NPU/MEP Software Stack Debug Guide

Revision 0.5

March 2024

Intel Confidential

Document Number: xxxxxx

Contents

Revision History

Document Number	Revision Number	Description	Revision Date
<XXXX>	0.5	• Initial release	March 2024

§

1 Introduction

This document is aimed to provide a system level debug guide of the features related to MEP. You will learn more about how to triage and clarify issues before filing a bug and know how to collect logs for further investigation.

1.1 Reference Documents

Table 1-1. Reference Documents

Document	Document No./Location
Meteor Lake iVPU MEP Opt-in Guide for OEMs Installation Guide	776500
Meteor Lake Reference Validation Platform (RVP) Running Windows* Microsoft* Effects Package on Intel® Installation Guide	777824

2 Initial Triage

Below ingredients may affected MEP test result, we can clarify which one is the key ingredient to cause the problem to narrow down the issue as initial triage.

- **OS:** Does the issue happen on a specific OS version? For example: issue occurs after doing Windows Update.
- **NPU:** Does the issue occur with a specific NPU driver?
- **MEP:** Does the issue occur with a specific MEP driver?
- **Camera**: Any 3rd party camera driver is installed on the system? If yes, please remove it and check the result with Microsoft inbox camera driver. For camera HLK test failure issue, please test with MEP camera opt-out to clarify if it's MEP related.
- **Graphic:** Some performance (FPS, glitch, flicker…etc.) issue may related to graphic driver, we can remove it and use inbox driver to verify the result
- **Image/Others:** Because MEP can work with NPU and MEP driver installed, to see if you can reproduce the issue with install these two drivers only on a clean OS is an easy way to confirm if it's NPU/MEP issue or other drivers or applications related.

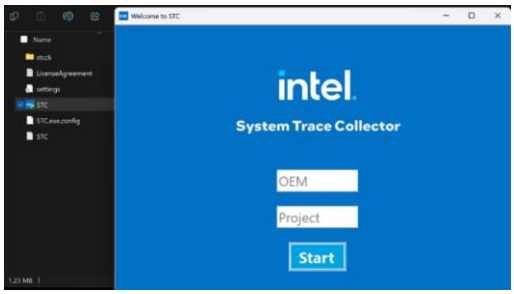
3 Collect Debug Logs

Below logs will help for bug analysis. In this section, we list detailed steps on how to collect MEP relative logs so that customer can clearly know and help to attach the log when creating a sighting. For MEP issues, we suggest to get Microsoft camera trace and NPU log as a basic log (section 3.1) for debugging. We usually need Xperf and GPUView log for checking performance issues and a memory dump for BSOD or TDR. Please collect both pass and fail logs if possible.

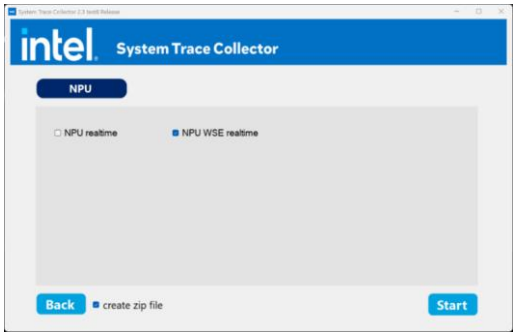
We suggest customers refer to Chapters 5 for common issue debugging and troubleshooting to make sure it's not a known issue or behavior before filing a bug.

3.1 Microsoft Camera Trace and NPU UMD, KMD log

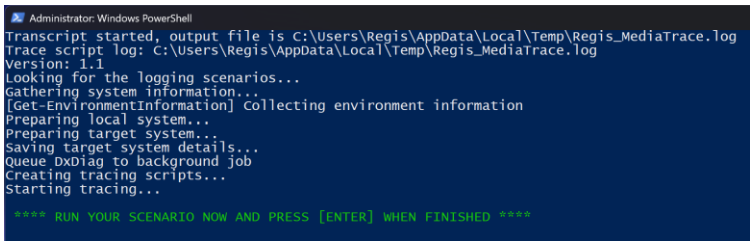
- Download Intel System Trace Collector (STC) Tool from Intel RDC Kit#765450, launch it by STC.exe and enter OEM/Project name.



