

**AMD**  
EPYC

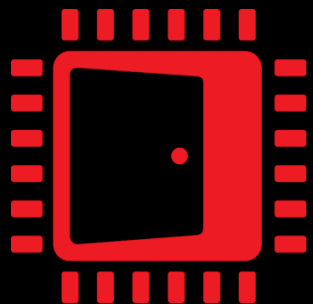
**AMD**  
RYZEN

**AMD**  
RADEON

# OPTIMIZING GAME PERFORMANCE WITH THE RADEON™ DEVELOPER TOOL SUITE

CHRIS HESIK

CAN ALPER



**AMD**  
GPUOpen

**AMD**  
together we advance\_

# ABOUT THIS PRESENTATION

## Part 1: Overview of the Radeon™ Developer Tool Suite

- What is in it
- What's new since GDC 2022
- Dive into new features and improvements
- RADV & Steam Deck support

## Part 2: Collaboration with external tool developers


- Microsoft® PIX on Windows
- RenderDoc
- GFXReconstruct

# AMD

## RADEON

Developer Tool Suite

**AMD **  
**RADEON**  
GPU Profiler

**AMD **  
**RADEON**  
Memory Visualizer

**AMD **  
**RADEON**  
Raytracing Analyzer

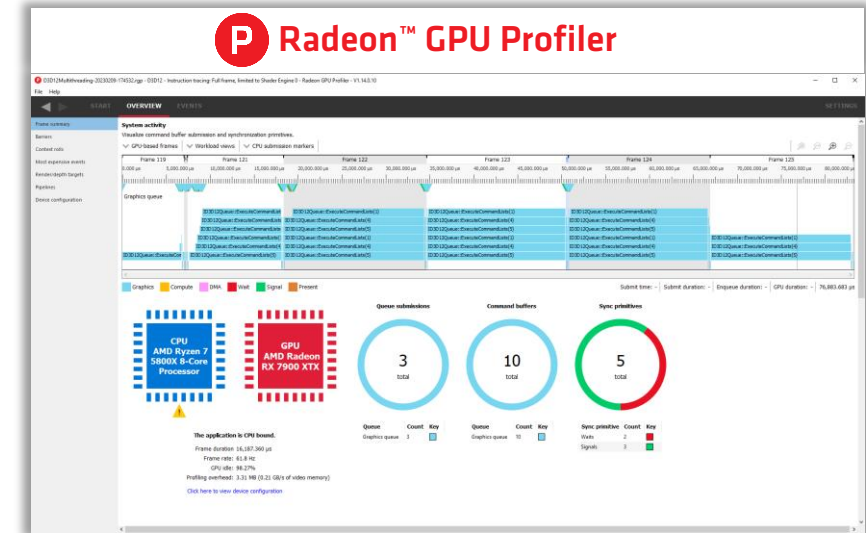
**AMD **  
**RADEON**  
GPU Analyzer

**AMD **  
**RADEON**  
Developer Panel

# RADEON DEVELOPER TOOL SUITE - OVERVIEW

## Radeon GPU Profiler (RGP)

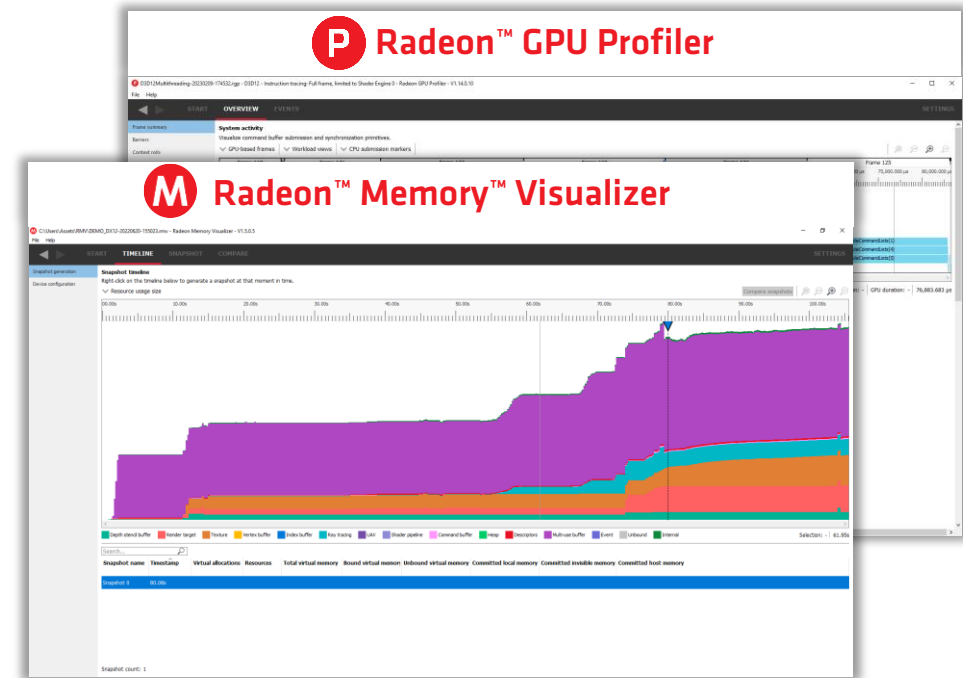
- Get a bird's eye view of how your command buffers got submitted to each GPU queue, including synchronization between the queues
- Understand how your wavefronts were pushed through the GPU, and correlate wavefronts to the API-level Draws/Dispatches which launched them
- Learn how your frame utilizes the various GPU memory caches
- Quickly and easily find hotspots in your shaders using the instruction timing view
- Identify parts of the frame using ray tracing and gain insights into the different shaders that make up a ray tracing pipeline, including which parts may be the bottleneck
- Find out which barriers flushed caches, caused a synchronization point or even ran their own, internal shaders
- Discover which events and which pipelines are the most expensive.
- Get insight into the render targets used by your frame



# RADEON DEVELOPER TOOL SUITE - OVERVIEW

## Radeon Memory Visualizer (RMV)

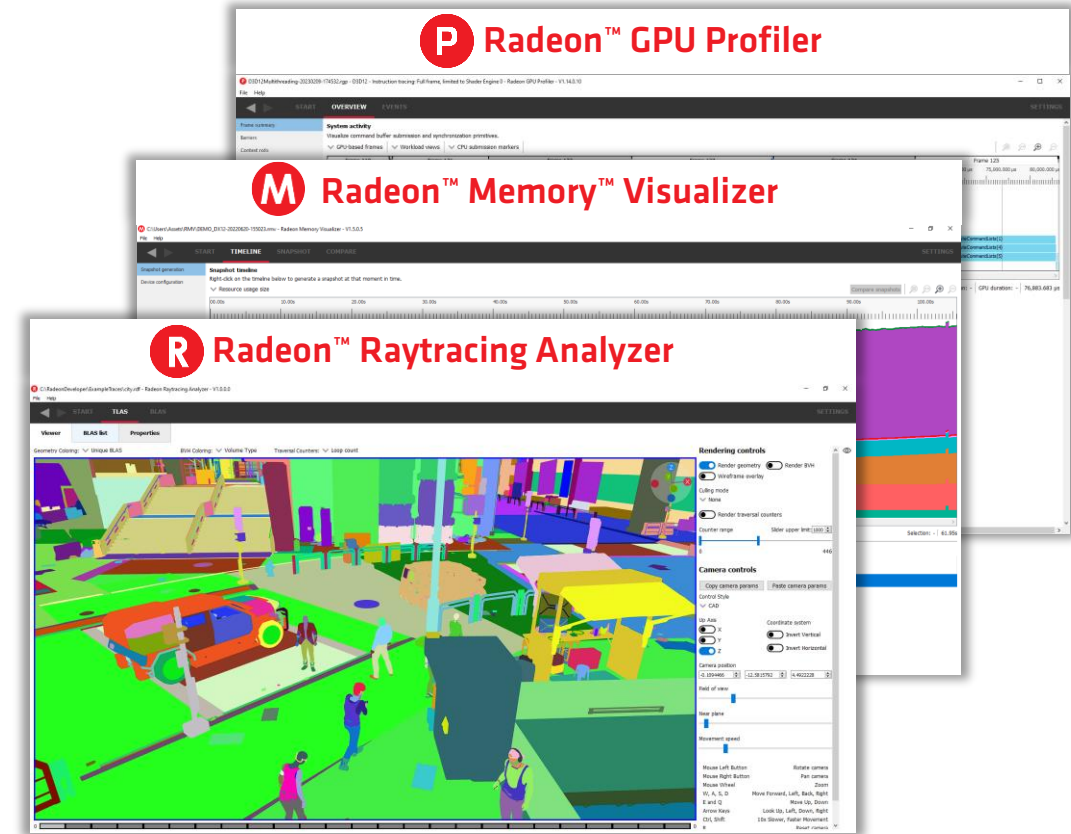
- Tracks the GPU memory usage of a running application
- Create detailed snapshots of memory state
- Visualize heap over-subscription
- Quickly find resources which are not in the optimal heaps
- Understand which resources require the most memory, which heaps they are in, and how your heaps are sub-allocated
- Identify fragmentation and understand how you are managing memory in each heap
- Understand the balance between dedicated and placed resources
- Easily find memory leaks in your application by comparing snapshots
- Use the DirectX® 12 or Vulkan® API to customize the names of your resources then search and view the details in the Resource list



# RADEON DEVELOPER TOOL SUITE - OVERVIEW

## Radeon Raytracing Analyzer (RRA)

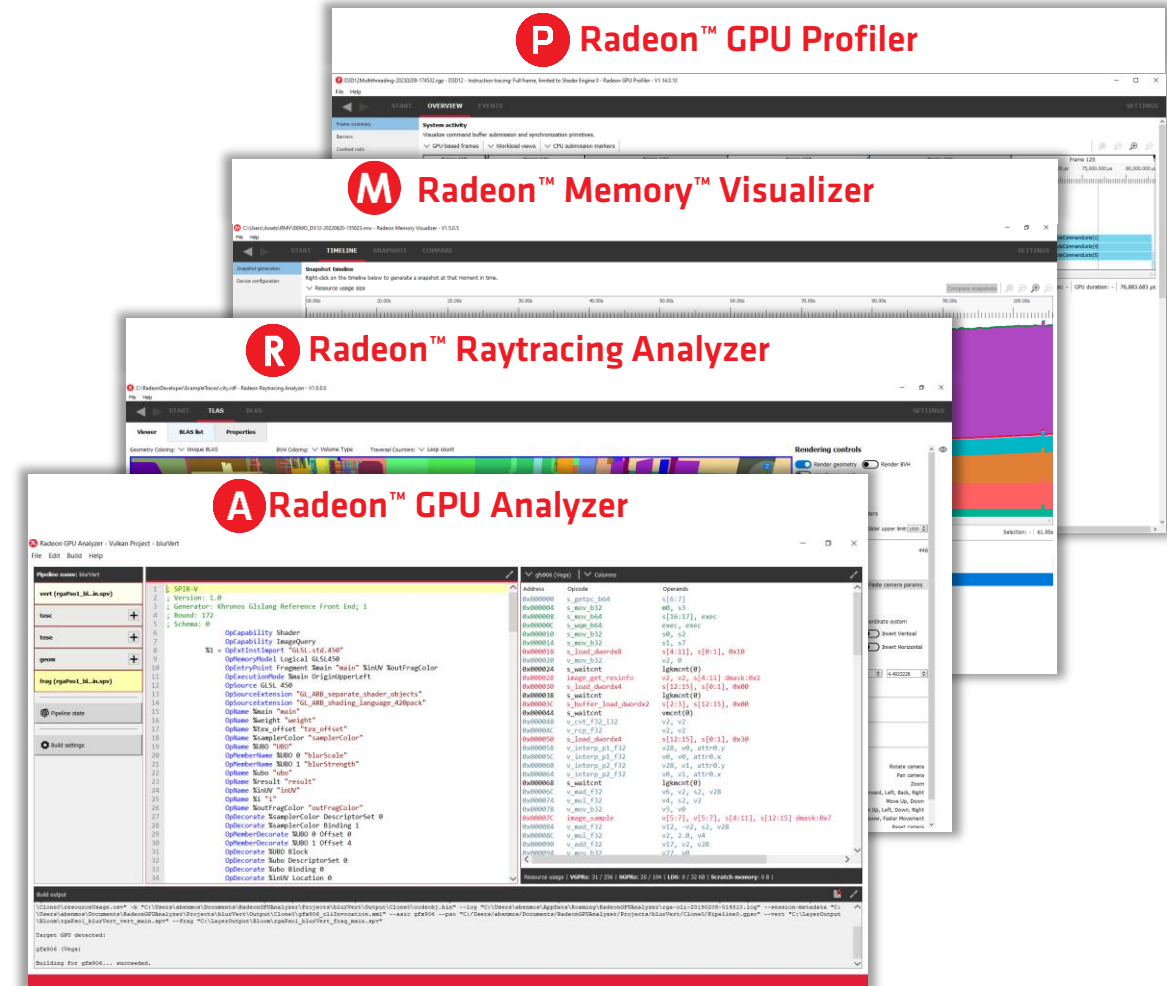
- Visually inspect your acceleration structures (TLAS and BLAS), select coloring modes to highlight areas of interest
- Use traversal counter rendering mode to find hotspots for loop count, instance overlaps and more
- View and validate primitives and attributes that were passed to the driver by your application
- High-level statistics for every acceleration structure in your scene, including things such as memory usage and triangle counts



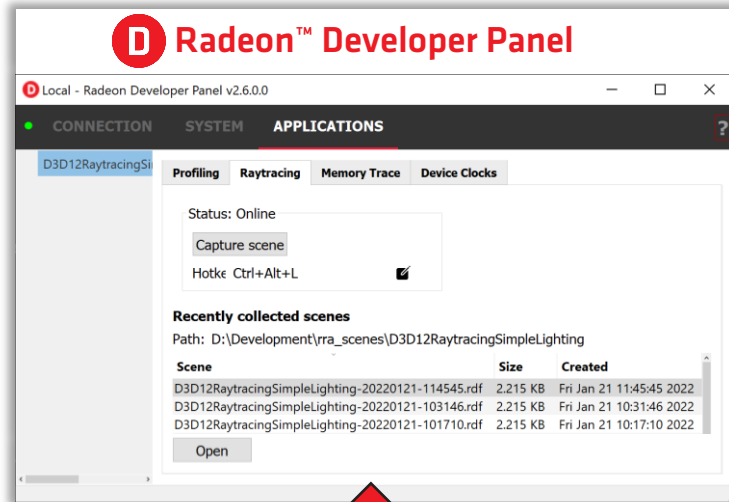
# RADEON DEVELOPER TOOL SUITE - OVERVIEW

## Radeon GPU Analyzer (RGA)

- Compile and analyze shaders and kernels for AMD GPUs
- Inspect disassembly and static hardware resource usage
- Identify areas with high VGPR pressure
- Independent of the physically installed GPU
- CLI support for DirectX®12, DXR, DirectX®11, Vulkan®, SPIR-V™, OpenGL®, and OpenCL™
- GUI support for Vulkan and OpenCL



# RADEON DEVELOPER TOOL SUITE - OVERVIEW



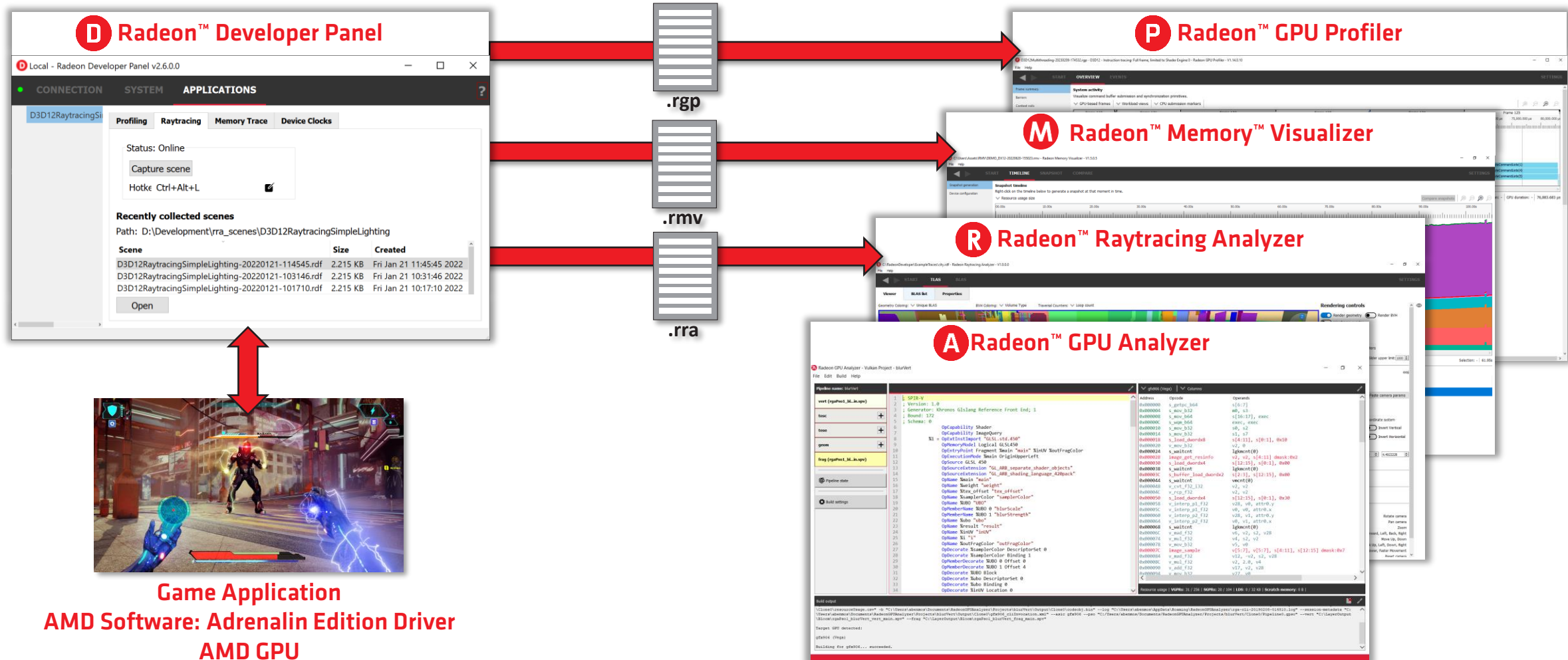
**Game Application**  
**AMD Software: Adrenalin Edition Driver**  
**AMD GPU**

