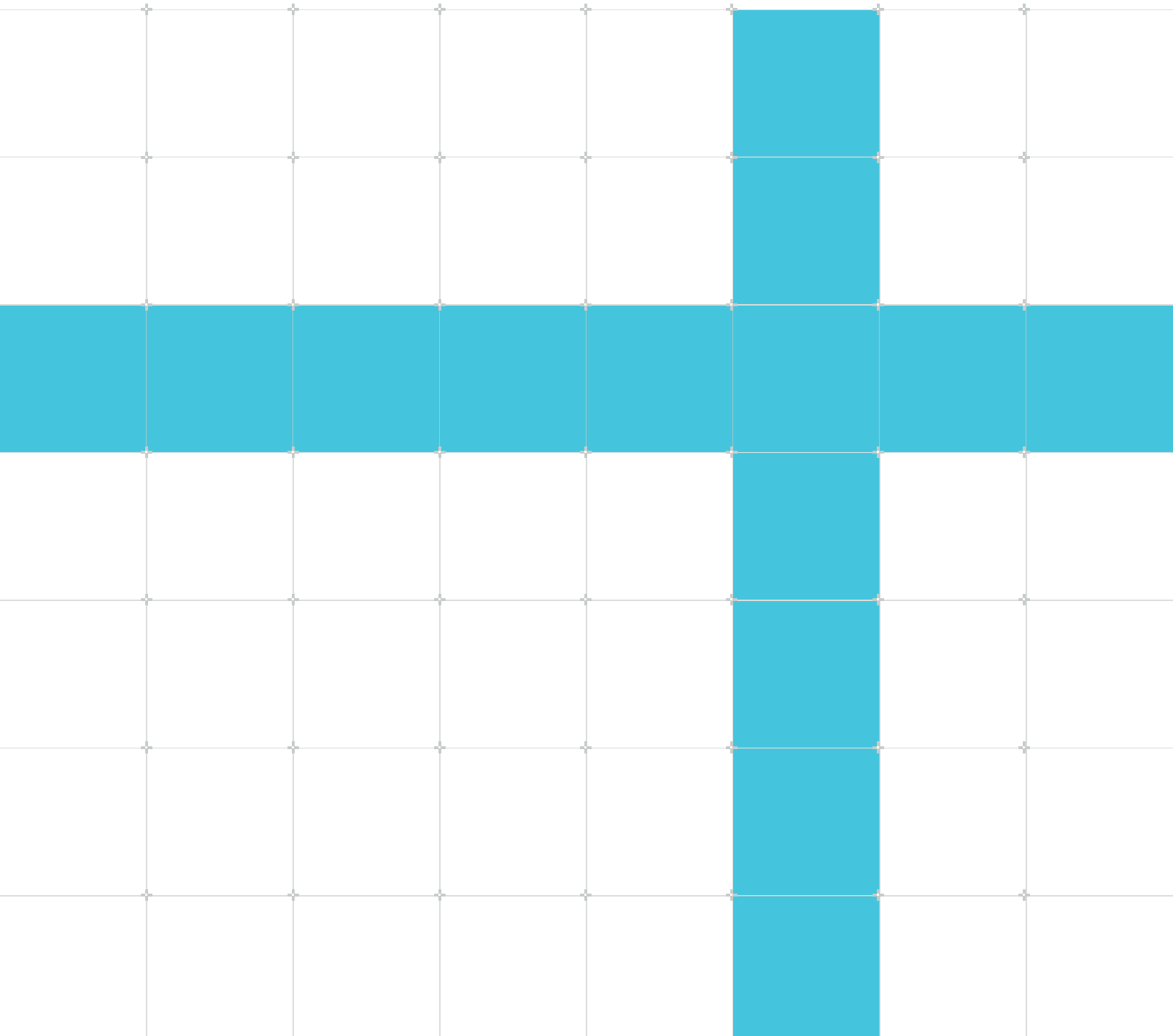
Arm Mobile Studio FAQ

2023.x

Issue 01
102718_2023.x_01_en



Arm Mobile Studio FAQ

Release information

Document history

Issue	Date	Confidentiality	Change
0102-00	28 June 2022	Non-Confidential	LWI directory update
0102-01	30 November 2022	Non-Confidential	Remove obsolete FAQ, Pro edition change, install instructions.
2023-00	10 May 2023	Non-Confidential	Performance Advisor has moved into Streamline. Add migration instructions, get help chapter and update other FAQs. Remove obsolete FAQs.
2023-01	16 June 2023	Non-Confidential	Add FAQs for script merge and no debuggable applications listed in Streamline.