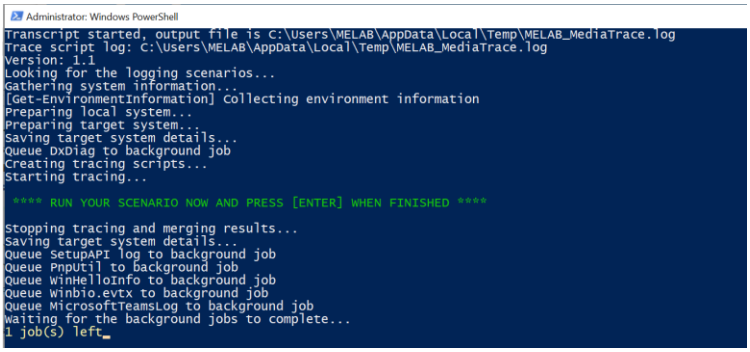
- Select "NPU WSE realtime" (this includes NPU user mode/kernel mode trace and camera trace) and then click Start button.



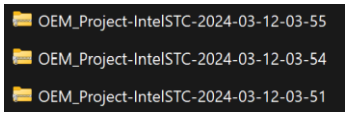
- You will see another console pop up.



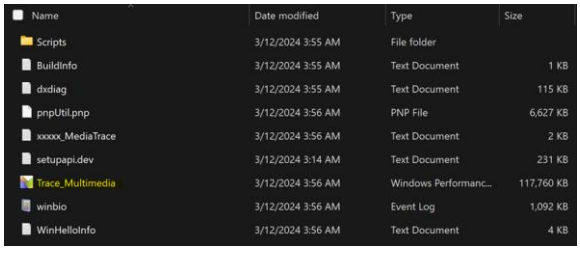
- Please reproduce the issue and press ENTER when finished.

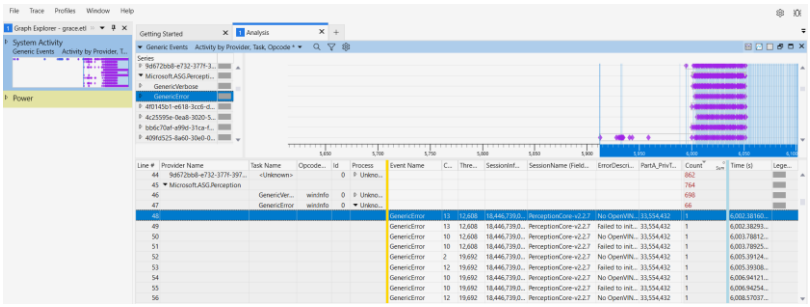


- It will auto zip the log



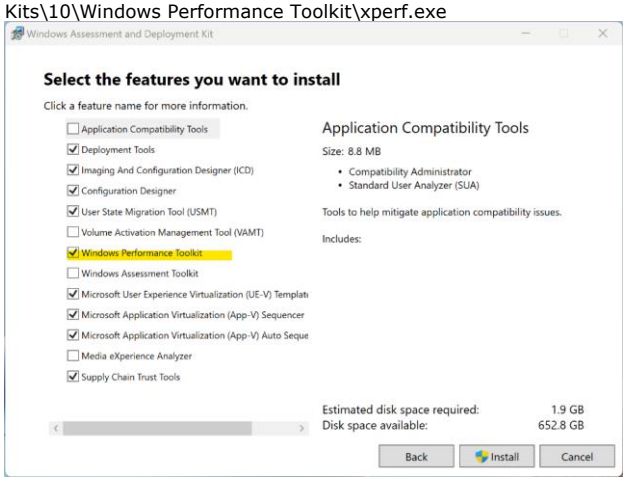
- You can use Windows Performance Analyzer (WPA) tool in ADK to open Trace_MultImedia.etl to check if any errors, abnormal events or attached log on the sighting.