## Radeon Developer Panel (RDP)

- RDP connects to the Radeon™ Adrenalin driver to enable developer mode
- Developer mode generates:
  - Correlated hardware tracing, driver and application timing data for the Radeon™ GPU Profiler (RGP)
  - Memory usage data for the Radeon™ Memory Visualizer (RMV)
  - Acceleration structure information for RRA
- Supports local and remote data gathering
- RDP supports default and custom workflows to simplify data gathering from your applications



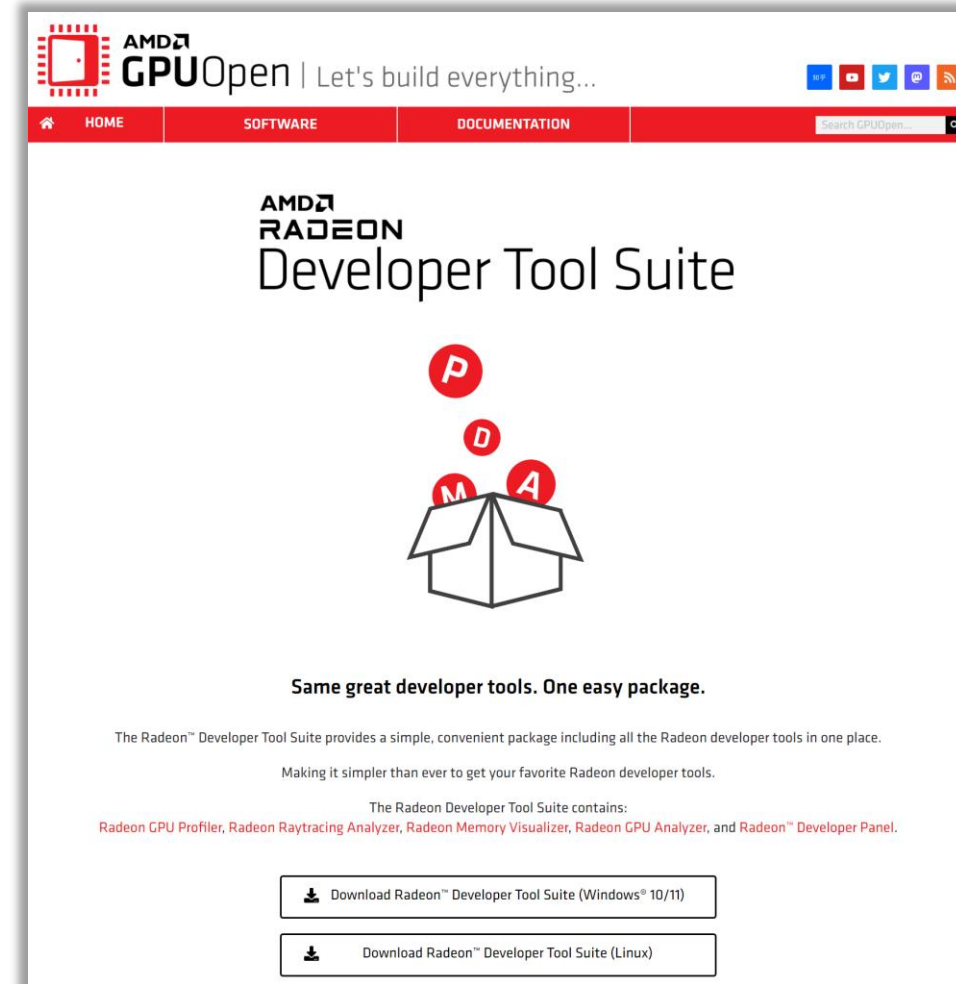
# RADEON DEVELOPER TOOL SUITE - OVERVIEW



## RGA is standalone and does not use RDP

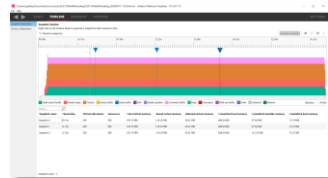
# RADEON DEVELOPER TOOL SUITE

- Download: [gpuopen.com/tools](https://gpuopen.com/tools)
- ZIP file containing
  - Radeon GPU Profiler
  - Radeon Memory Visualizer
  - Radeon Raytracing Analyzer
  - Radeon GPU Analyzer
  - Radeon Developer Panel
- No installer needed
- Videos: [gpuopen.com/videos](https://gpuopen.com/videos)



# WHAT'S NEW?

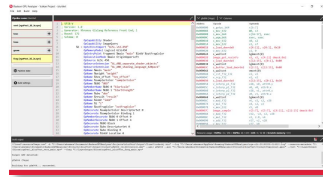
GDC 2022



**RMV 1.4**  
Smart Access Memory support



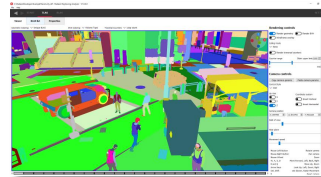
**RGP 1.13**  
Raytracing perf counters  
Inline raytracing support



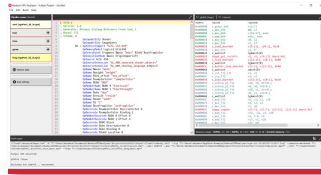
**RGA 2.6.1**  
Support for RX 680M  
VS2022 compilation  
DXR/DX12 offline mode



**RMV 1.4.1**  
Ubuntu 20.04



**RRA 1.0**  
BVH Viewer



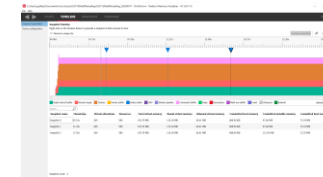
**RGA 2.6.2**  
OpenGL offline mode



**RGP 1.13.1**  
Bug fixes



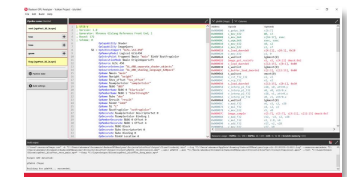
**RRA 1.1**  
OpenSource code,  
rebraiding, triangle splitting,  
instance masks, traversal  
counter histogram