Proprietary Notice

This document may include technical inaccuracies or typographical errors.

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL ARM BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF ARM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document consists solely of commercial items. You shall be responsible for ensuring that any use, duplication or disclosure of this document complies fully with any relevant export laws and regulations to assure that this document or any portion thereof is not exported, directly or indirectly, in violation of such export laws. Use of the word “partner” in reference to Arm’s customers is not intended to create or refer to any partnership relationship with any other company. Arm may make changes to this document at any time and without notice.

This document may be translated into other languages for convenience, and you agree that if there is any conflict between the English version of this document and any translation, the terms of the English version of the Agreement shall prevail.

Arm Limited. Company 02557590 registered in England.

110 Fulbourn Road, Cambridge, England CB1 9NJ.

(LES-PRE-20349|version 21.0)

Confidentiality Status

This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by Arm and the party that Arm delivered this document to.

Unrestricted Access is an Arm internal classification.

Product Status

The information in this document is Final, that is for a developed product.

Feedback

Arm welcomes feedback on this product and its documentation. To provide feedback on the product, create a ticket on https://support.developer.arm.com

To provide feedback on the document, fill the following survey: https://developer.arm.com/documentation-feedback-survey.

Inclusive language commitment

Arm values inclusive communities. Arm recognizes that we and our industry have used language that can be offensive. Arm strives to lead the industry and create change.

We believe that this document contains no offensive language. To report offensive language in this document, email terms@arm.com.

Contents

1. Overview

Here are answers to common questions about Arm Mobile Studio. This document relates to versions 2023.x of Arm Mobile Studio.

Please ensure you are using the latest version of Arm Mobile Studio, which you can download here. Note that you need to log in with an Arm account to access the download.

2. How to install Arm Mobile Studio

From version 2022.4 onwards, installation instructions for each platform can be found in the Arm Mobile Studio Release Notes. Access them from your installation directory, or from the Arm Developer website.

Related information

- Go to the product download page.
- More FAQs.
- Ask a question on the Arm Community forum.

3. Device connection issues

In some cases, Arm Mobile Studio tools can not find the connected Android device. Here are some common reasons why this happens, and how to resolve them.

Condition

Affected releases: All.

Check that ADB is installed

Check that you have Android Debug Bridge (ADB) installed and working correctly:

1. In a command terminal, type `adb --help` to check that ADB is installed - this should return the list of options for the `adb` command.

   If you can not run the `adb` command successfully, it may not be installed, or you may not be able to run it from your current location.
2. Install Android Debug Bridge (ADB). ADB is available with the Android SDK platform tools, which are installed as part of Android Studio, or you can download them separately here.
3. Edit your `PATH` environment variable to add the path to the ADB executable, so that you can run the `adb` command from any directory.

Check your device is set up correctly

Do the following:

1. Update the firmware on your device to the latest version.
2. Check that the device is switched on, connected to your host machine through USB, and is running Android 8 or later.
3. Ensure Developer Mode is enabled on the device, and enable USB Debugging using Settings > Developer options.
4. Type `adb devices` into a terminal on your host machine, to return the ID of any connected devices.

   ```
   adb devices
   List of devices attached
   ce12345abcdf1a1234       device
   ```

   If you see that the device is listed as unauthorized, try disabling and re-enabling USB debugging on the device, and accept the authorization prompt on the device to enable the connection.

For more information, refer to the setup instructions in the Performance Advisor get started tutorial.

Errors running the connection script

Performance Advisor uses a connection script to set up the device ready for capture. Use the `lwi_me.py` script as described in the Get started guide to do this. If the script fails, check the following:

1. Ensure that Python 3.6 or later is installed, and the path to the Python executable is added to your `PATH` environment variable, so that you can run Python commands from any directory. To test Python, in a terminal, type `python --help` which returns the list of available command options.
2. Ensure you are using the latest version of Mobile Studio. You need to log in to the Arm Developer website to access this download.
3. Ensure that you are running the `lwi_me.py` script from its location in the Arm Mobile Studio installation directory: `<install_directory>/streamline/bin/android`. Do not move the script to another location, as it relies on the current directory structure to locate other required files that exist within the Arm Mobile Studio installation directory.