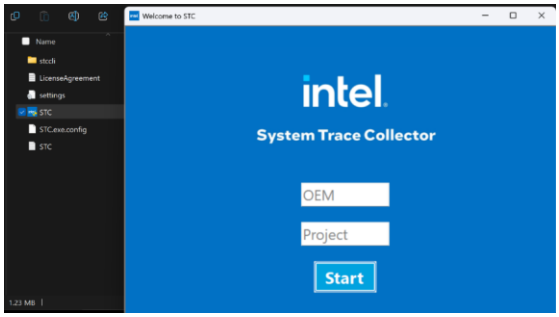


3.2 ETL Trace with Xperf

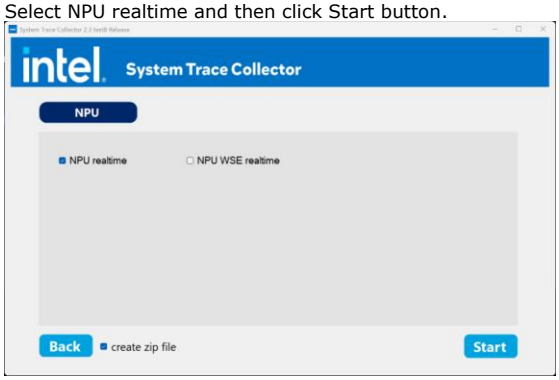
- Xperf tool from Windows ADK is required for this log. Please download Windows* Assessment and Deployment Kit (Window*s ADK), select and install the "Windows Performance Toolkit", which contains the Xperf in C:\Program Files (x86)\Windows Kits\10\Windows Performance Toolkit\xperf.exe



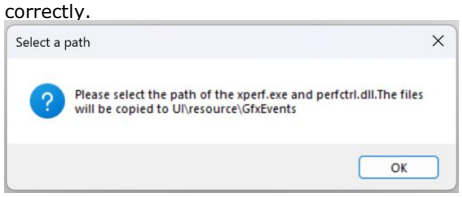
- Download Intel System Trace Collector (STC) Tool from Intel RDC Kit#765450, launch it by STC.exe and enter OEM/Project name.



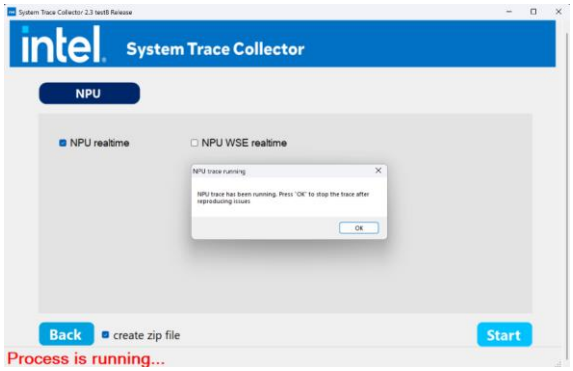
- Select NPU realtime and then click Start button.



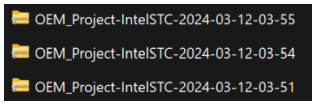
If you see this message after click "Start", please make sure Xperf is installed correctly.



- Reproduce the issue.
- Press OK to stop the trace.

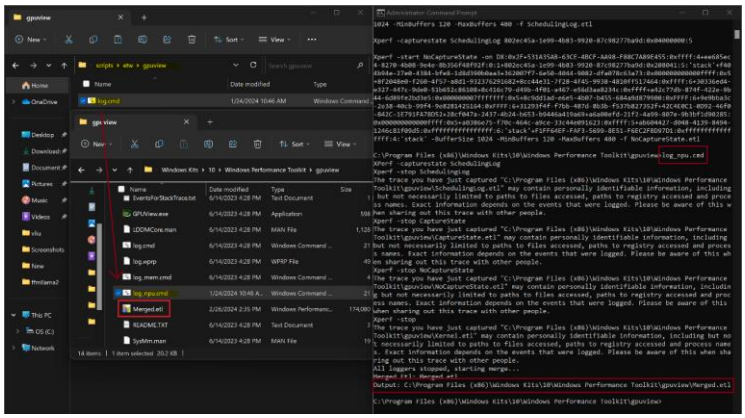


- It will auto zip the log.

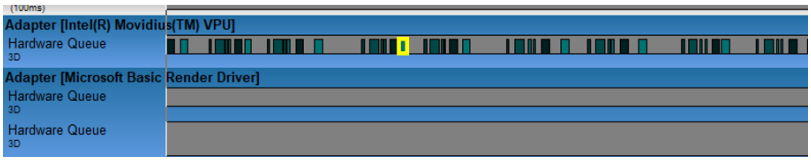


3.3 GPUView

- GPUView tool from Windows ADK is required for this log. Please refer to Chapter 3.2 to install "Windows Performance Toolkit" from ADK.
- In NPU driver release package, rename script/etw/gpuview/log.cmd to log_npu.cmd and copy it to Windows Performance Toolkit\gpuview folder.
- Execute cmd.exe as administrator and run **log_npu.cmd.**
- Reproduce the issue.
- When test complete, go back to the Admin cmd window and run **log_npu.cmd** again to generate the Merged.etl file (Please zip this file for analysis)

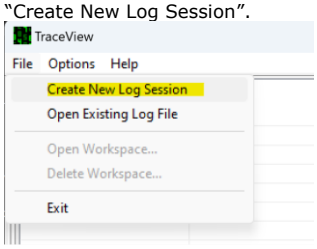


GPUView is useful to debug performance issues. The adaptor NPU will record the inference event, click the block can get this inference details, including execution time, submission and complete time.