**RMV 1.5**  
Raytracing support



**RGP 1.14**  
HIP support, OpenCL/HIP  
instruction timing,  
ExecuteIndirect Raytracing  
support



**RGA 2.7**  
DirectX 11 offline mode  
VGPR Pressure UI

**RDNA3 support**

March

April

May

June

July

Aug

Sept

Oct

Nov

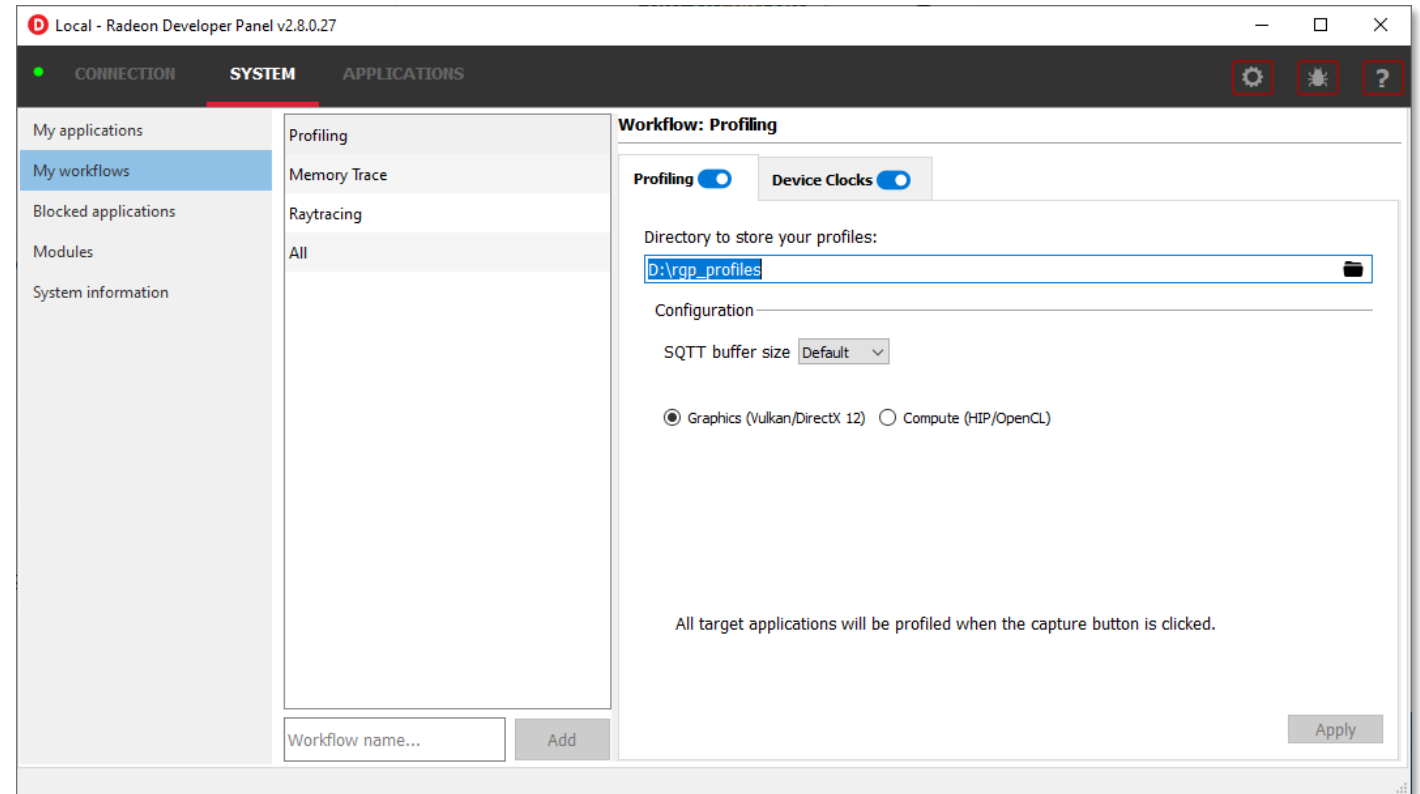
Dec

GDC 2023

# AMD RADEON Developer Panel

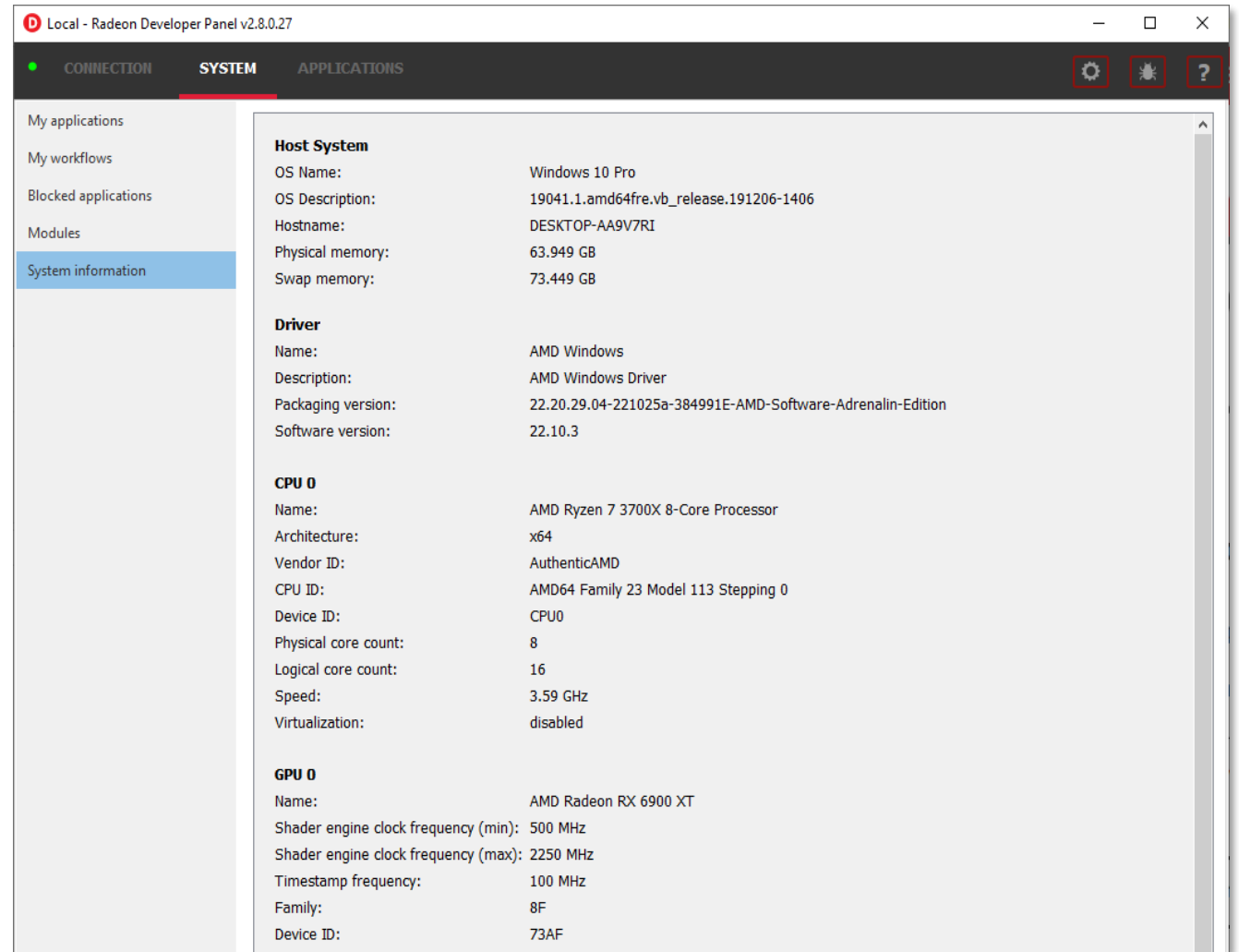
# RADEON DEVELOPER PANEL 2.8 UPDATE

- Workflow UI improvements



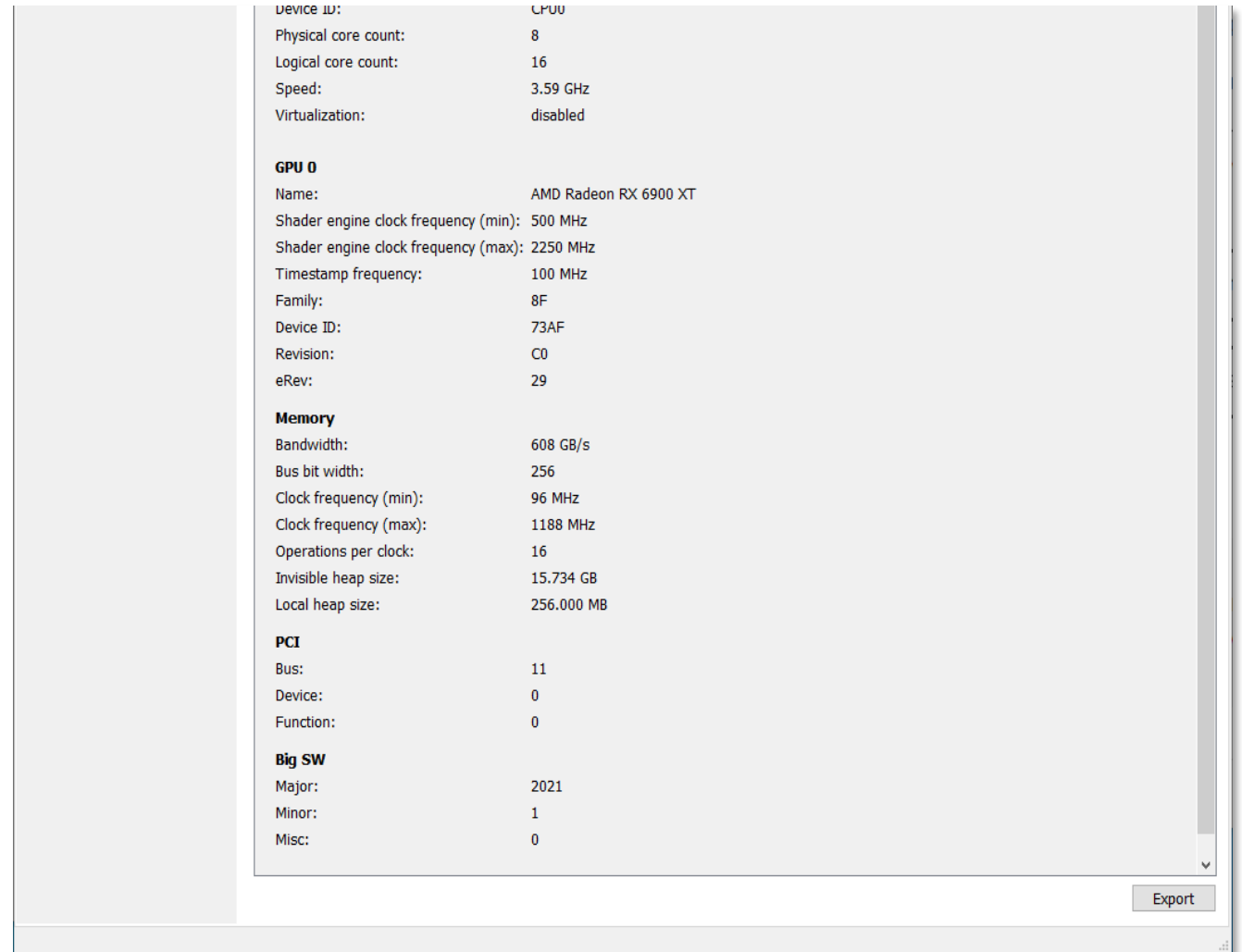
# RADEON DEVELOPER PANEL 2.8 UPDATE

- Workflow UI improvements
- New - System Information



# RADEON DEVELOPER PANEL 2.8 UPDATE

- Workflow UI improvements
- New - System Information
  - With Export (JSON)
- RDNA3 Support
- Stability

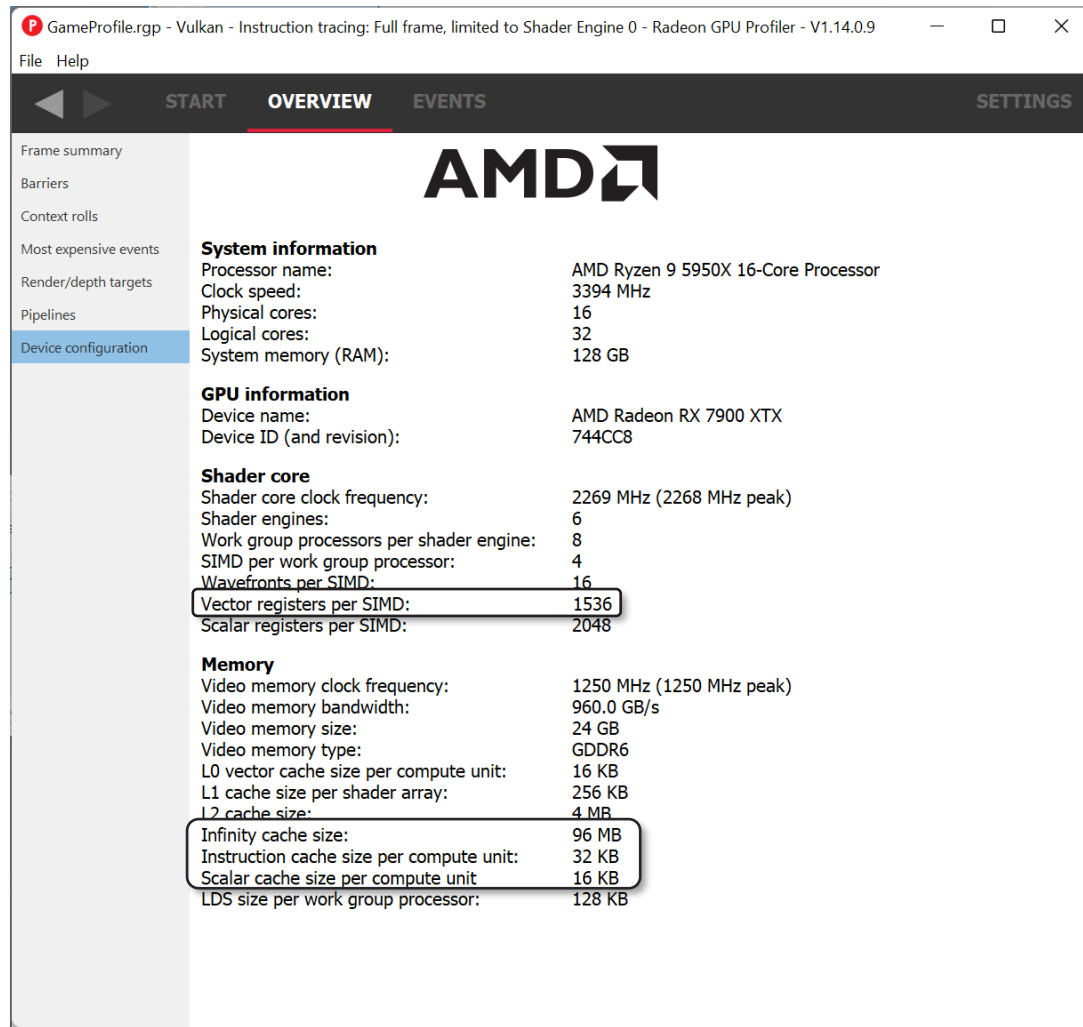


# AMD RADEON GPU Profiler

[https://github.com/GPUOpen-Tools/radeon\\_gpu\\_profiler](https://github.com/GPUOpen-Tools/radeon_gpu_profiler)



# RADEON GPU PROFILER 1.14 – RDNA 3 SUPPORT



- Additional cache size info in the Device Configuration pane
- More Vector registers per SIMD

# RADEON GPU PROFILER 1.14 – RDNA 3 SUPPORT

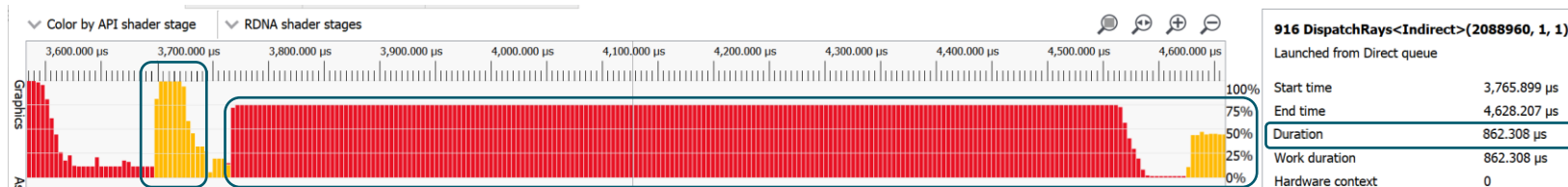
## AMD Radeon RX 6800

AMD Ryzen 7 5800X 8-Core Processor, 3793 MHz, 64GB RAM, Microsoft Windows 11 Pro



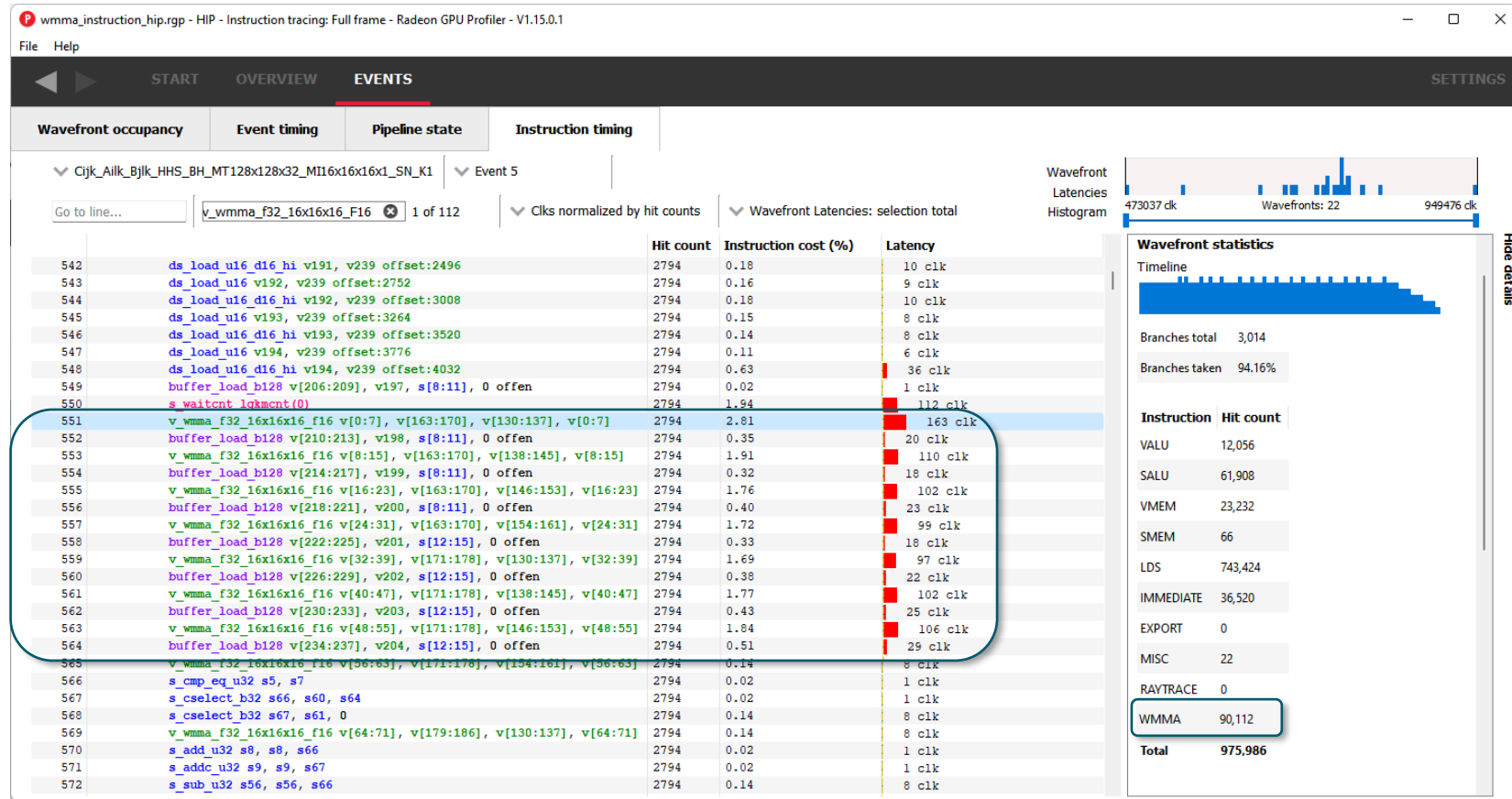
## AMD RX 7900 XTX

AMD Ryzen 7 5800X 8-Core Processor, 3793 MHz, 64GB RAM, Microsoft Windows 11 Pro



- Additional cache size info in the Device Configuration pane
- More Vector registers per SIMD
- Improved Wavefront occupancy for some workloads, including ray tracing events that use Indirect shader pipelines

# RADEON GPU PROFILER 1.14 – RDNA 3 SUPPORT



- Additional cache size info in the Device Configuration pane
- More Vector registers on RDNA 3
- Improved Wavefront occupancy for some workloads, including ray tracing events that use Indirect shader pipelines
- WMMMA instructions

# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

- RGP can identify parts of the frame which perform inline raytracing
  - Pipelines
  - Events



# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

- RGP can identify parts of the frame which perform inline raytracing
  - Pipelines

Frame summary

Barriers

Context rolls

Most expensive events

Render/depth targets

Pipelines

Device configuration

Pipeline summary

All detected pipeline configurations

VS

HS

DS

GS

PS

CS

RT

158 pipelines

VS

HS

DS

GS

PS

CS

RT

66 pipelines

VS

HS

DS

GS

PS

CS

RT

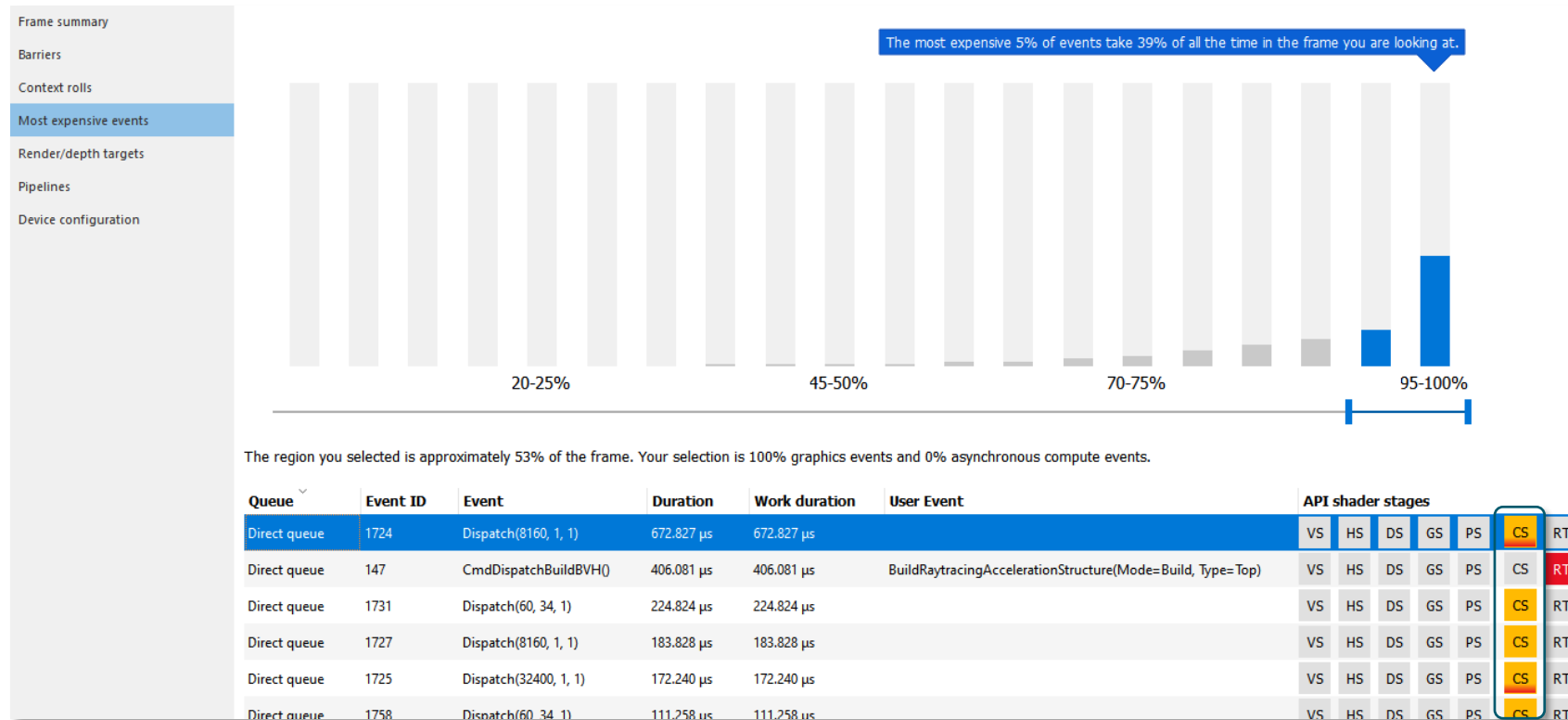
1 pipeline

Pipelines

Bucket ID	Hash (API PSO + API shader)	Duration	Event Count	Avg event duration	Occupancy	VGPRs	SGPRs	Scratch mem	Wave mode	Stages		
> 214	0xCC879BC2012076A7	672.827 μs [4.05%]	1 [0.05%]	672.827 μs	16 - 16	68 - 68	42 - 42	No		VS HS DS GS PS	CS	RT
> 86	0x53BE122CEF6FB004	231.513 μs [1.39%]	47 [2.52%]	18.121 μs	16 - 16	44 - 60	80 - 86	No		VS HS DS GS PS	CS	RT
> 219	0x0103BAAFDDED89A9	224.824 μs [1.35%]	1 [0.05%]	224.824 μs	16 - 16	49 - 49	80 - 80	No		VS HS DS GS PS	CS	RT
> 88	0x241BAF7E975C2995	220.115 μs [1.33%]	107 [5.74%]	20.707 μs	16 - 16	41 - 44	70 - 80	No		VS HS DS GS PS	CS	RT
> 87	0xD86383187F807978	196.500 μs [1.18%]	42 [2.25%]	19.496 μs	16 - 16	38 - 44	66 - 80	No		VS HS DS GS PS	CS	RT
> 89	0xF8570C5FAA57EB25	188.968 μs [1.14%]	46 [2.47%]	39.009 μs	16 - 16	44 - 54	80 - 102	No		VS HS DS GS PS	CS	RT
> 216	0xA09B0F2620310864	183.828 μs [1.11%]	1 [0.05%]	183.828 μs	16 - 16	57 - 57	92 - 92	No		VS HS DS GS PS	CS	RT
> 215	0xA23F9C83ED74A372	172.240 μs [1.04%]	1 [0.05%]	172.240 μs	16 - 16	68 - 68	42 - 42	No		VS HS DS GS PS	CS	RT