Device support

As Arm-based processors are very widespread in the smartphone industry it is impossible for us to test our tools on all devices available for sale to the public. Check our supported devices page for a list of devices we have tested and confirm will work with Arm Mobile Studio.

Get help

If you're still experiencing connection issues, ask for help on the Arm Community graphics forum.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

4. Why can I no longer buy professional edition?

Arm Mobile Studio professional edition is no longer available.

Cause

From Arm Mobile Studio version 2022.4, ll the professional edition features are now included in the free version. The professional version of Arm Mobile Studio has ceased to exist.

Workaround

To access these features, simply download the free version of Arm Mobile Studio.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

5. Security warning when starting Arm Mobile Studio tools on macOS

When launching Arm Mobile Studio tools on macOS 10.15 (Catalina), in some cases, the application fails to start, and issues the following error message:

```
"<toolname>" cannot be opened because the developer cannot be verified.
```

or

```
"toolname" is an app downloaded from the internet, are you sure you want to open it.
```

Condition

Affected releases: Mobile Studio (all versions).

Affects macOS 10.15 (Catalina) and later platforms.

Cause

macOS Catalina introduced changes to its Gatekeeper functionality. Because some of the Arm Mobile Studio tool binaries are not in the '.app' format, macOS issues a security warning. You can verify the integrity of the binary by entering the following command in a Terminal window:

```
spctl -a -vv <toolname>
<toolname>: rejected (the code is valid, but does not seem to be an app)
origin=Developer ID Application: ARM Ltd (S345482SL3)
```

Solution

If you get the second message listed above, you can simply click Open to enable the tool.

If you get the first message listed above, enable the tool in your macOS security settings:

1. Go to System Settings > Privacy & Security.
2. Scroll down to Security and check if the tool is listed here.
3. Enable access to the tool.

If this doesn't work, gatekeeper checks can be disabled by manually removing the `com.apple.quarantine` flag from the installed binary. Open a Terminal window, and run the following commands to disable gatekeeper:

1. Navigate to the Arm Mobile Studio tool installation directory, for example, Mali Offline Compiler:

   ```
   cd /Applications/Arm_Mobile_Studio_2023.1/mali_offline_compiler/
   ```

2. Enter the following command:

```
sudo xattr -d com.apple.quarantine malioc
```

Related information

- Get started with Mali Offline Compiler
- More FAQs
- Ask a question on the Arm Community forum.

6. The Performance Advisor pa command no longer exists

After upgrading Arm Mobile Studio to version 2023.1 or later, the Performance Advisor `pa` command is no longer supported.

Condition

Affects versions: 2023.1 and later

Cause

From Arm Mobile Studio version 2023.1, Performance Advisor is part of Streamline. This means that there are changes to the installation directory, and the `pa` command used to generate reports with Performance Advisor has been replaced by Streamline's equivalent command `Streamline-cli -pa`.

Workaround

Ideally you should upgrade to the latest version of Arm Mobile Studio as we are continually adding fixes, performance enhancements and new functionality. But, if you are upgrading from version 2023.0 or earlier, and you have existing scripts that you use to generate Performance Advisor reports, you will need to update them to:

1. Run the `lwi_me.py` script from its new location in the installation directory, `<installation_directory>/streamline/bin/android`
2. Wherever you use the `pa` command to generate a report, replace this with the new report generation command, `Streamline-cli -pa`. For example:

   ```
   Streamline-cli -pa capture.apc [options]
   ```

   All the existing command-line options are still supported.
3. Update your `PATH` environment variable to point to the location of the `Streamline-cli` executable, so that you can run this command from any location. The new location is `<installation_directory>/streamline/`. This step is not necessary on Windows, as it is updated automatically when you install Arm Mobile Studio. On macOS, you can use the `Streamline-cli-launcher` file to set this for you.
4. If you need to point to the OpenGL ES or Vulkan layer library files, these are now located in `<installation_directory>/streamline/bin/android/<arm|arm64>`. There are different libraries for 32-bit (arm) and 64-bit (arm64) applications.