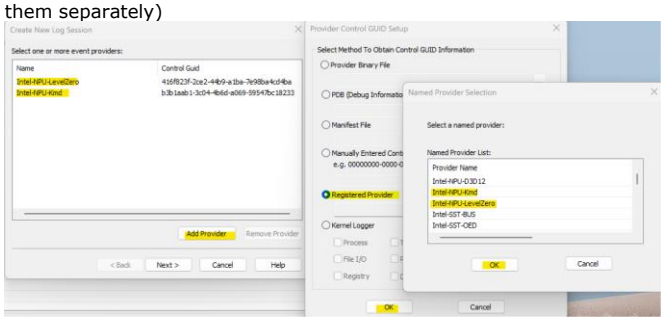


3.4 Trace View

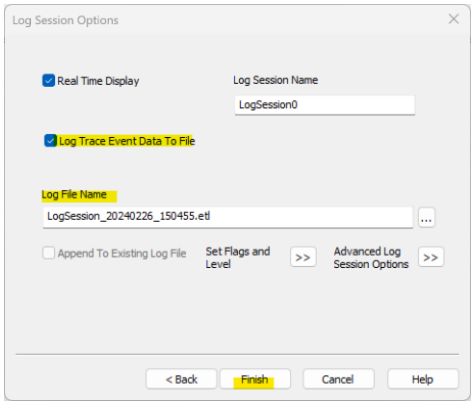
- Download and install Windows Driver Kit (WDK)
  The traceview.exe can be found at the path below.
  C:\Program Files (x86)\Windows Kits\10\Tools\10.0.22621.0\x64\traceview.exe
- Right click **traceview.exe** and select "Run as administrator". Selecting "File" -> "Create New Log Session".



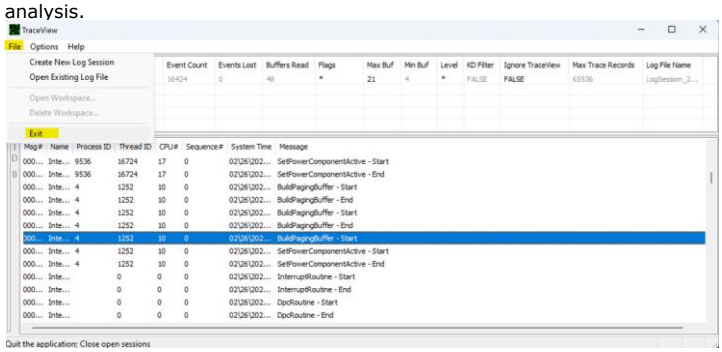
- In Registered Provider, add Intel-NPU-Kmd and Intel-NPU-LevelZero (Need to add them separately)

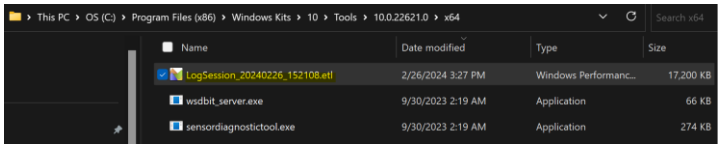


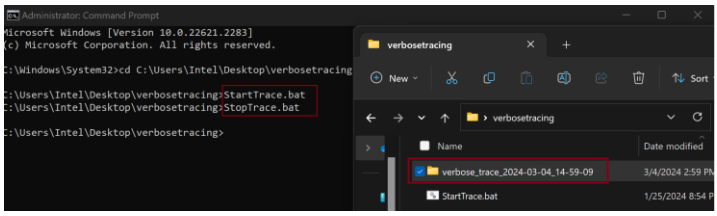
- Click Next and check "Log Trace Event Data To File" and then "Finish" to start the log session.



- Run your application/use case to reproduce the issue.
- Selecting "File"> "Exit" to stop the log session and share the etl trace to us for analysis.