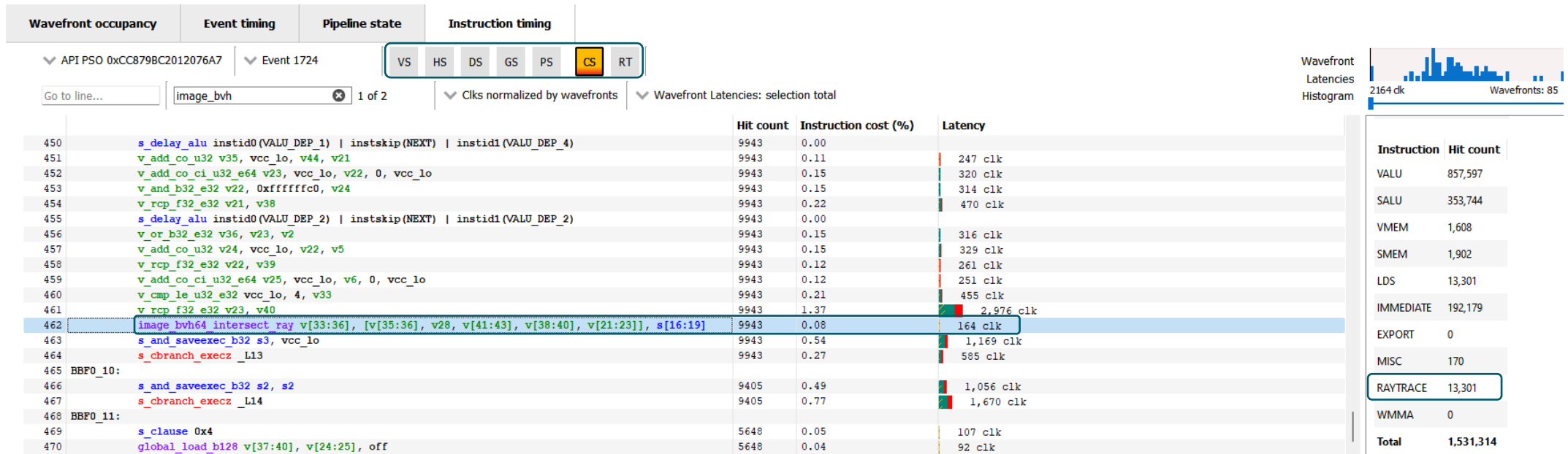
# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

- RGP can identify parts of the frame which perform inline raytracing
  - Pipelines
  - Events



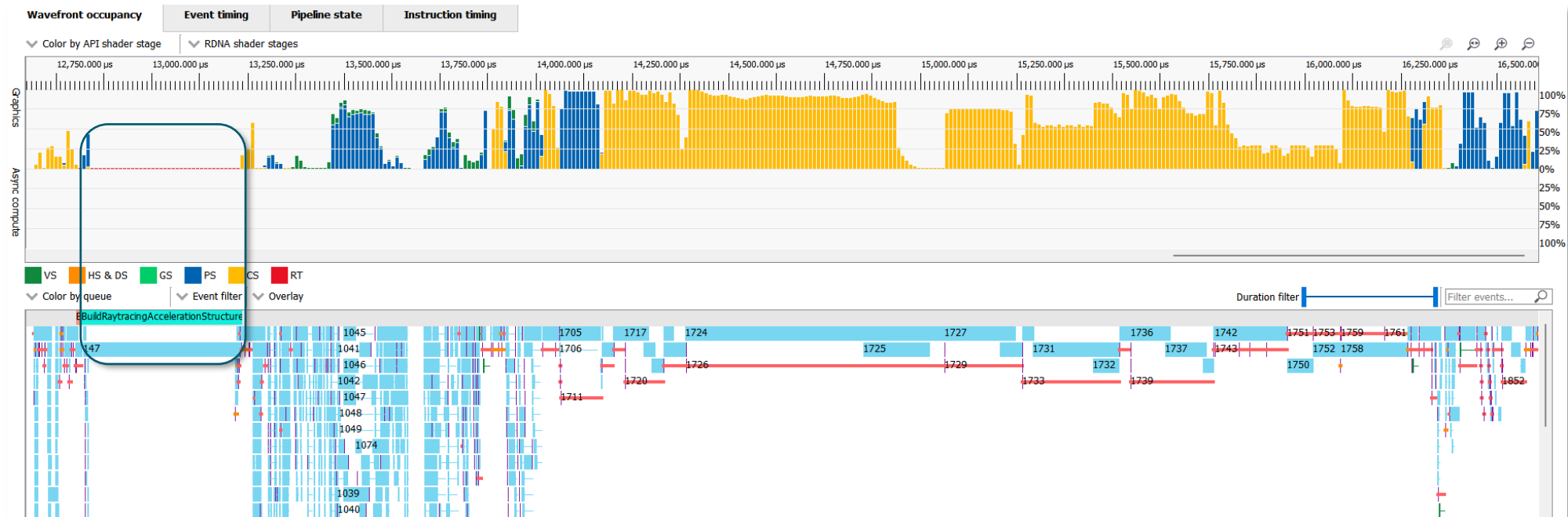
# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

- RGP can identify parts of the frame which perform inline raytracing
  - Pipelines
  - Events



# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

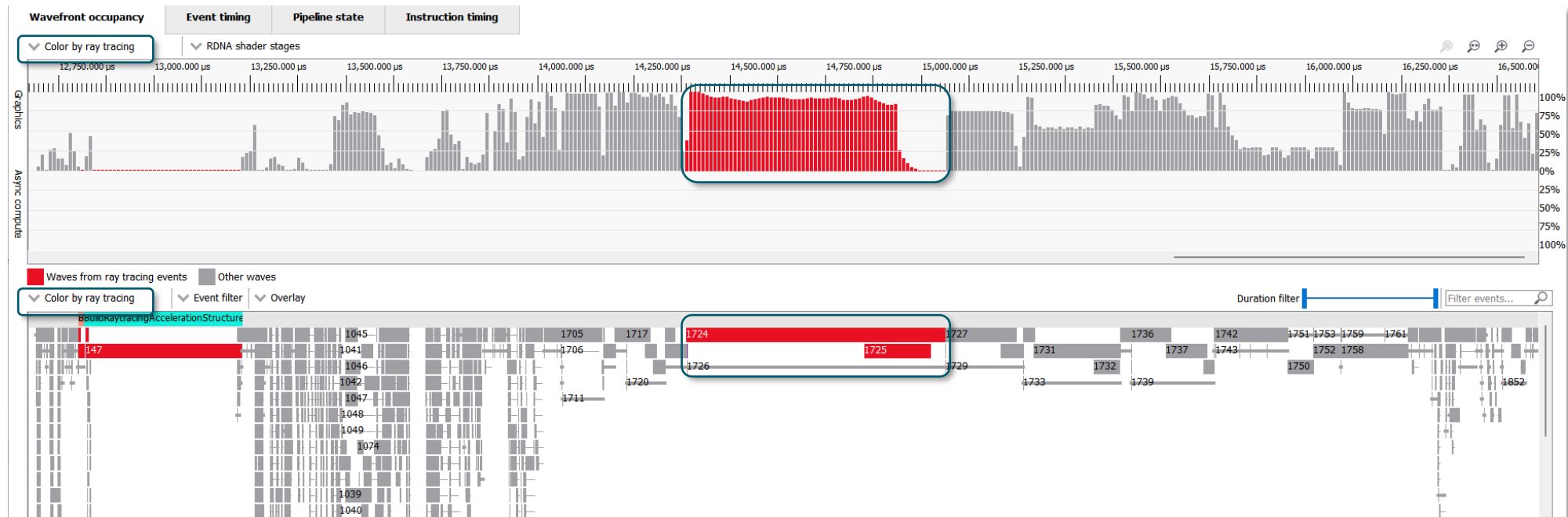
- RGP can identify parts of the frame which perform inline raytracing
  - Pipelines
  - Events
  - Wavefronts





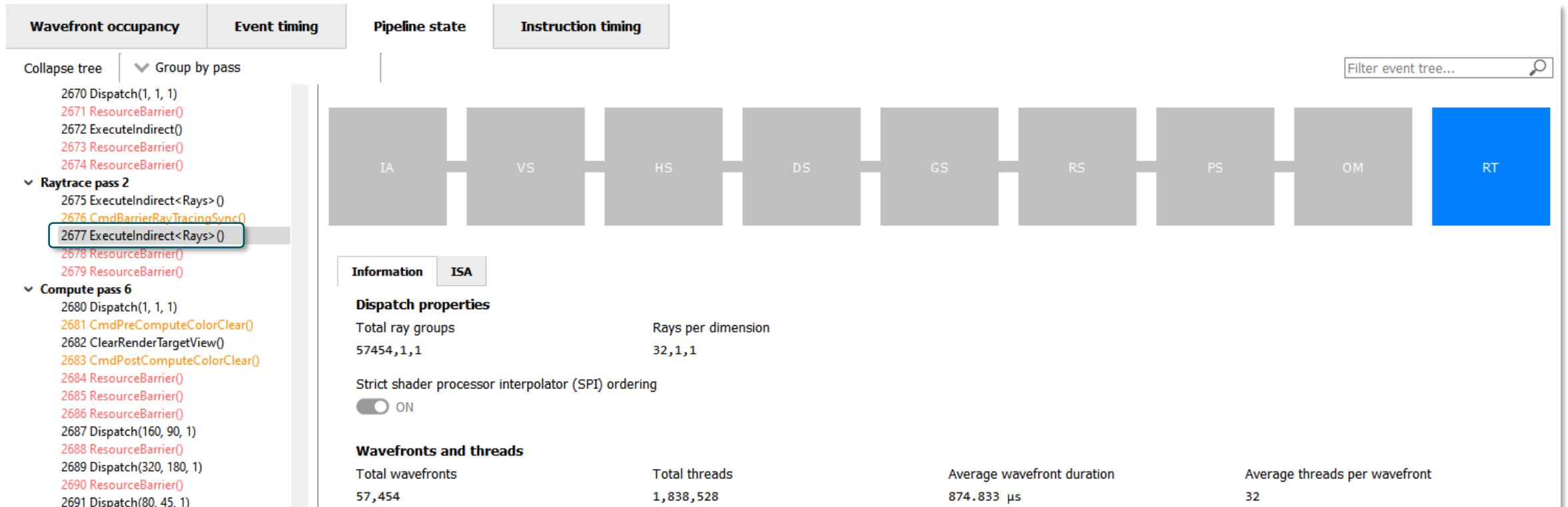
# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

- RGP can identify parts of the frame which perform inline raytracing
  - Pipelines
  - Events
  - Wavefronts



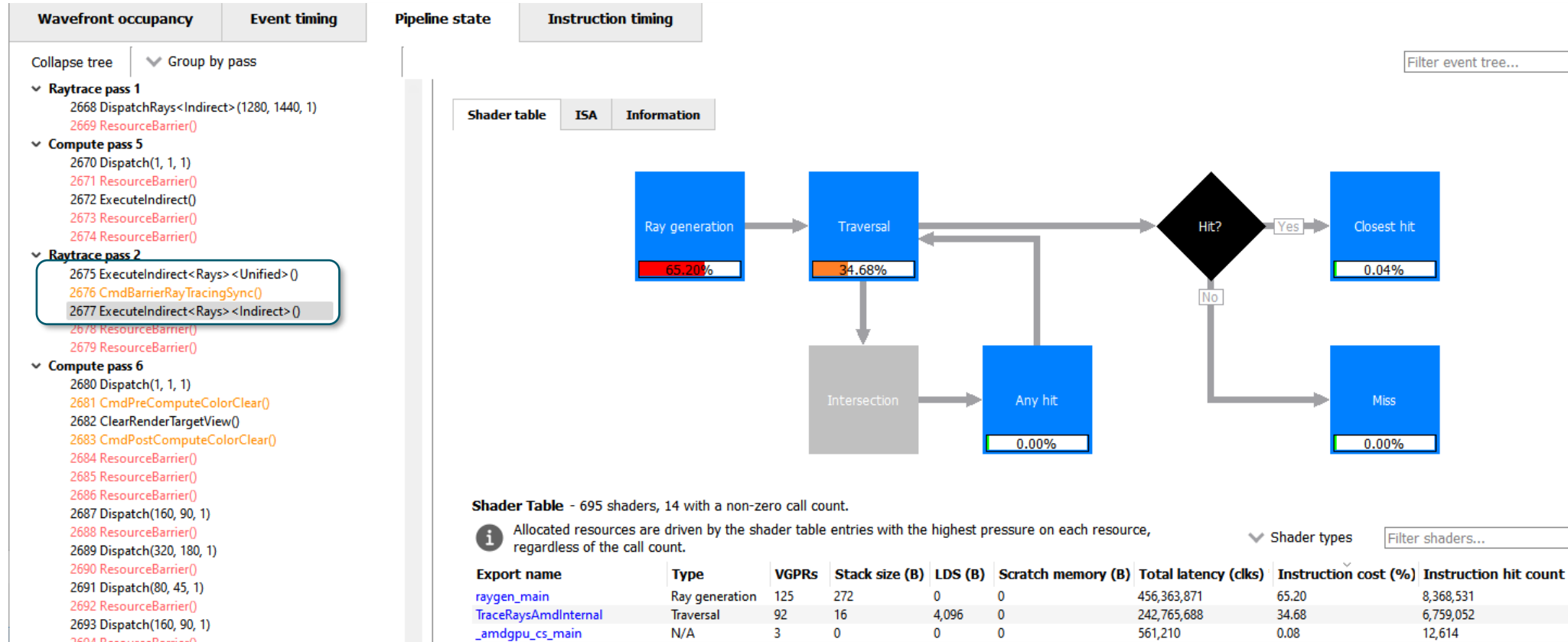
# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