For full instructions on how to work with Performance Advisor, refer to the Get started with Performance Advisor tutorial.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

7. The lwi_me.py and gator_me.py scripts no longer exist

After upgrading Arm Mobile Studio to version 2023.2 or later, the Performance Advisor `lwi_me.py` script and the Streamline `gator_me.py` script are no longer available.

Condition

Affects Arm Mobile Studio versions 2023.2 and later.

Cause

From Arm Mobile Studio version 2023.2, the `lwi_me.py` and `gator_me.py` scripts have been merged into a single script named `streamline_me.py`.

Solution

Use the `streamline_me.py` script in place of `lwi_me.py` and/or `gator_me.py`.

The only command-line option that is different, is `--lwi-mode`, which is now set to `off` by default. If you have manually included frame boundary annotations in your application, you don't need to do anything. If you have not, then you must set this option to `counters` or `screenshots` for Performance Advisor to be able to find frame boundaries.

All the other command-line options are still supported.

For full instructions on how to work with Performance Advisor, refer to the Get started with Performance Advisor tutorial.

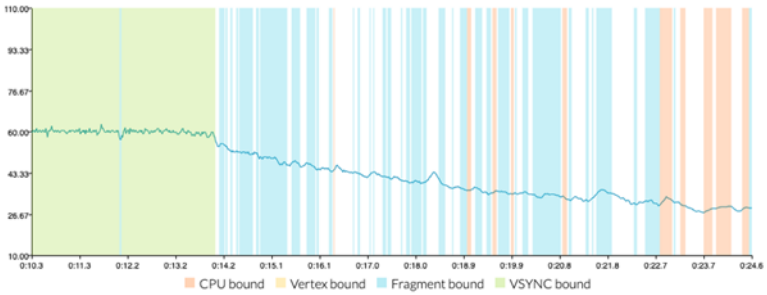
Related information

- More FAQs
- Ask a question on the Arm Community forum.

8. Blank sections in Performance Advisor FPS chart

When generating a report with Performance Advisor, some sections of the FPS chart are shown in white, instead of showing a color to indicate the type of boundness: CPU-bound, fragment-bound, vertex-bound, or performing well (VSYNC).

Figure 8-1: FPS summary chart showing blank sections



Condition

Affected releases: All.

Cause

In some cases, Performance Advisor might be unable to determine the exact cause of the problem, and has assigned a cause of 'unknown' to that region of the chart. Unknown regions might occur if:

- The data from Streamline is too noisy
- Frame markers don't align well with the workload boundaries
- There is a scheduling issue

Workaround

Do the following:

1. Update the firmware on your device to the latest version.
2. Try capturing the profile again with Streamline, using the latest version of Mobile Studio.
3. When generating the report with Performance Advisor, use the `--main-thread` option to explicitly provide the name of the main thread in your application. If there are scheduling issues, Performance Advisor will report it.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

9. Performance Advisor fails to capture OpenGL ES applications on Android 9

Performance Advisor fails to generate a performance report, giving the error 'Can't find any frames'.

Condition

Applications installed on a device running Android 9.

Cause

Performance Advisor uses the Android layer driver mechanism to inject the light-weight interceptor into the application. OpenGL ES layers are only available on Android 10 or newer.

From Arm Mobile Studio version 2023.2, the –lwi-mode=counters option enables frame boundary instrumentation. This option is required unless you have manually included frame boundary annotations in your application. Check that you have used this option when running the `streamline_me.py` script.

Workaround

OpenGL ES applications on Android 9 can use manual Streamline annotation instrumentation to emit the necessary frame boundaries that Performance Advisor requires for frame rate analysis.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

10. What is the difference between Graphics Analyzer and Mali Graphics Debugger?

Graphics Analyzer is a rebranded version of Mali Graphics Debugger, and will replace Mali Graphics Debugger for future tool releases. The name was changed to highlight that the tool is no longer restricted to supporting platforms only running on a Mali GPU, and will now enable debug of graphical applications irrespective of the GPU in the target device.