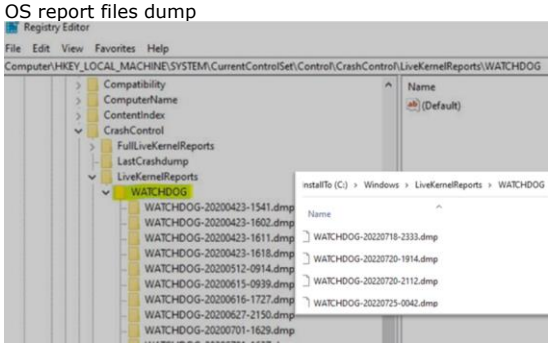


3.5 Live Kernel Dump

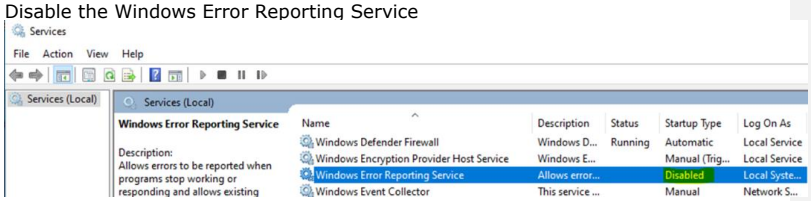
NPU driver also supports kernel live dump. The dump file is in C:\Windows\LiveKernelReports\WATCHDOG when TDR was triggered. Using Windbg to check who causes the TDR.

If dump was not observed

- Make sure sub reg keys under Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\CrashControl\LiveKernelReports\WATCHDOG are cleaned up, if it reaches 10, you won't see OS report files dump

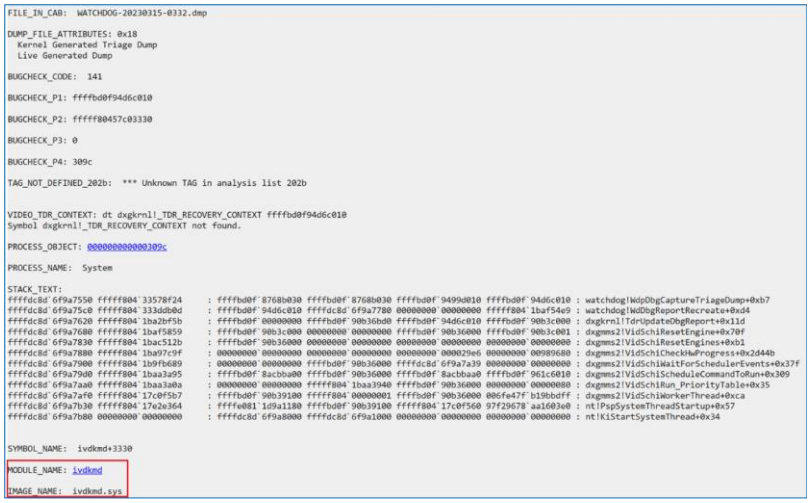


- Disable the Windows Error Reporting Service



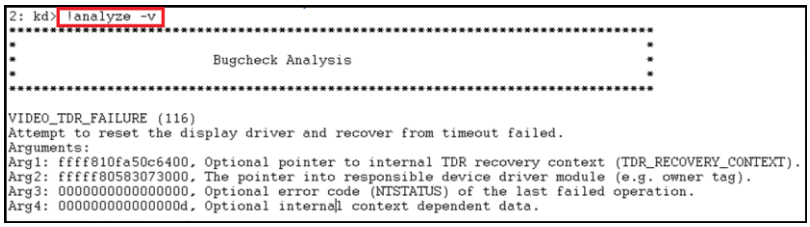
Below is an example, if you see npu_kmd.sys or ivdkmd.sys in the stack, please contact NPU team for further analysis.

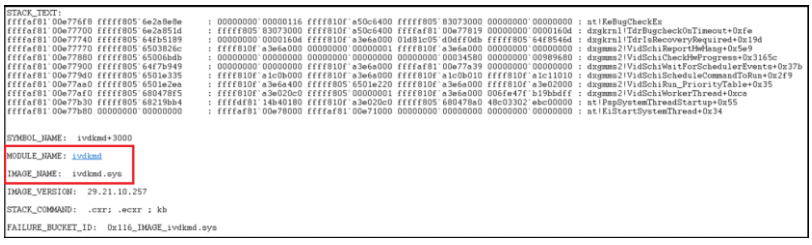
```
Loading User Symbols
Mini Kernel Dump does not contain unloaded driver list
For analysis of this file, run !analyze -v
watchdog!WdpDbgCaptureTriageDump+0xb7:
fffff804`3356396d 488b4628        mov     rax,qword ptr [rsi+28h] ds:002b:ffffbd0f`9499d038=??????
9: kd> !analyze -v
*******************************************************************************
*                                                                             *
*                        Bugcheck Analysis                                    *
*                                                                             *
*******************************************************************************

VIDEO_ENGINE_TIMEOUT_DETECTED (141)
One of the display engines failed to respond in timely fashion.
(This code can never be used for a real BugCheck.)
Arguments:
Arg1: ffffbd0f94d6c010, Optional pointer to internal TDR recovery context (TDR_RECOVERY_CONTEXT).
Arg2: fffff80457c03330, The pointer into responsible device driver module (e.g. owner tag).
Arg3: 0000000000000000, The secondary driver specific bucketing key.
Arg4: 000000000000309c, Optional internal context dependent data.
```