- Support for ExecuteIndirect calls that perform ray tracing

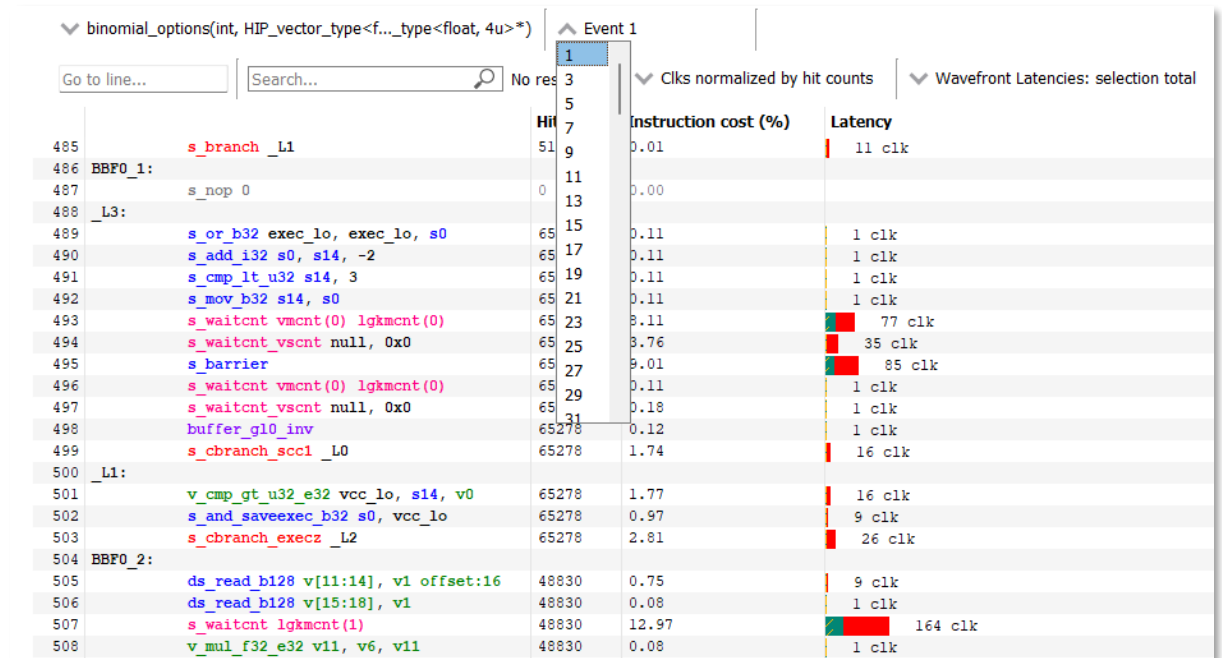
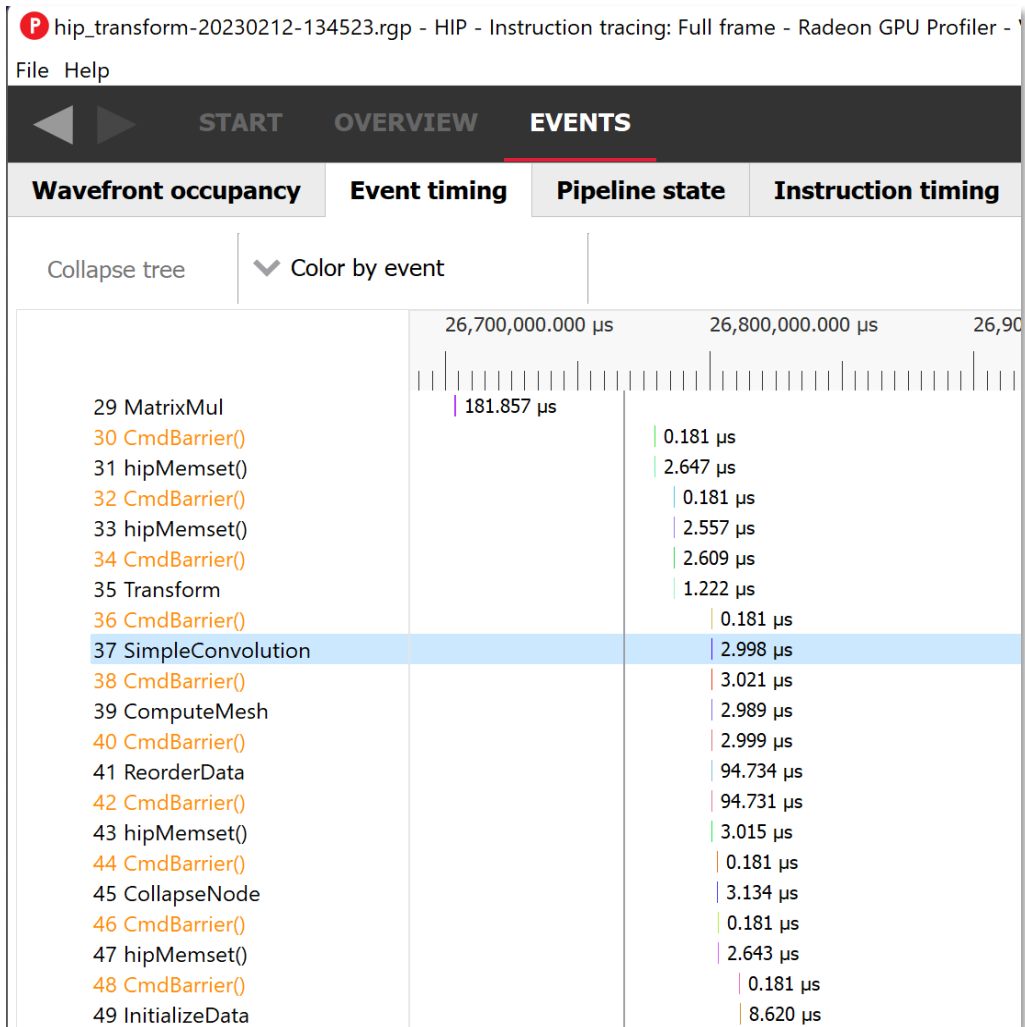


# RADEON GPU PROFILER 1.14 - RAY TRACING UPDATES

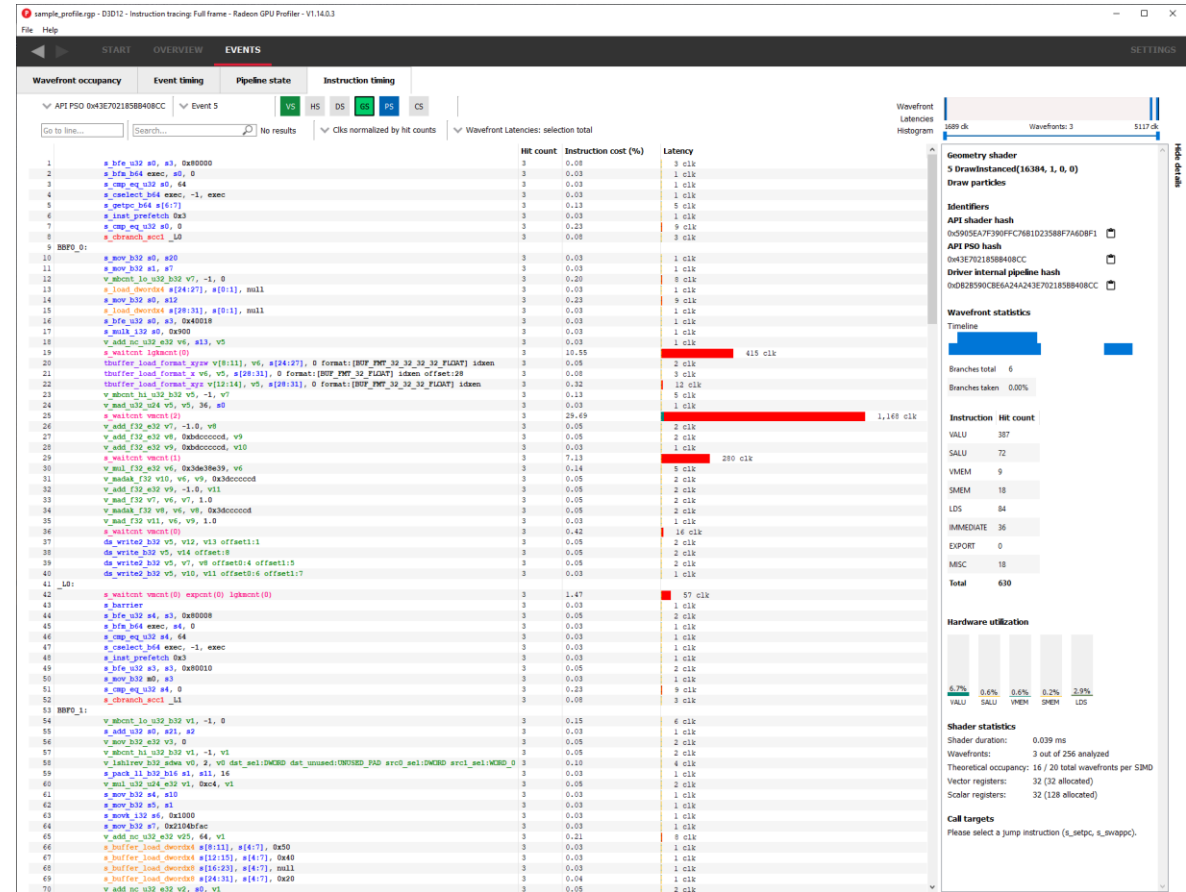
- Support for ExecuteIndirect calls that perform ray tracing



# RADEON GPU PROFILER 1.14 - HIP/OPENCL SUPPORT



- Motivation – differences in current ISA views



# AMD

# RADEON

## Raytracing Analyzer



### Detailed video of RRA

"Introduction to Raytracing with the Radeon Developer Tools Suite"

[gpuopen.com/videos](https://gpuopen.com/videos)

<https://www.youtube.com/watch?v=i4Evh4SjtUg>

# RADEON DRIVER - NEW RAYTRACING FEATURES

## Instance Rebraiding and Triangle Splitting

- Improved spatial coherence
- Overall traversal speedup
- May occupy larger memory
- Might take longer to build
- Automatically activated by the driver
- No user input required!

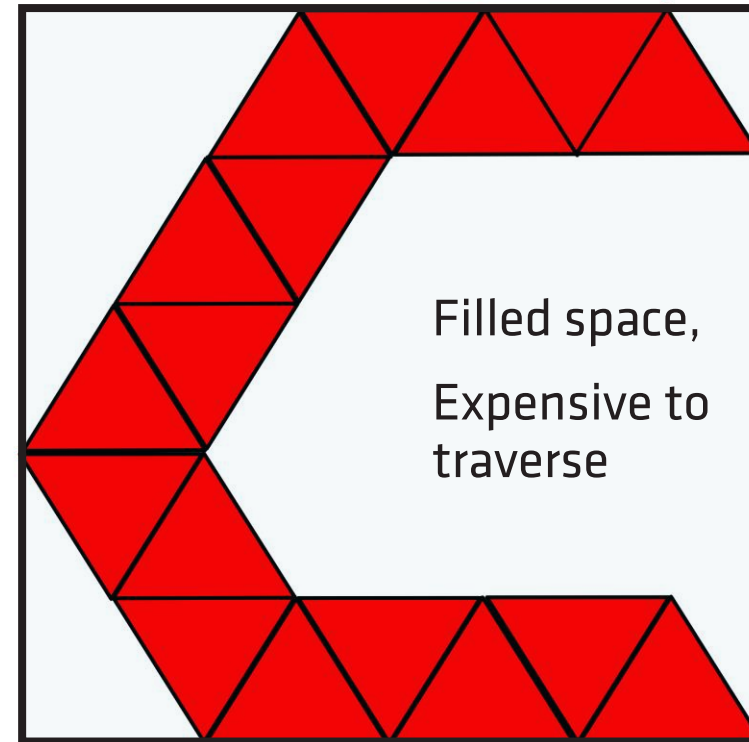




# RADEON RAYTRACING ANALYZER 1.1 - UPDATE

## Instance Rebraiding

- Instances are automatically rebraided based on transform and BLAS complexity by the driver
- The immediate child nodes of BLAS root will be encoded into the TLAS when the instance is rebraided
- Traversal loop may perform less iterations as a result of reduced surface area

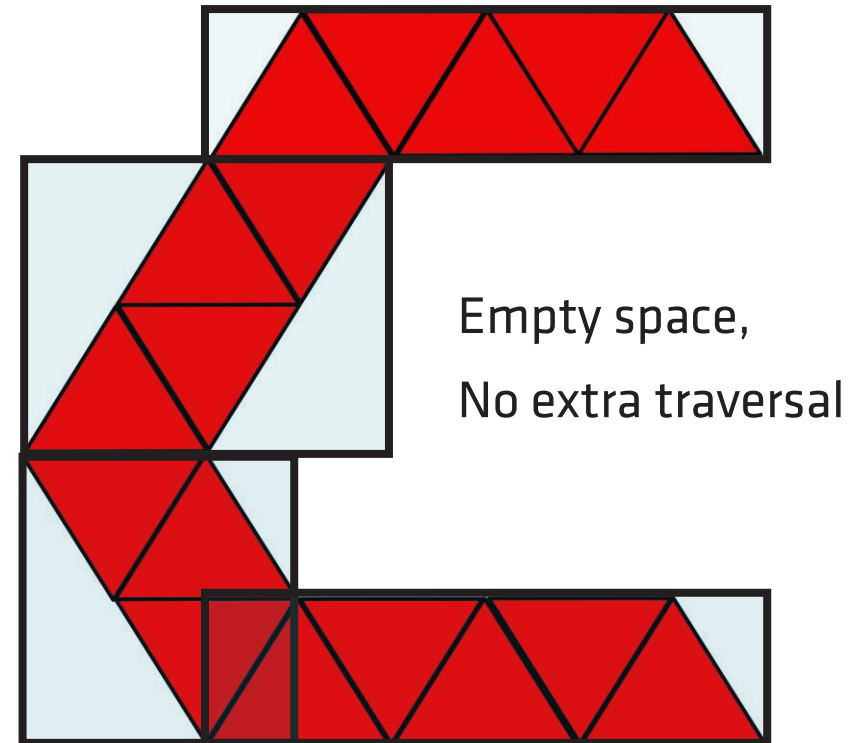




# RADEON RAYTRACING ANALYZER 1.1 - UPDATE

## Instance Rebraiding

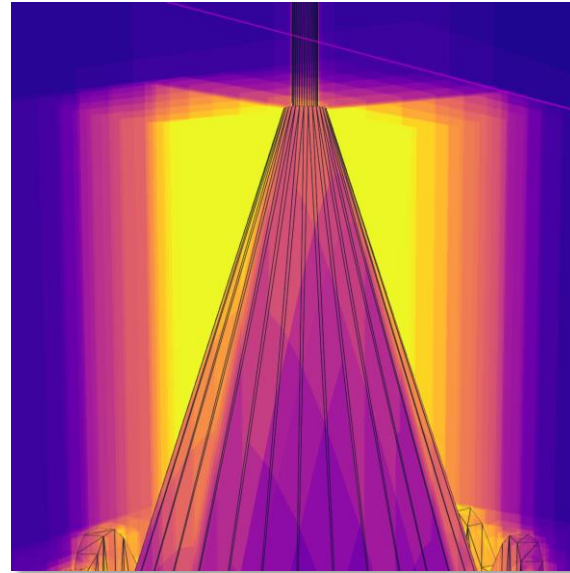
- Instances are automatically rebraided based on transform and BLAS complexity by the driver
- The immediate child nodes of BLAS root will be encoded into the TLAS when the instance is rebraided
- Traversal loop may perform less iterations as a result of reduced surface area



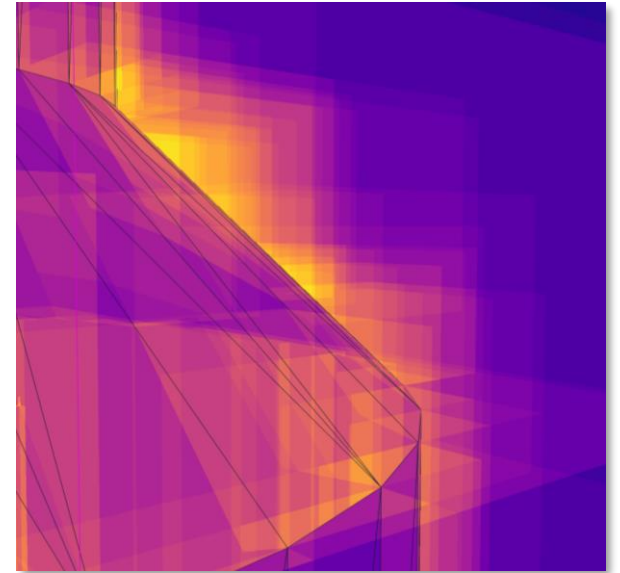
# RADEON RAYTRACING ANALYZER 1.1 UPDATE

## Triangle Splitting

- Triangle nodes are automatically split based on size and orientation by the driver to provide better early termination in ray-triangle tests
- If a triangle node is split, the parent bounding boxes will overlap with parts of the triangle but still point to the same geometry and primitive index



Without triangle splitting,  
Compounding traversal cost



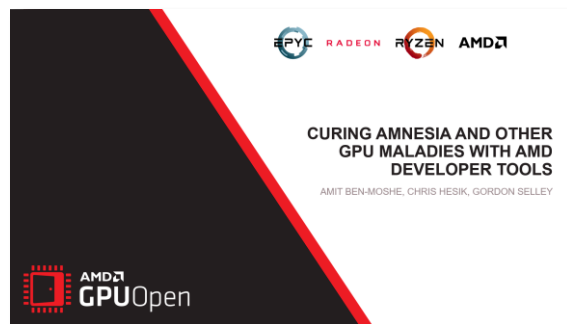
With triangle splitting  
Reduced traversal cost

(Heatmap - yellow is more expensive to traverse)