Download Graphics Analyzer

Graphics Analyzer is included with Arm Mobile Studio and Arm Development Studio.

For graphics and gaming developers, download the free Arm Mobile Studio tools and start using Graphics Analyzer today.

For embedded C/C++ software developers, try Arm Development Studio for free for 30 days.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

11. Graphics Analyzer becomes unresponsive on closing

Graphics Analyzer does not close and becomes unresponsive.

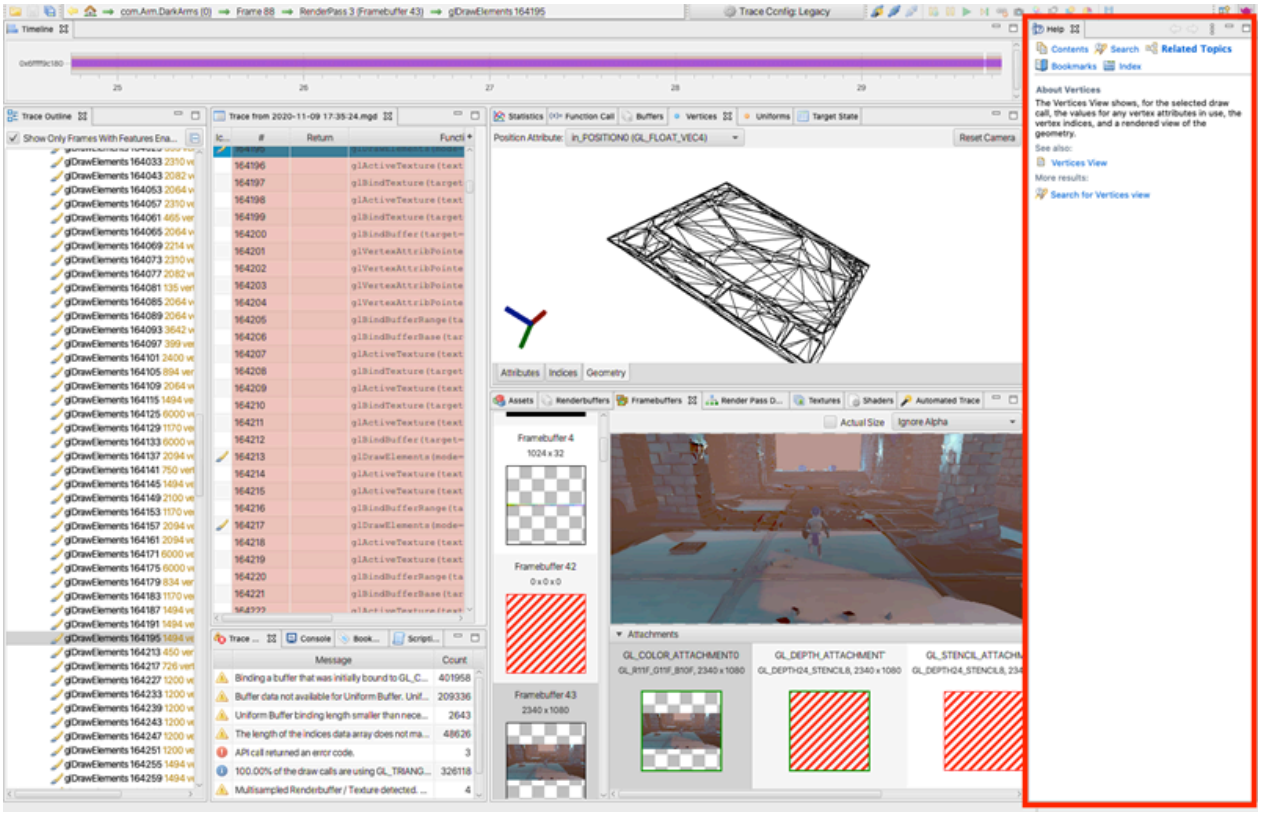
Condition

Affected platforms: Linux.

Cause

Linux platforms, if you try to close Graphics Analyzer while the dynamic help view is open, Graphics Analyzer does not close and becomes unresponsive.

Figure 11-1: Graphics Analyzer with dynamic help active



Workaround

To avoid this issue, ensure that you close the dynamic help view before closing Graphics Analyzer. Then you can close the application normally.