3.6 Memory Dump

If the system shows blackscreen or blue screen (BSOD), see if any memory dump created under C:\Windows\MEMORY.DMP. Using Windbg tool to check if NPU driver npu_kmd.sys or ivdkmd.sys in Windbg **!analyze –v**

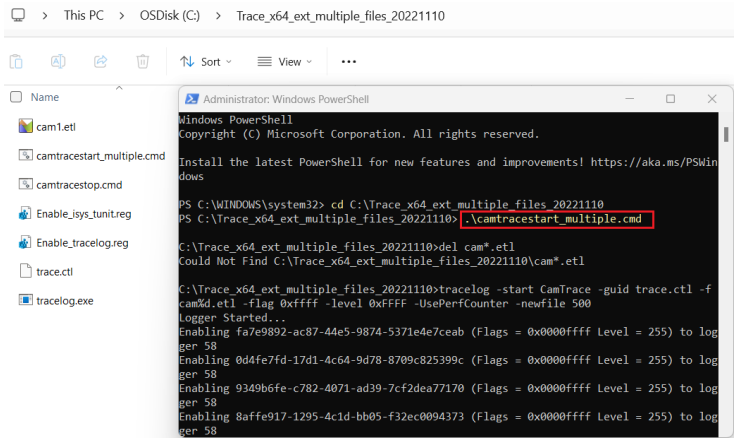




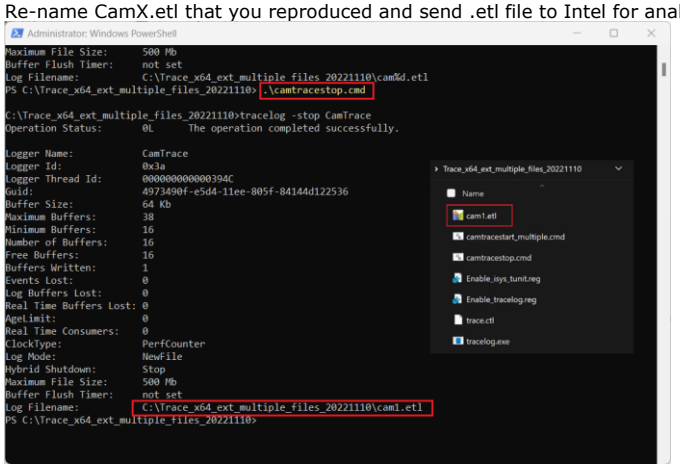
3.7 Mipi Camera Trace

Please get Mipi Camera Trace tool from Intel camera team and follow below steps to collect the log.

- Copy tool to your test device.
- Navigate to the tool folder with Windows PowerShell open with Administrator.
- Type "camtracestart_multiple.cmd" in the PowerShell and DO NOT close the PowerShell window.



- Start re-producing your issue.
- When the issue is reproduced, in previous PowerShell windows, type "camtracestop.cmd" to stop your logging.
- There'll be a newly generated CamX.etl in your tool folder (Cam1.etl will be the first log. If larger than 512MB, will create Cam2, Cam3, … and so on)
- Re-name CamX.etl that you reproduced and send .etl file to Intel for analysis.



4 Customer Bug Report Template

When reporting an issue to Intel or creating a sighting, please help provide the information below so that we can perform the first triage smoothly without wasting too much time in back-and-forth check.

- Issue description
- Is it a regression (Any pass conditions)
- Reproduced steps
- Reproduced video or screenshot
- Expected result
- Actual result
- Recovery steps
- Related logs
- System configuration
  - Platform
  - OS version
  - NPU driver version
  - MEP version
  - Camera driver (IPU/MIPI driver version), if use USB camera, please tell us if you use Microsoft inbox driver or 3rd party driver
  - Others: for example: Microsoft Camera app version

5 Trouble Shooting

The following are some common issues (including expected behaviors) for reference, please take a look before reporting issues to Intel.

