Run "`adb devices`"

If these steps do not help, restart the system and then run "`adb devices`" again. You may also need to consult product documentation for your device to see if a custom USB driver needs to be installed on devices based on Intel architecture and third-party ARM*-based devices.

Known Issues and Limitations

General

- Selecting all ergs might cause a significant memory usage in Graphics Frame Analyzer.
- Intel® GPA uses sophisticated techniques for analyzing graphics performance which may conflict with third-party performance analyzers. Therefore, ensure that other performance analyzers are disabled prior to running any of these tools. For third-party graphics, consult the vendor's website.
- Intel® GPA does not support use of Remote Desktop Connection.
- Graphics Frame Analyzer (DirectX* 9,10,11) runs best on systems with a minimum of 4GB of physical memory. Additionally, consider running the Graphics Frame Analyzer (DirectX* 9,10,11) in a networked configuration (the server is your target graphics device, and the client running the Graphics Frame Analyzer is a 64-bit OS with at least 8GB of memory).
- On 64-bit operating systems with less than 8GB of memory, warning messages, parse errors, very long load times, or other issues may occur when loading a large or complex frame capture file.

Analyzing Android* Workloads

- Graphics Frame Analyzer does not currently support viewing all OpenGL/OpenGL ES* texture formats.
- Intel® GPA provides limited support for analyzing browser workloads on Android*. You can view metrics in the System Analyzer, but the tools do not support creating or viewing frame capture files or trace capture files for browser workloads. Attempting to create or view these files may result in incorrect results or program crashes.
- Intel® GPA may fail to analyze OpenGL* multi-context games.

If you want to profile Android* workloads from an OS X* system, make sure to install Platform Analyzer.
If you choose not to install Platform Analyzer, you are recommended to disable tracing support in System Analyzer. Press Crtl+F1 in the System Analyzer Connection window and deselect the "Enable tracing on Android devices" checkbox.

Analyzing Windows* OS Workloads

- Texture 2x2 experiment might work incorrectly for some DirectX 12 workloads.
- Overdraw experiment for Render Targets with 16-bit and 32-bit Alpha channel is not supported now.
- To view Render Targets with 16-bit and 32-bit Alpha channel you should disable Alpha channel in the Render Targets viewer.
- DirectX 12.1 API is not supported now.
- To ensure accurate measurements on platforms based on Intel® HD Graphics, profile your application in the full-screen mode. If windowed mode is required, make sure only your application is running. Intel® GPA does not support profiling multiple applications simultaneously.
- For best results when analyzing frame or trace capture files on the same system where you run your game, follow these steps:
  - Run your game and capture a frame or trace file.
  - Shut down your game and other non-essential applications.
  - Launch the Intel® GPA.
- To run Intel® GPA on hybrid graphics solutions (a combination of Intel Processor Graphics and third-party discrete graphics), you must first disable one of the graphics solutions.
- Secure Boot, also known as Trusted Boot, is a security feature in Windows* 8 enabled in BIOS settings which can cause unpredictable behavior when the **"Auto-detect launched applications"** option is enabled in Graphics Monitor Preferences. Disable Secure Boot in the BIOS to use the auto-detection feature for analyzing application performance with Intel® GPA. The current version of the tools can now detect Secure Boot, and will warn you of this situation.
- To view the full metric set with the tools for Intel Processor Graphics on systems with one or more third-party graphics device(s) and platforms based on Intel® HD Graphics, ensure that Intel is the preferred graphics processor. You can set this in the Control Panel application for the third-party hardware. Applications running under Graphics Monitor and a third-party device show GPU metrics on DirectX* 9 as initialized to 0 and on DirectX* 10/11 as unavailable.
- When using the Intel® GPA, disable the screen saver and power management features on the target system running the Graphics Monitor — the Screen Saver interferes with the quality of the metrics data being collected. In addition, if the target system is locked (which may happen when a Screen Saver starts), the connection from the host system to the target system will be terminated.
- Intel® GPA does not support frame capture or analysis for:
  - applications that execute on the Debug D3D runtime system
  - applications that use the Reference D3D Device
- System Analyzer HUD may not operate properly when applications use copy-protection, anti-debugging mechanisms, or non-standard encrypted launching schemes.
- Intel® GPA provides analysis functionality by inserting itself between your application and Microsoft DirectX*. Therefore, the tools may not work correctly with certain applications which themselves hook or intercept DirectX* APIs or interfaces.
- Intel® GPA does not support Windows* 8 OS Store Applications where the graphics API uses compositing techniques such as HTML5 or XAML interop. Only traditional DirectX* rendering is supported. To workaround this limitation, port your application as a Desktop application, and then use the full Intel GPA suite of tools.
- In some cases, the Overview tab in Graphics Frame Analyzer (DirectX* 9,10,11) can present GPU Duration values higher than Frame Duration values measured during game run time. This could be a result of Graphics Frame Analyzer (DirectX* 9,10,11) playing the captured frame back in off-screen mode which can be slower than on-screen rendering done in the game.
  To make playback run on-screen use this registry setting on the target system:
  HKEY_CURRENT_USER\Software\Intel\GPA\16.3\ForceOnScreenPlaybackForRemoteFA = 1 and connect to the target with Graphics Frame Analyzer (DirectX* 9,10,11) running on a separate host. If these requirements are met, the playback will run in off-screen mode on the target. If the frame was captured from the full-screen game, but playback renders it in a windowed mode, then try to press Alt+Enter on the target to switch playback to full-screen mode.

- Frame capture using Graphics Monitor runs best on 64-bit operating systems with a minimum of 4GB of physical memory. On 32-bit operating systems (or 64-bit operating systems with <4GB of memory), out of memory or capture failed messages can occur.
- A defect in the Unity* 4 Integration with Oculus PC SDK ver. 0.5 prevents proper capture in Unity VR titles. Only Oculus* hardware is verified. Other VR devices might work but have not been tested.

Analyzing Ubuntu* Workloads

- Tracing functionality is currently not supported for Ubuntu* targets.
- GPU metrics for Ubuntu* targets are available on Intel HD Graphics 2500/4000 platforms only.
- When analyzing frames for Unigine Heaven* and Valley* benchmarks on systems with 4th Generation Intel Core™ Processors, edit /etc/drirc configuration file to set the flags for the local player to the exact values that Unigine uses for this application. For example:

```xml
<application name="" executable="glesplayer">
        <option name="force_glsl_extensions_warn" value="true" />
        <option name="disable_blend_func_extended" value="true" />
        <option name="force_glsl_version" value="130" />
        <option name="disable_shader_bit_encoding" value="true" />
        <option name="allow_glsl_extension_directive_midshader" value="true" />
</application>
```

Otherwise, Graphics Frame Analyzer may fail to play the frame correctly.