Related information

- More FAQs

- Ask a question on the Arm Community forum.

12. Graphics Analyzer can not access Khronos reference pages

In Graphics Analyzer, double-clicking a function does not open the Khronos reference page for that function, as expected.

Condition

Affected platforms: Windows, Internet Explorer.

Cause

Some browsers are not compatible with the Khronos reference page format. In particular, some versions of Internet Explorer do not support this functionality.

Workaround

Try installing a different browser and setting it to the default browser on your Windows machine.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

13. Graphics Analyzer playback and capture buttons unavailable

In some cases, when capturing a trace of an Unreal Engine application using Graphics Analyzer, the playback and capture buttons are not available and appear grayed-out.

Figure 13-1: Capture buttons are unavailable in Graphics Analyzer

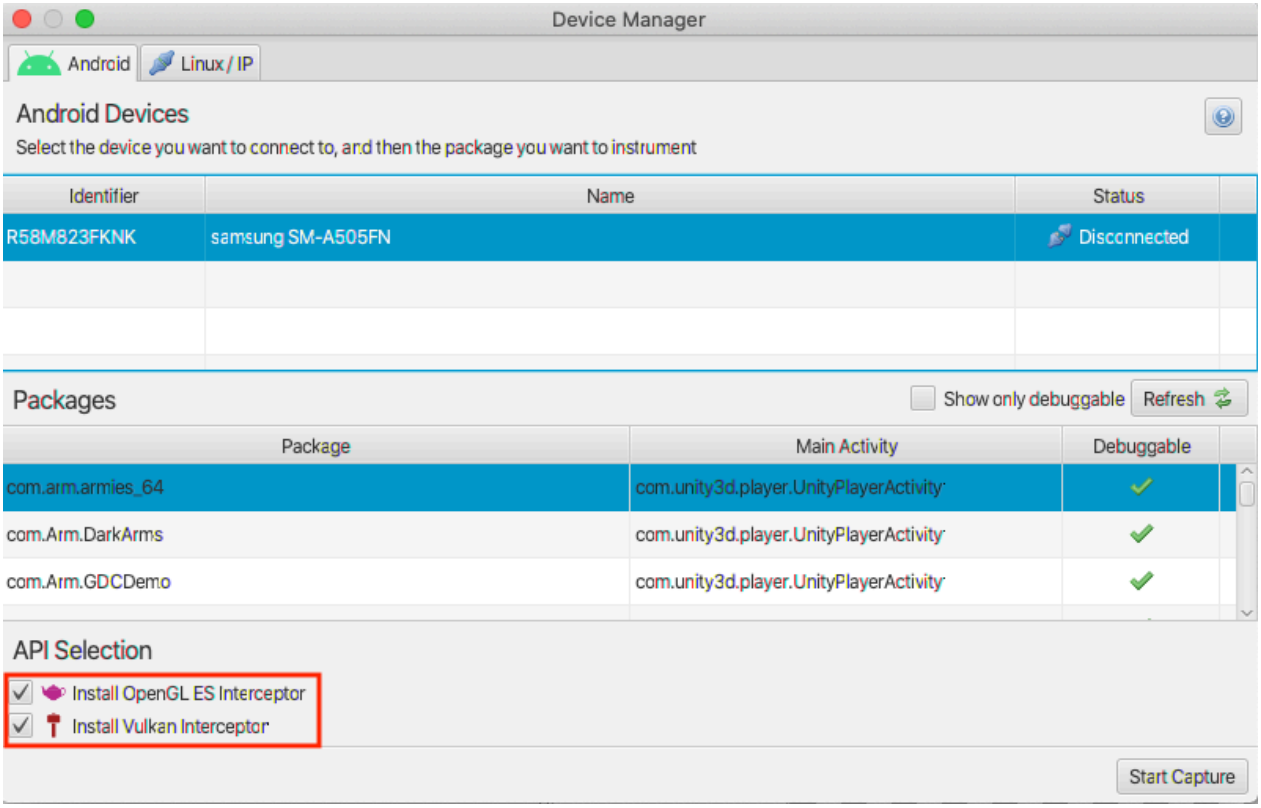
Condition

Applies to all versions of Graphics Analyzer when tracing Unreal Engine applications that use mixed Vulkan and OpenGL ES.