5.1 No MEP options in system tray or Settings

Commented [ZX1]: Suggest to use WseEnablingStatus to check the MEP optin status

Commented [PH2R1]: Thanks Zhan, added. Since MSFT only provide this tool for seed projects so far, I also keep the manual check method.

Please get **WseEnablingStatus** tool from Microsoft to check MEP opt-in status.

You will see MEP opt-in camera and Windows Studio Effect version when you run this tool if you opt-in MEP successfully.

```
C:\WseEnablingStatus>WseEnablingStatus.exe
System Name: LAPTOP-CD63QRRN
System OS Info: Windows 10 Pro (26085.1)
Opt-In Camera Status: True
Opt-In Camera FriendlyName: 9MP Camera
Opt-In Camera Hardware ID: USB\VID_0408&PID_546E&REV_0006&MI_00
Opt-In Camera Driver: 10.0.26085.1
Windows Studio Effects Camera: 1.0.38.0
```

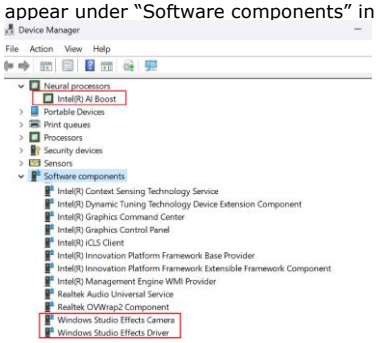
The tool will point out the problem if you don't opt-in MEP correctly.
For example, from the message below, we know MEP was not correctly deployed, please make sure NPU extension and MEP package was installed successfully.

```
F:\WseEnablingStatus>WseEnablingStatus.exe
can not find 'Windows Studio Effects Camera' in device manager, extension .inf for MEP camera was not correctly deployed
```

```
F:\WseEnablingStatus>WseEnablingStatus.exe
can not find Opt-in camera instance in registry, there is no 'FSMEnableMsEffects' key in registry
```

If you don't have this tool, please follow the steps below to check it manually.

- Check if NPU and MEP are installed correctly.
  To see if "Windows Studio Effects Driver" and "Windows Studio Effects Camera" appear under "Software components" in device manager.

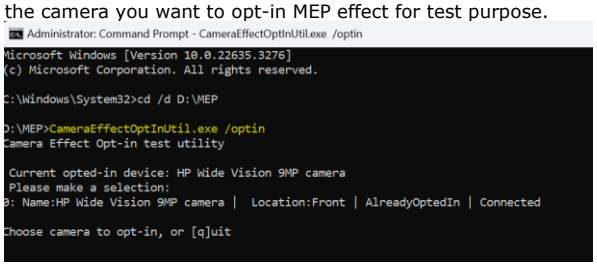


If you cannot find MEP components in device manager, please make sure NPU and NPU extension driver was installed correctly, NPU extension driver will create MEP software device node for installing Microsoft MEP package.

- Change NPU subsystem ID in system BIOS
- Modify NPU extension inf (replace your own GUID and subsystem ID for your project being targeted for MEP opt-in)

For details, please refer to RDC document #776500 NPU MEP opt-in guide.

- Check if the camera was opt-in correctly.
  Please refer to Microsoft document "Windows 11 SV2 - AI Experiences on NPU - OEM Enablement Guide" to know how to opt-in of MEP Camera AI effects. You can also get "CameraEffectOptInUtil.exe" tool from Microsoft and use /optin to select the camera you want to opt-in MEP effect for test purpose.

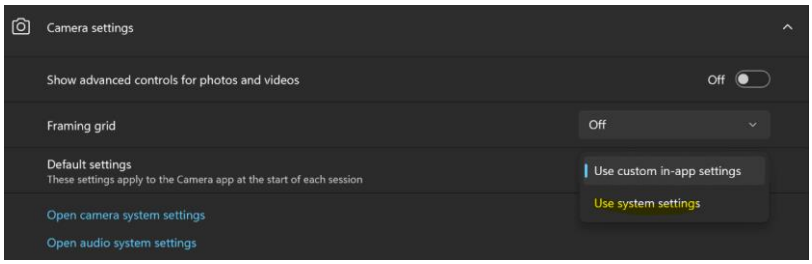


5.2 Super resolution does not work normally

Super resolution only works under AC mode when Auto Framing is enabled. It's expected to see super resolution doesn't work under DC mode.

5.3 MEP effect on/off status is not expected

Check in the camera application to see if you follow system global settings or in-app settings. If you select use system settings, it's expected to see the effects follow global settings instead of in-app settings.