Cause

This problem can occur if both GLES and Vulkan APIs are selected when starting the capture.

Figure 13-2: Device Manager in Graphics Analyzer with multiple APIs selected



Workaround

The following workarounds apply:

- If your application is OpenGL ES only, deselect the option to Install Vulkan Interceptor from the Device Manager when setting up the capture
- If you need to capture both OpenGL ES and Vulkan API calls, you will need to select both interceptors on the Device Manager. In this case, use the global pause button to pause the application as soon as tracing starts, and then press play to resume. The capture buttons should now be available.

Related information

- More FAQs
- Ask a question on the Arm Community forum.

14. Graphics Analyzer fails to open on RHEL 8 CentOS 8

On CentOS 8 systems, Graphics Analyzer fails to launch and reports the following error:

```
Gdk-CRITICAL **: 12:35:22.301: gdk_x11_display_set_window_scale: assertion
 'GDK_IS_X11_DISPLAY (display)' failed
```

Condition

Arm Development Studio installed on RHEL 8 CentOS 8 systems.

Cause

A known issue related to the backend code that JavaFX uses is causing the application to crash.

Workaround

Set the environment variable `GDK_BACKEND` to `x11`:

```
export GDK_BACKEND=x11
```

Related information

- More FAQs
- Ask a question on the Arm Community forum.

15. Why is the list of debuggable Android applications in Streamline empty?

In Streamline, when choosing which device and application to profile, no debuggable applications are displayed for the selected device.

Condition

- Affected products: Arm Mobile Studio, Streamline.

Check that debuggable applications exist on the selected device

Currently, you can only profile applications that are debuggable.

To set an application to be debuggable:

- In Android Studio, create a build variant that includes `debuggable true` in the build configuration. Alternatively, you can set `android:debuggable=true` in the application manifest file.
- In Unity, select Development Build under File > Build Settings when building your application.

Check if you have multiple user profiles enabled on the device

Streamline currently does not support profiling applications that are installed under different user profiles. You can only profile applications that are installed under the default user, with user ID 0.

To check if this is the problem, look at the Streamline error log located in Window > Error Log, and search for security exceptions like the following:

```
Exception occurred while executing 'list':
java.lang.SecurityException: Shell does not have permission to access user 10
```

Workaround

Always install the application that you want to profile with the default user, user ID 0.

You can list the users on the device by running the following `adb` command:

```
adb shell pm list users
```

This returns something like:

```
Users:
UserInfo{0:UserA:c13} running
UserInfo{10:UserB:1000} running
```

To list the packages for a particular user by running the following `adb` command:

```
adb shell pm list packages -e -3 --user 10
```

Related information

- More FAQs
- Ask a question on the Arm Community forum.

16. Get help with Arm Mobile Studio

if you are still experiencing problems using Arm Mobile Studio tools, here are some resources to help you.

Tutorials

Here are some tutorials that might help you.

- Get started with Performance Advisor
- Get started with Graphics Analyzer
- Get started with Mali Offline Compiler
- Android performance triage with Streamline

User guides

Each tool has a comprehensive user guide that describes all the functionality in detail.

- Streamline user guide
- Performance Advisor user guide
- Graphics Analyzer user guide
- Mali Offline Compiler user guide

Training videos

Work through our video tutorials to see how the tools can be used to analyze a mobile game.

- Episode 3.1 - Introducing Arm Mobile Studio (10 mins)
- Episode 3.2 - Performance Advisor (11 mins)
- Episode 3.3 - Streamline (8 mins)
- Episode 3.4 - Graphics Analyzer (7 mins)
- Episode 3.5 - Mali Offline Compiler (7 mins)

There are also videos that describe mobile systems and graphics fundamentals, and explain how to design content that will run efficiently on mobile. Here are links to the first videos in modules 1 and 2 of the Mali GPU training. Use the links below each video to navigate to the next video.

- Introduction to mobile systems
- Best practice principles for mobile game development

Ask a question

If you need to talk to an Arm expert, ask a question on the Arm Community forum.

If you would rather discuss your problem in private, you can email us at mobilestudio@arm.com.