5.4 The effects are not available in photo mode of camera app

This is expected. It is due to the fact that when an app selects HighQuaityPhoto profile, the effects are disabled by design. Hence it is observable easily on MIPI camera with this profile advertised. This avoids the mismatch between effects applied on preview and absent on actual photo.

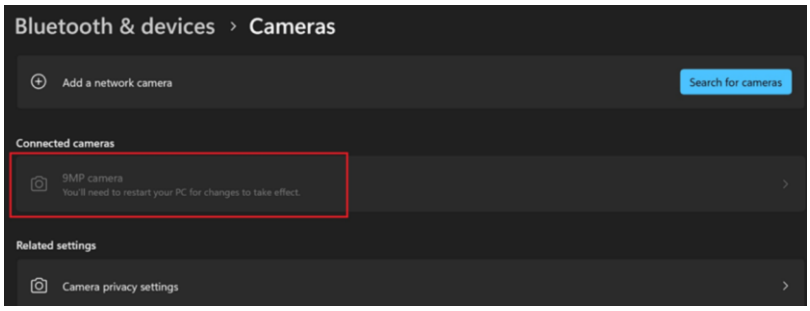
5.5 MEP effects in settings page will be impacted after launch Camera app in photo mode

This is expected behavior. The settings app does not re-initialize the camera stream and will not override other app's changes to the stream, so the preview in settings can be affected by other apps using the camera.

Camera app enables photo mode which disables the MEP effects, whether settings app is running or not. Camera stream is not re-initialized when settings app is opened, or when camera app is closed while setting app is opened. The resulting stream then would be configured for photo mode and not show any MEP effects. When all open apps are closed, this stream is released, and when settings app is next opened a new stream is created which has not been set to photo mode, allowing MEP effects.

Camera shows gray out after disabling it in camera settings, need to restart the system to enable it back

It’s not an issue. Usually the request of "restart your PC" comes from kernel mode driver, and the reason is some handles opened by user mode components not released in time (might be frameserver/frameservermonitor). This is somewhat common, even with devices without MEP installed. MEP might slow down frameserver's response time that make this hit rate higher, but it's not considered as a bug.

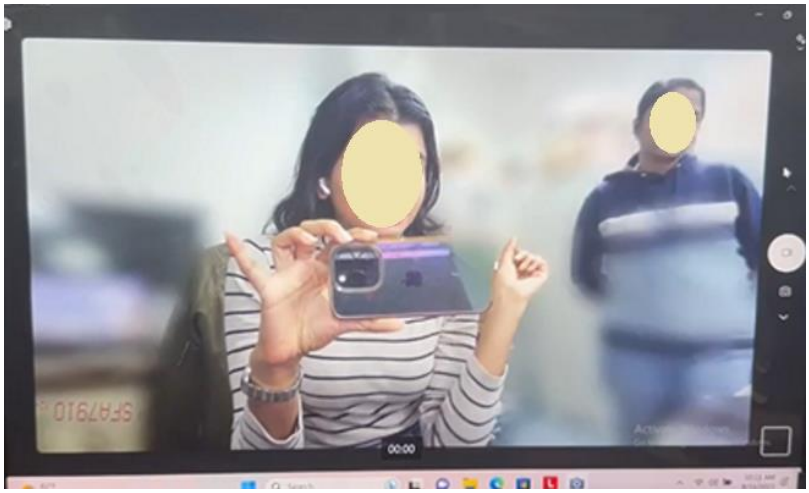


5.6 Eye Contact doesn't work in portrait mode

Eye Contact v1 supports landscape orientation devices only, Eye Contact v2 provides support for multiple device orientations including portrait mode.

5.7 Image of secondary user on camera preview glitches when background effects enabled

The background blur feature works for 2 people, if they are at equal distance. If the relative distance of the 2 persons is at the margin of algorithm's decision, the flickering looks as expected and by-design.



5.8 Automatic framing initial time is longer while Background effects is enabled

The cause for this issue is known- when background blur is enabled, tracked faces are used as autoframing input. When blur is disabled, non-tracked face detections are used as input. Tracked faces are an amalgam of recent face detection rectangles and therefore have different spatial and stability properties. Specifically, tracked rectangles tend to be larger and result in a looser autoframed compositions (and indirectly on whether/when zoom occurs near thresholds)

Later releases of Auto Framing are expected to have more consistent compositions between modes, but the changes to enable that consistency were deemed too risky to adopt for recent MTL releases.