# AMD

# RADEON

## Memory Visualizer



### Detailed video of RMV

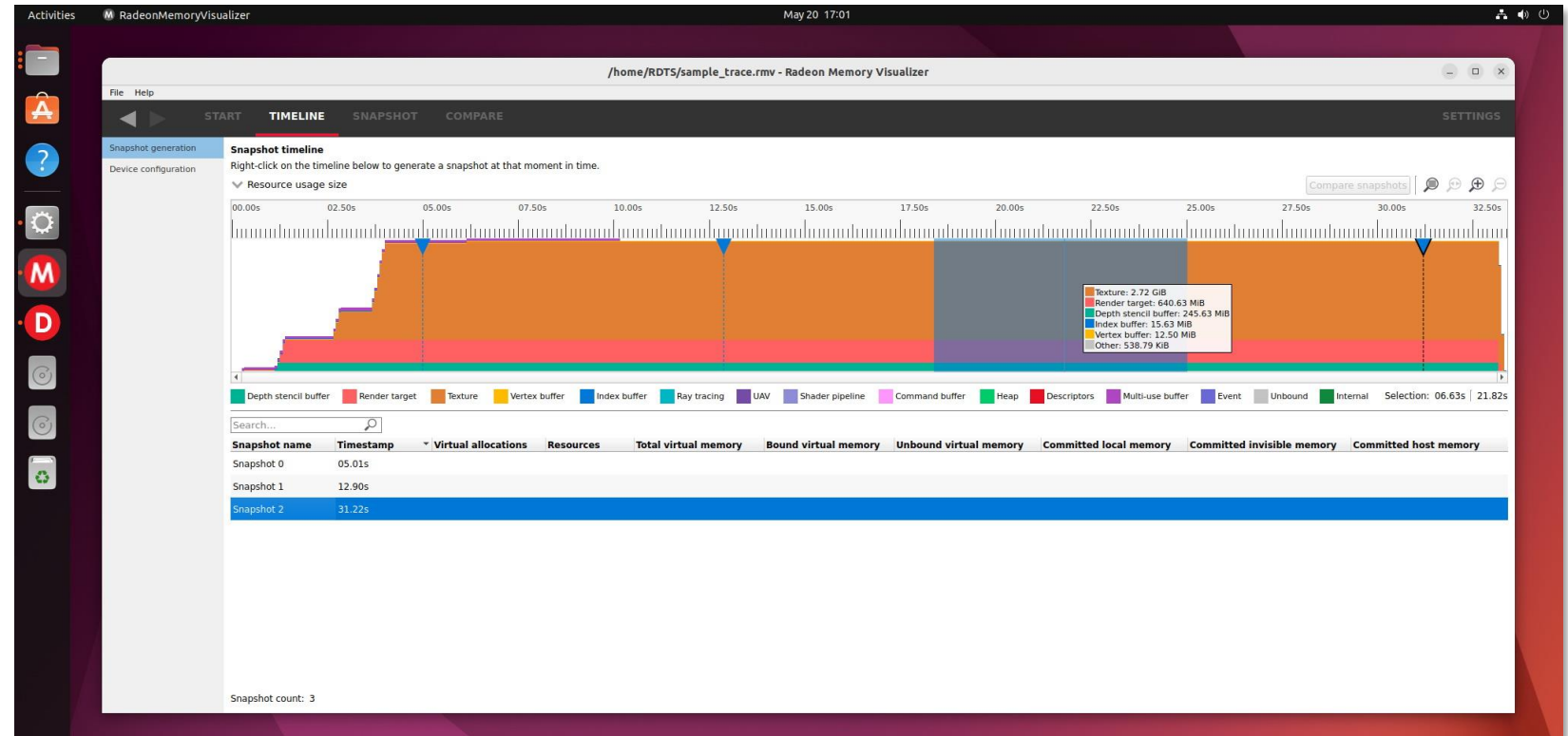
“Curing Amnesia and Other GPU Maladies with AMD Developer Tools”

[gpuopen.com/videos](https://gpuopen.com/videos)

<http://www.youtube.com/watch?v=2tmLQVn36P8?t=24m48s>

# RADEON MEMORY VISUALIZER 1.4.1 - UPDATE

Introduced full capture and visualization on Linux

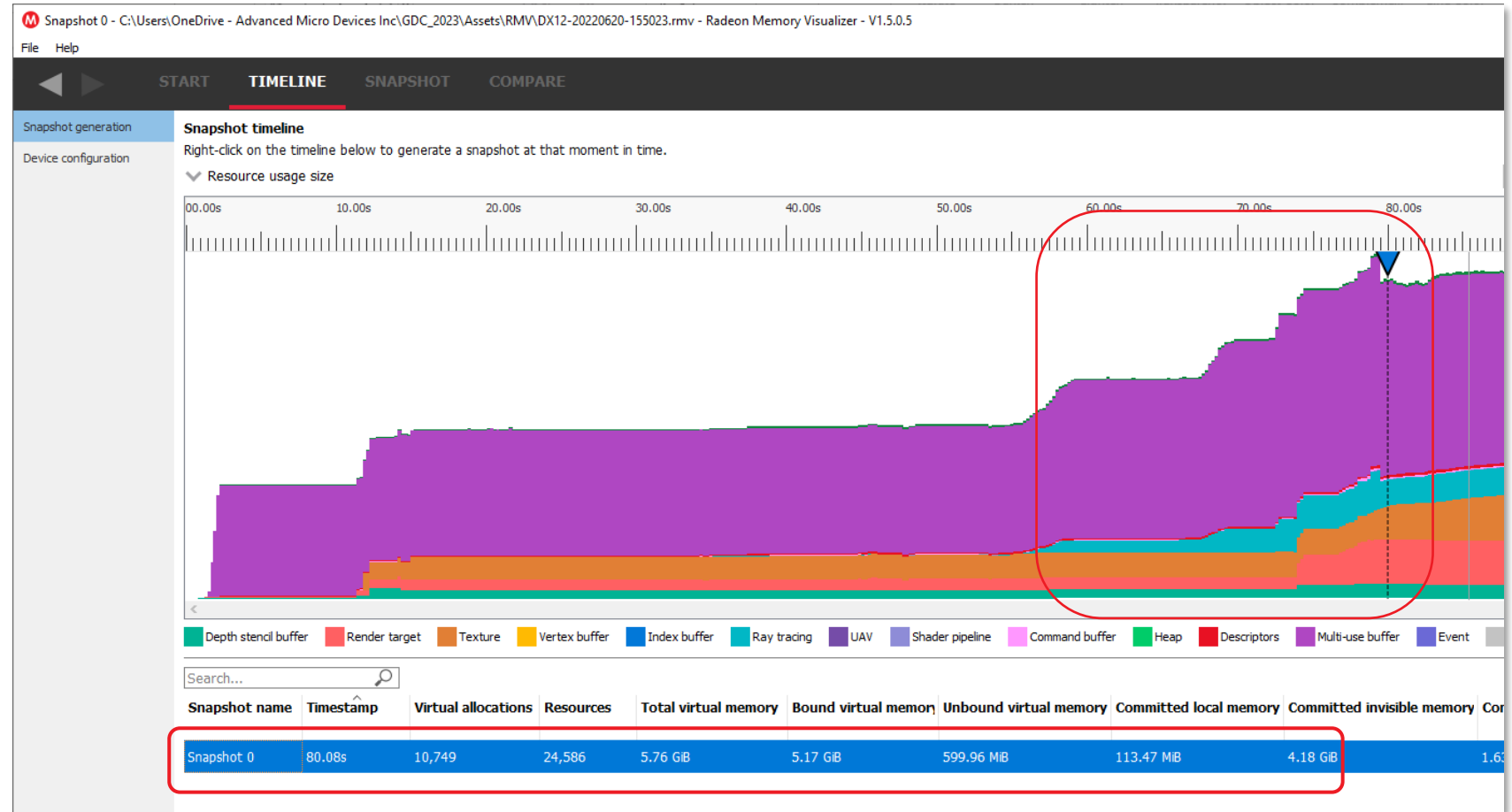


\* Note: Raytracing support dependent on AMD driver version

# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

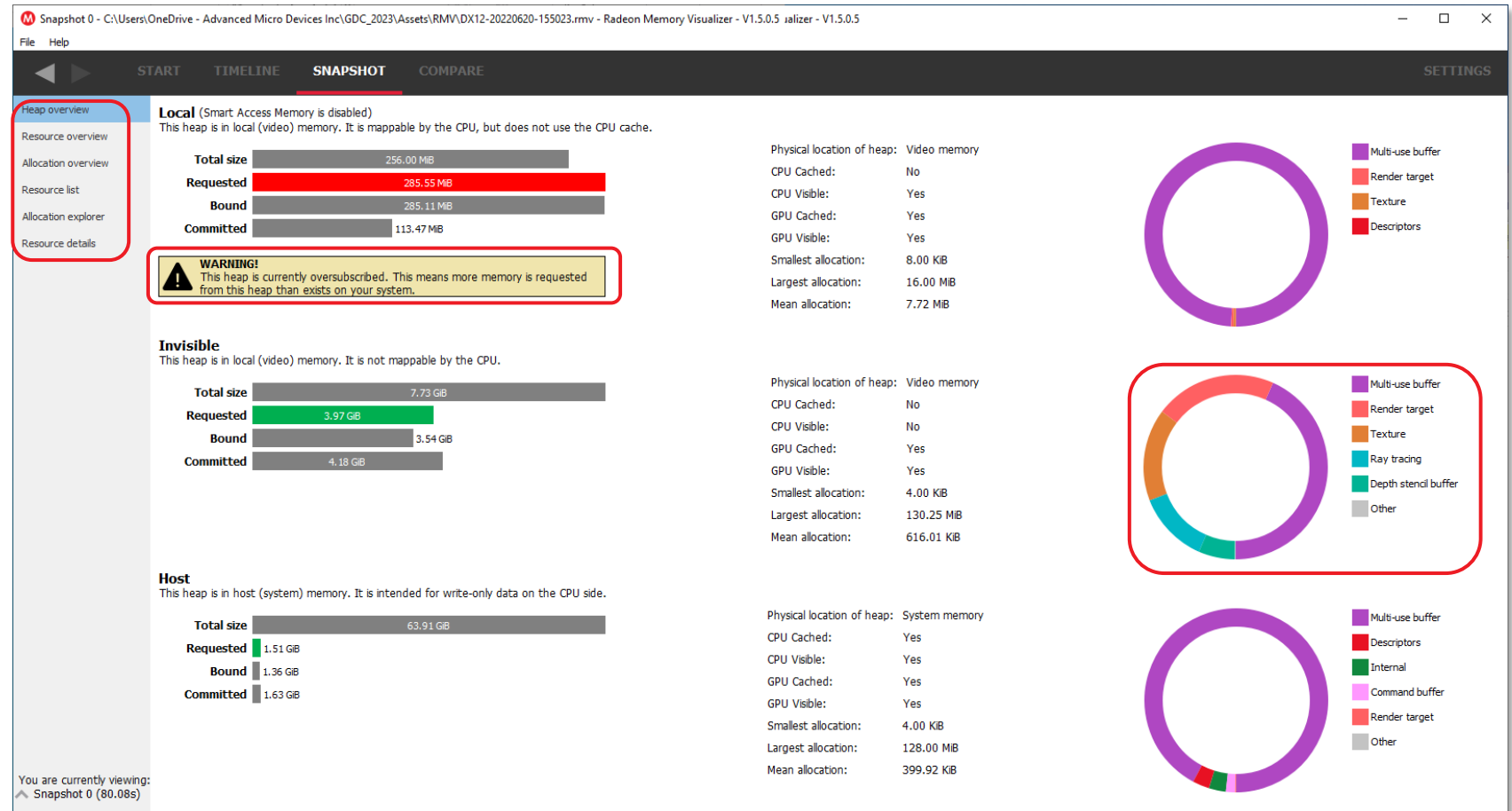
- Timeline view



# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

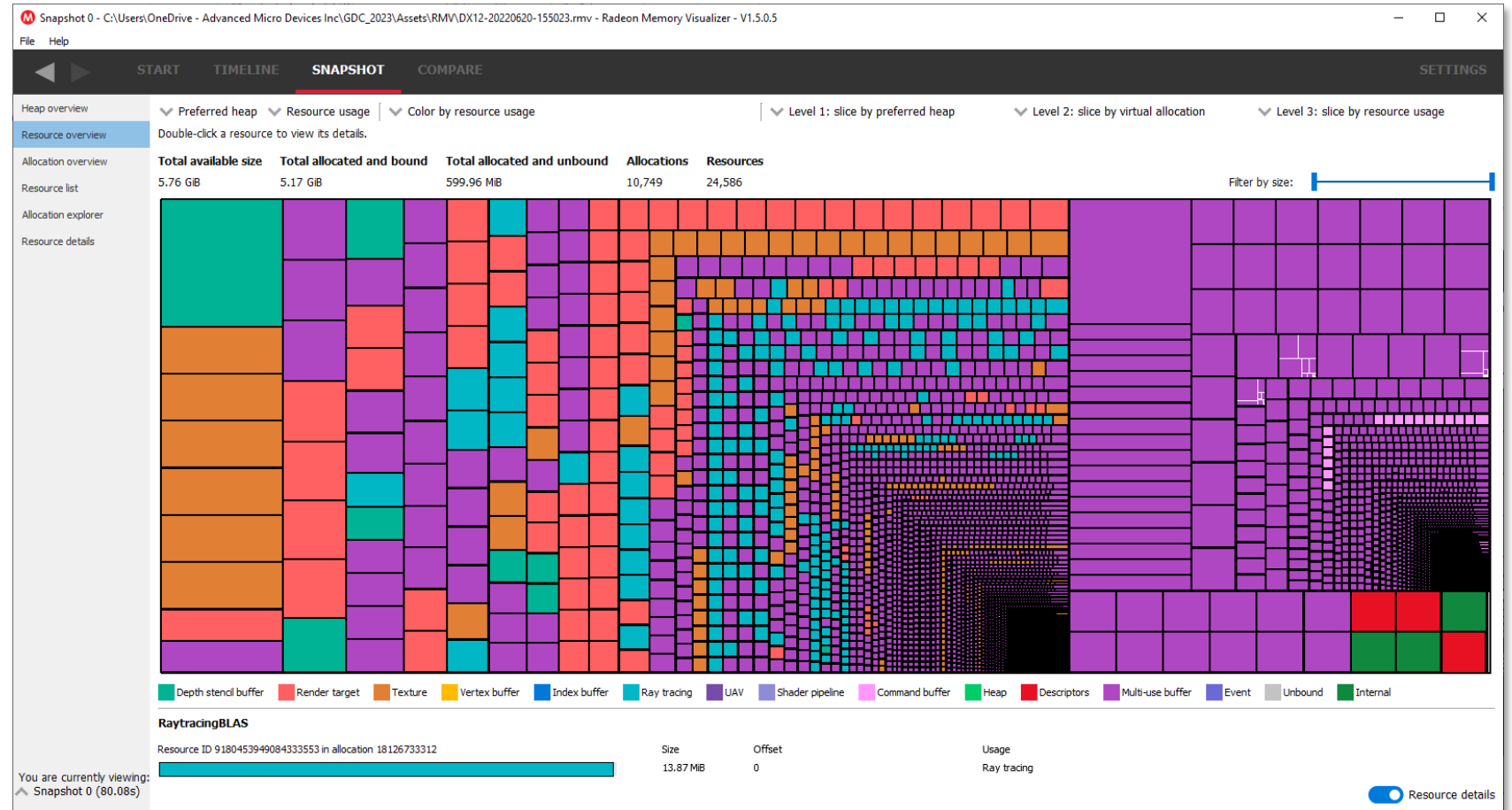
- Timeline view
- Heap overview



# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

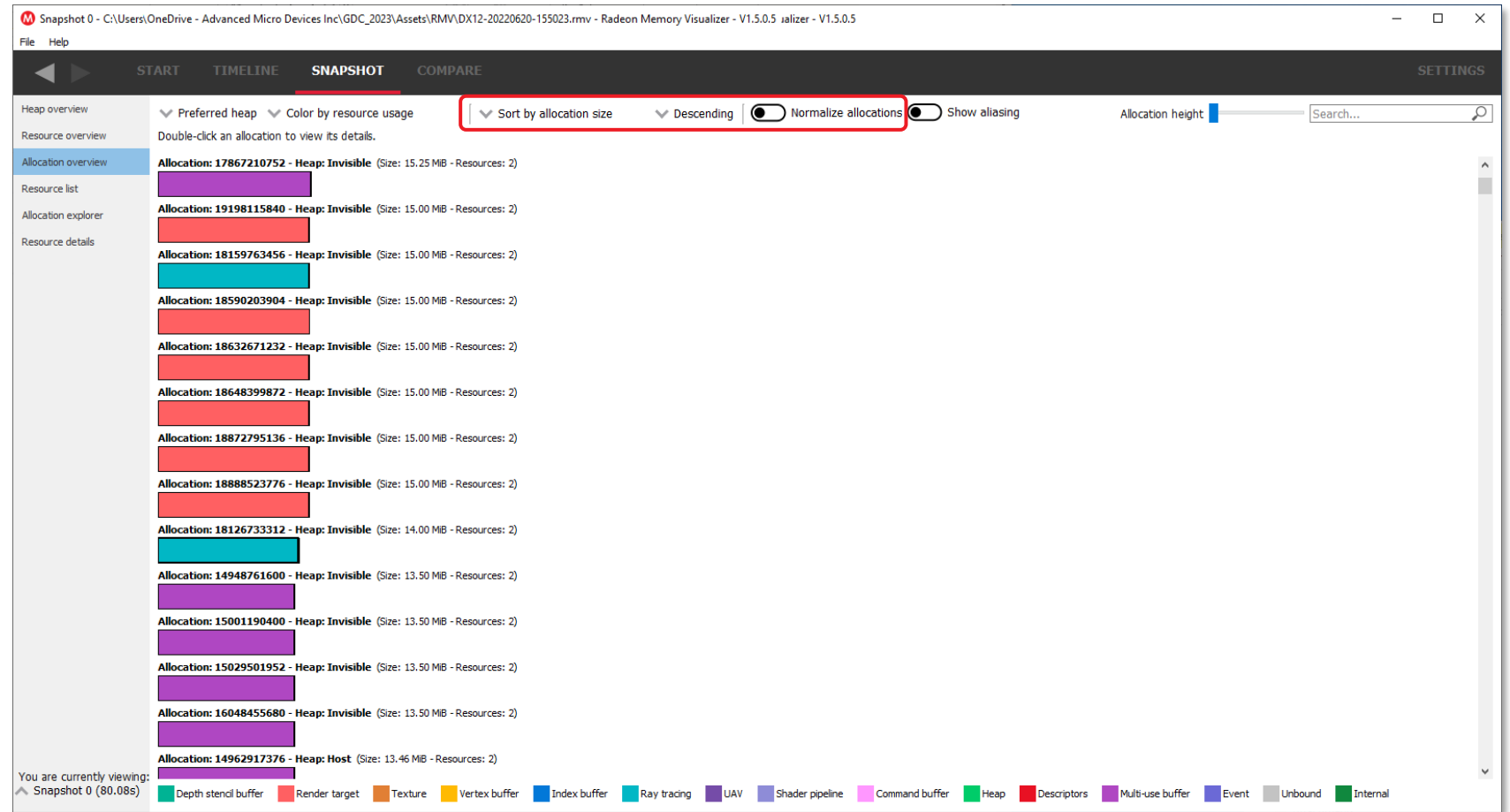
- Timeline view
- Heap overview
- Resource overview



# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

- Timeline view
- Heap overview
- Resource overview
- Allocation overview

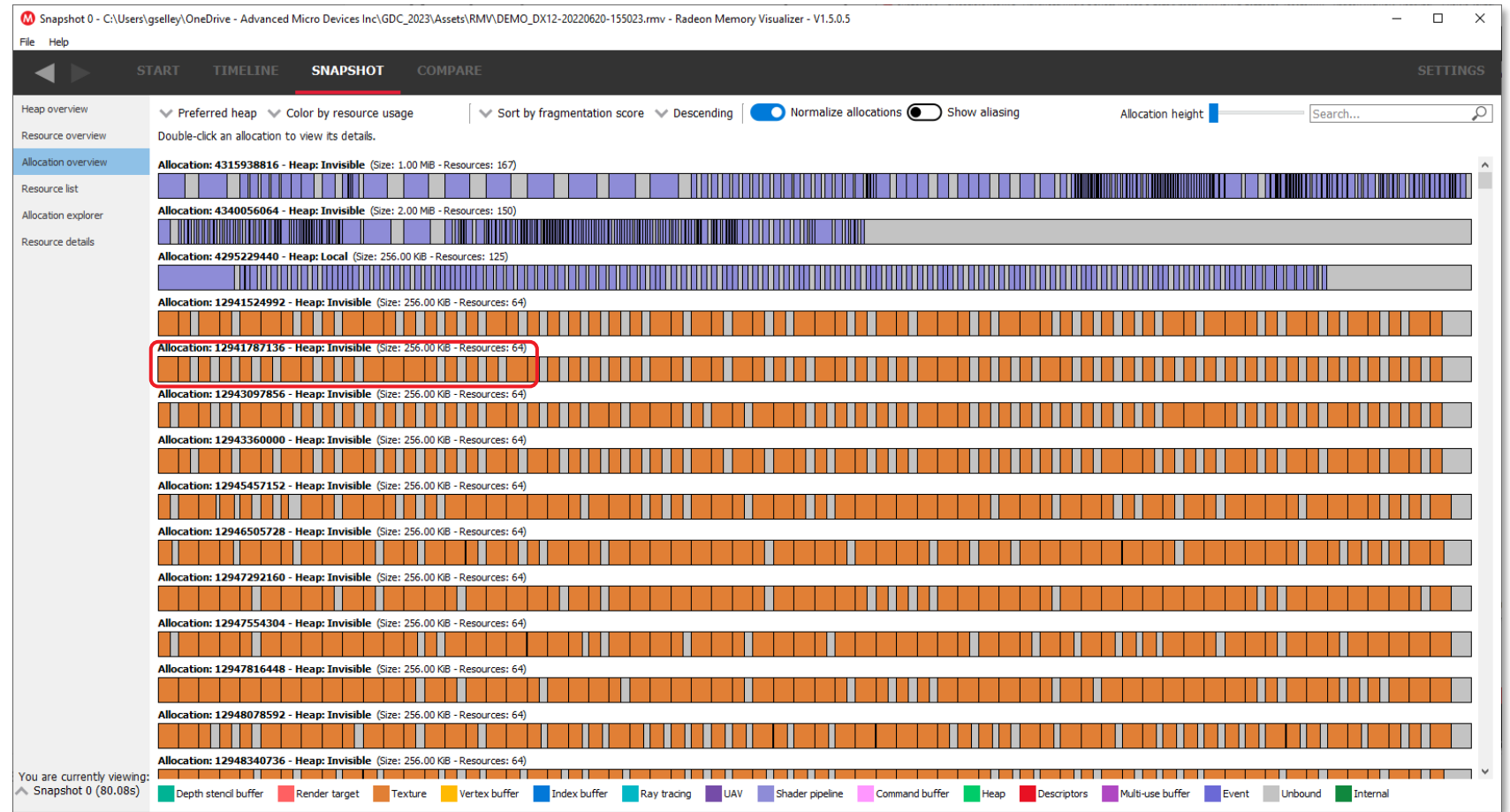




# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

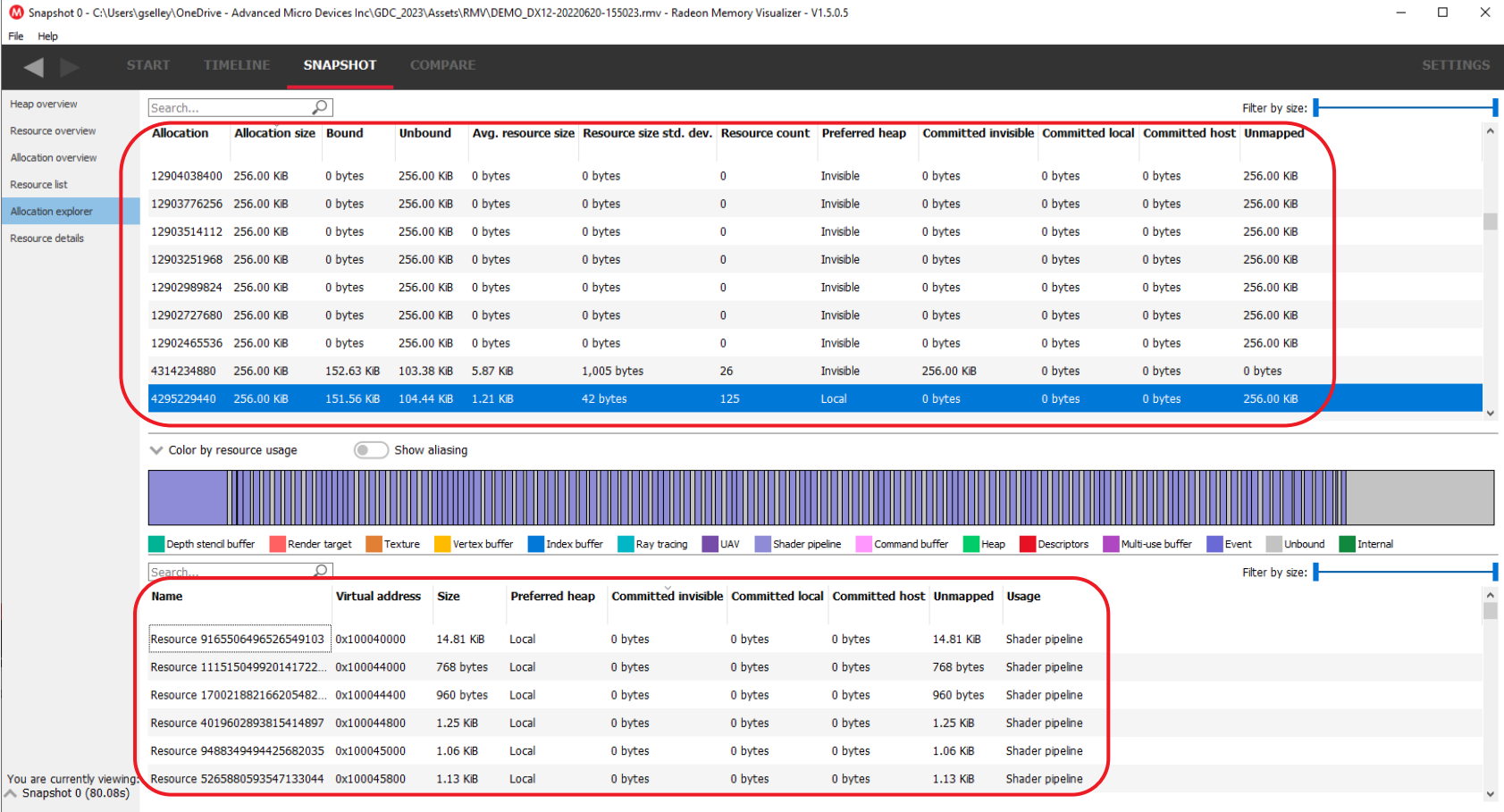
- Timeline view
- Heap overview
- Resource overview
- Allocation overview



# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

- Timeline view
- Heap overview
- Resource overview
- Allocation overview
- Allocation explorer



Snapshot 0 - C:\Users\gsley\OneDrive - Advanced Micro Devices Inc\GDC\_2023\Assets\RMV\DEMO\_DX12-20220620-155023.rmv - Radeon Memory Visualizer - V1.5.0.5

File Help

START TIMELINE **SNAPSHOT** COMPARE SETTINGS

Heap overview Search... Filter by size: [Slider]

Resource overview

Allocation overview

Resource list

**Allocation explorer**

Resource details

Allocation	Allocation size	Bound	Unbound	Avg. resource size	Resource size std. dev.	Resource count	Preferred heap	Committed invisible	Committed local	Committed host	Unmapped
12904038400	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
12903776256	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
12903514112	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
12903251968	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
12902989824	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
12902727680	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
12902465536	256.00 KiB	0 bytes	256.00 KiB	0 bytes	0 bytes	0	Invisible	0 bytes	0 bytes	0 bytes	256.00 KiB
4314234880	256.00 KiB	152.63 KiB	103.38 KiB	5.87 KiB	1,005 bytes	26	Invisible	256.00 KiB	0 bytes	0 bytes	0 bytes
4295229440	256.00 KiB	151.56 KiB	104.44 KiB	1.21 KiB	42 bytes	125	Local	0 bytes	0 bytes	0 bytes	256.00 KiB

Color by resource usage Show aliasing

Depth stencil buffer Render target Texture Vertex buffer Index buffer Ray tracing UAV Shader pipeline Command buffer Heap Descriptors Multi-use buffer Event Unbound Internal

Search... Filter by size: [Slider]

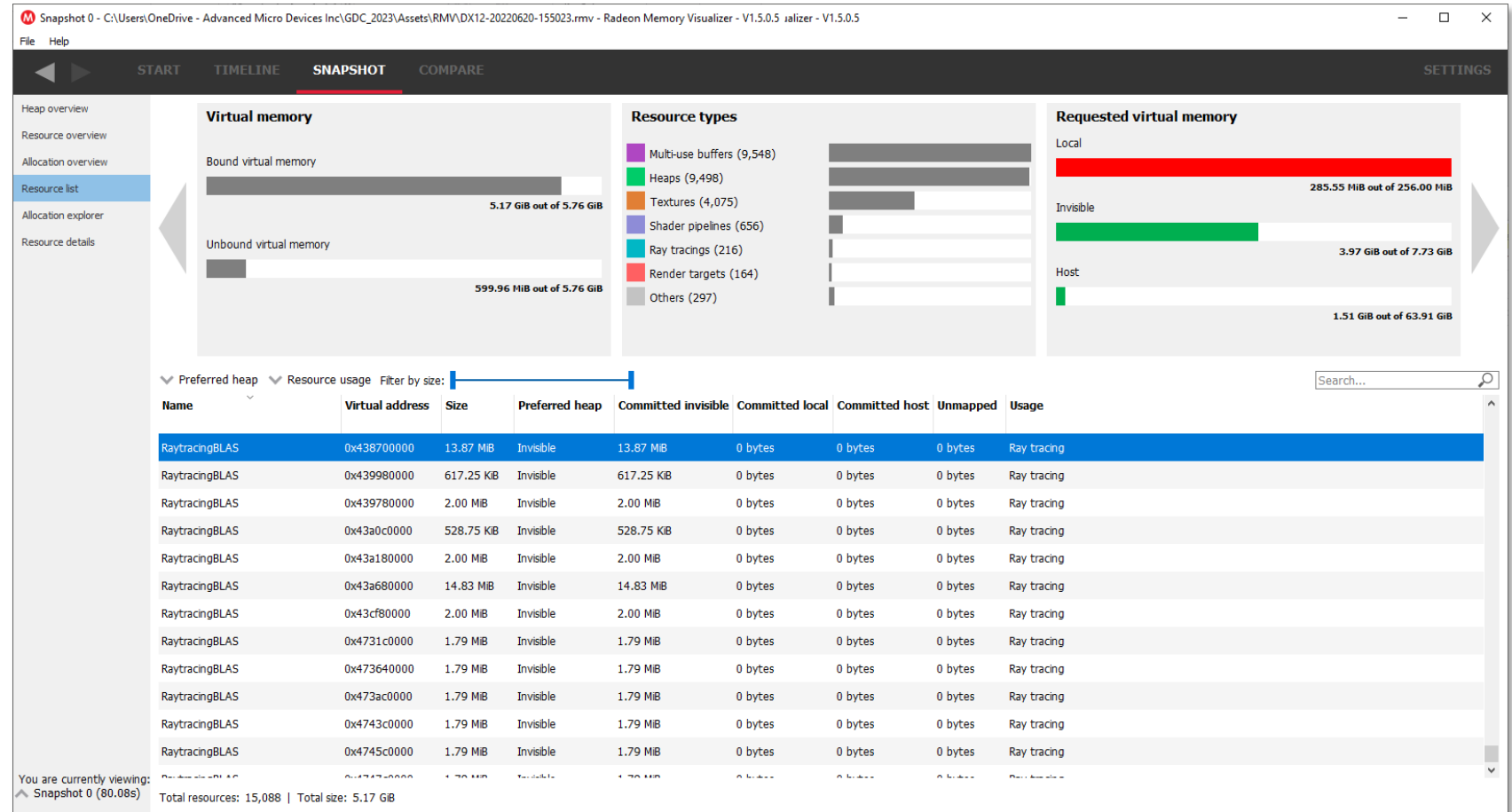
Name	Virtual address	Size	Preferred heap	Committed invisible	Committed local	Committed host	Unmapped	Usage
Resource 9165506496526549103	0x100040000	14.81 KiB	Local	0 bytes	0 bytes	0 bytes	14.81 KiB	Shader pipeline
Resource 111515049920141722...	0x100044000	768 bytes	Local	0 bytes	0 bytes	0 bytes	768 bytes	Shader pipeline
Resource 170021882166205482...	0x100044400	960 bytes	Local	0 bytes	0 bytes	0 bytes	960 bytes	Shader pipeline
Resource 4019602893815414897	0x100044800	1.25 KiB	Local	0 bytes	0 bytes	0 bytes	1.25 KiB	Shader pipeline
Resource 9488349494425682035	0x100045000	1.06 KiB	Local	0 bytes	0 bytes	0 bytes	1.06 KiB	Shader pipeline
Resource 5265880593547133044	0x100045800	1.13 KiB	Local	0 bytes	0 bytes	0 bytes	1.13 KiB	Shader pipeline

You are currently viewing: Snapshot 0 (80.08s)

# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

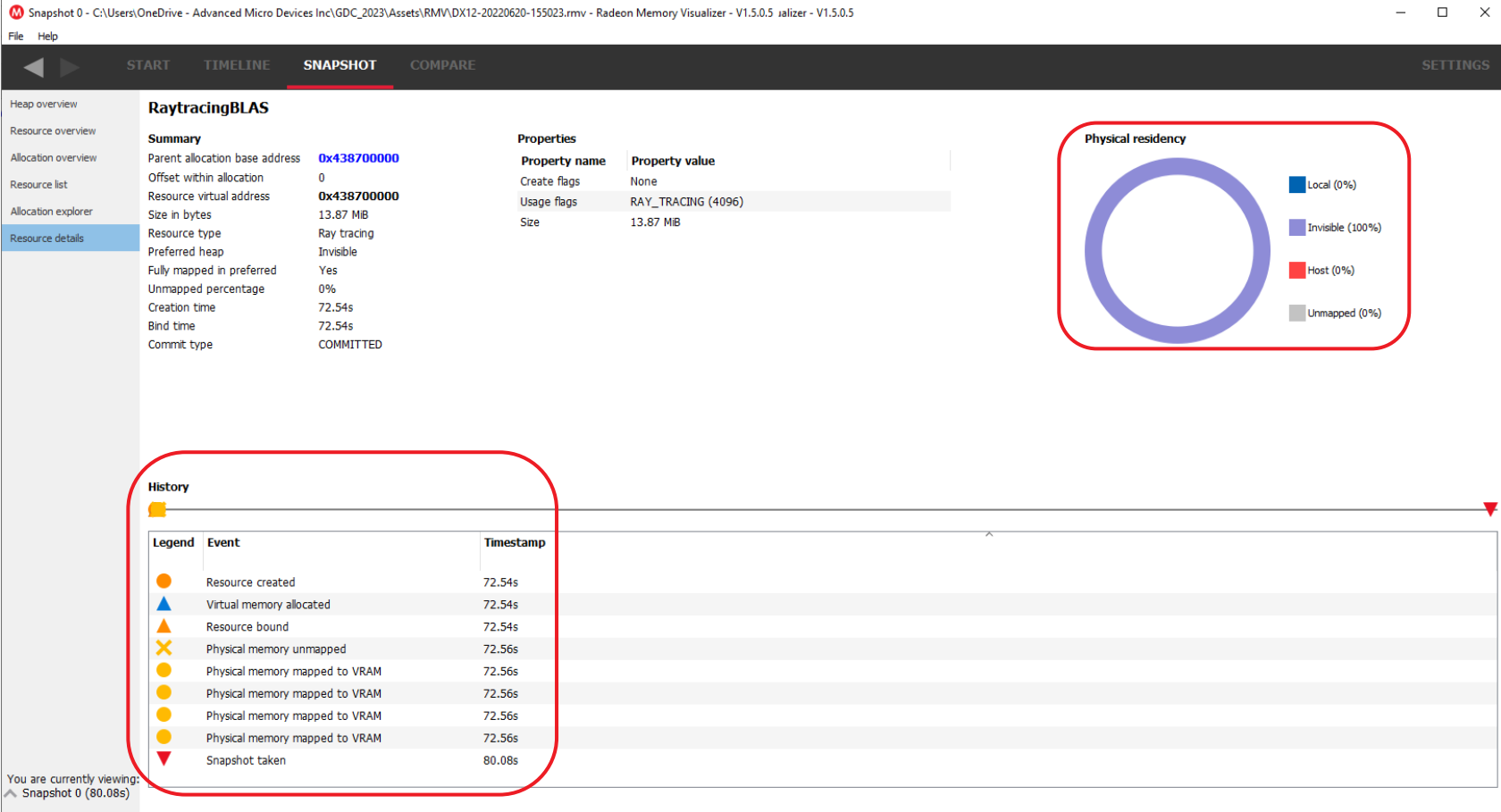
- Timeline view
- Heap overview
- Resource overview
- Allocation overview
- Allocation explorer
- Resource List



# RADEON MEMORY VISUALIZER 1.5 - UPDATE

Ray tracing resources are colored in all views as 

- Timeline view
- Heap overview
- Resource overview
- Allocation overview
- Allocation explorer
- Resource List
- Resource Details



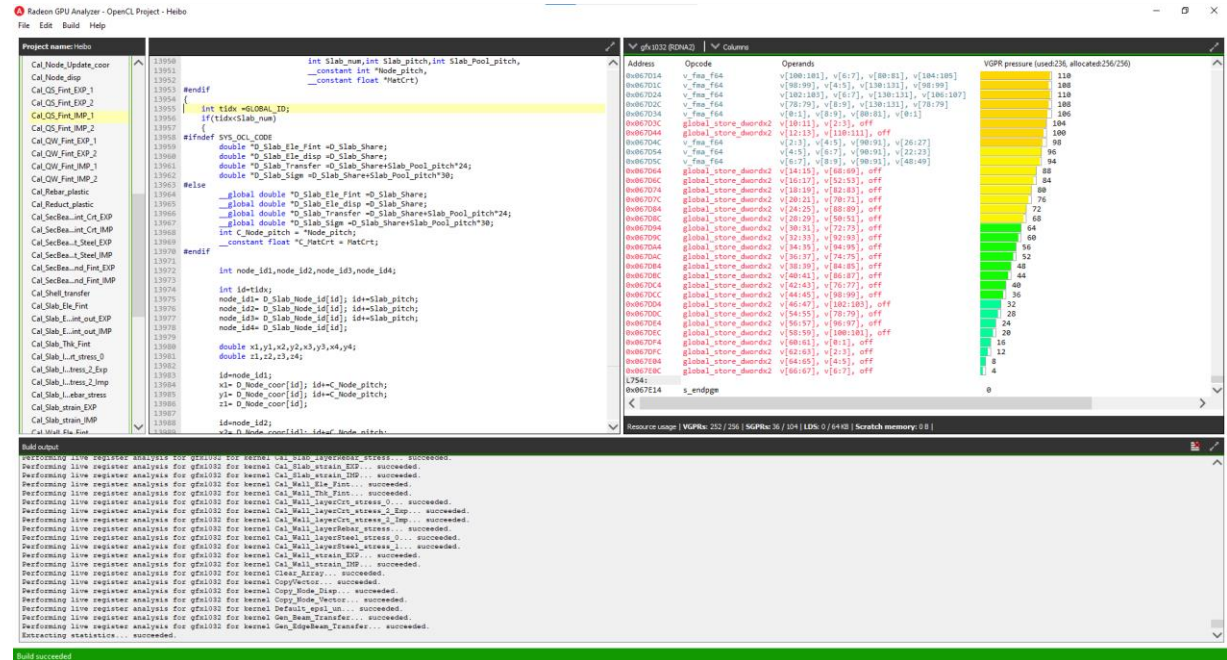
The screenshot displays the Radeon Memory Visualizer 1.5.0.5 interface. The main window shows the 'RaytracingBLAS' resource details. The 'Summary' section lists properties such as Parent allocation base address (0x43870000), Offset within allocation (0), Resource virtual address (0x43870000), Size in bytes (13.87 MiB), Resource type (Ray tracing), Preferred heap (Invisible), Fully mapped in preferred (Yes), Unmapped percentage (0%), Creation time (72.54s), Bind time (72.54s), and Commit type (COMMITTED). The 'Properties' section shows Property name (None), Create flags (None), Usage flags (RAY\_TRACING (4096)), and Size (13.87 MiB). The 'Physical residency' section shows a donut chart with a legend: Local (0%), Invisible (100%), Host (0%), and Unmapped (0%). The 'History' section shows a table of events with a legend: Resource created (orange circle), Virtual memory allocated (blue triangle), Resource bound (orange triangle), Physical memory unmapped (yellow X), Physical memory mapped to VRAM (yellow circle), and Snapshot taken (red inverted triangle). The table lists events with their timestamps, including Resource created (72.54s), Virtual memory allocated (72.54s), Resource bound (72.54s), Physical memory unmapped (72.56s), and multiple Physical memory mapped to VRAM events (72.56s). The bottom status bar indicates 'You are currently viewing: Snapshot 0 (80.08s)'.

Legend	Event	Timestamp
●	Resource created	72.54s
▲	Virtual memory allocated	72.54s
▲	Resource bound	72.54s
✕	Physical memory unmapped	72.56s
●	Physical memory mapped to VRAM	72.56s
●	Physical memory mapped to VRAM	72.56s
●	Physical memory mapped to VRAM	72.56s
●	Physical memory mapped to VRAM	72.56s
●	Physical memory mapped to VRAM	72.56s
▼	Snapshot taken	80.08s

# AMD RADEON GPU Analyzer

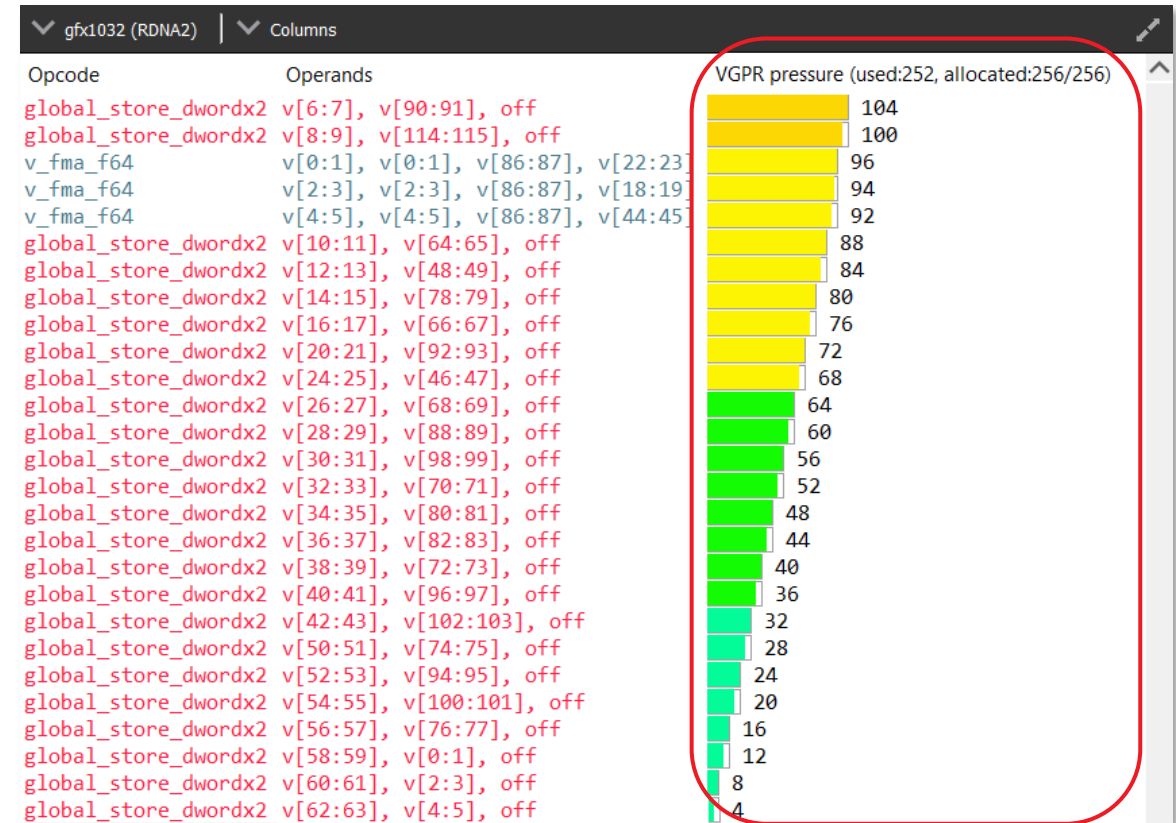
# RADEON GPU ANALYZER 2.7 - UPDATE

- Compile and analyze shaders and kernels for the RDNA3™ architecture (gfx1100, RX 7000 series GPUs)
  - Without a physical RDNA3™ card installed on your system
- New VGPR pressure GUI
  - Displays highest VGPR pressure and helps you identify where to focus your optimization
- DirectX®11 mode (-s dx11) is now an offline mode
  - Works regardless of the installed driver. Support in this mode was limited to VS, PS and CS
- Updates to Vulkan® and OpenGL® backend components
- OpenCL™ mode now uses an updated version of AMD's LLVM-based Lightning Compiler



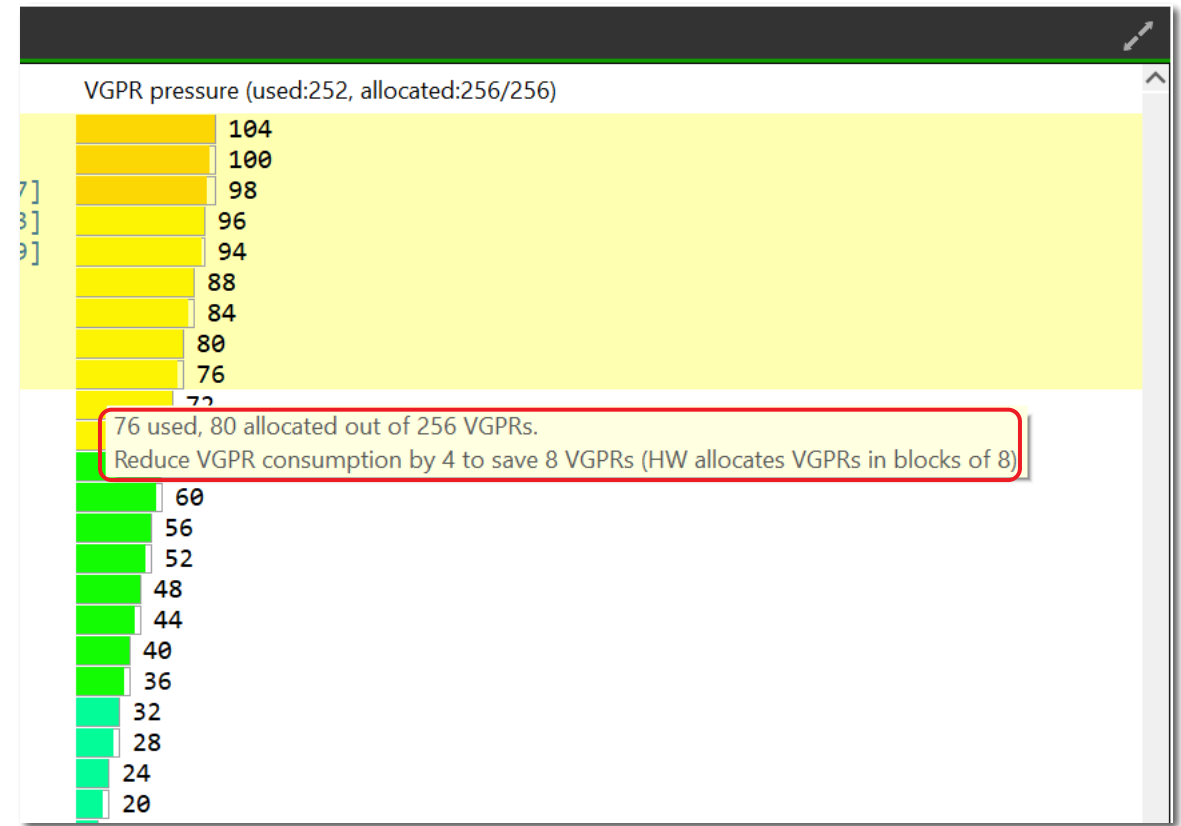
# RADEON GPU ANALYZER 2.7 - UPDATE

- Displays number of VGPRs used and allocated
- Utilization bars that visualize the VGPR usage and the VGPR block utilization



# RADEON GPU ANALYZER 2.7 - UPDATE

- Displays number of VGPRs used and allocated
- Utilization bars that visualize the VGPR usage and the VGPR block utilization
- Tooltips of live VGPRs per instruction with reduction hints





# RADEON GPU ANALYZER 2.7 - UPDATE

- Displays number of VGPRs used and allocated
- Utilization bars that visualize the VGPR usage and the VGPR block utilization
- Tooltips of live VGPRs per instruction with reduction hints
- F4 shortcut will cycle through the lines with maximum VGPRs. Shift+F4 will do the same in reverse order

gfx1032 (RDNA2)   Columns			
Opcode	Operands	VGPR pressure (used:236, allocated:256/256)	
v_fma_f64	v[144:145], v[26:27], v[198:199], v[216:217]		236
v_fma_f64	v[216:217], v[52:53], v[198:199], v[218:219]		236
v_fma_f64	v[172:173], v[4:5], v[198:199], v[172:173]		234
s_waitcnt	vmcnt(10)		232
v_fma_f64	v[198:199], v[26:27], v[202:203], v[220:221]		234
v_fma_f64	v[220:221], v[72:73], v[130:131], v[228:229]		234
s_waitcnt	vmcnt(8)		232
v_fma_f64	v[178:179], v[4:5], v[210:211], v[178:179]		232
v_fma_f64	v[228:229], v[52:53], v[210:211], v[246:247]		234
v_fma_f64	v[218:219], v[52:53], v[202:203], v[222:223]		234
v_mul_f64	v[126:127], v[200:201], v[174:175]		234
v_fma_f64	v[182:183], v[4:5], v[202:203], v[182:183]		234
v_fma_f64	v[202:203], v[68:69], v[130:131], v[224:225]		234
v_fma_f64	v[222:223], v[70:71], v[134:135], v[236:237]		234
v_fma_f64	v[224:225], v[74:75], v[134:135], v[238:239]		234
v_fma_f64	v[204:205], v[194:195], 0, v[204:205]		230
v_mul_f64	v[134:135], v[124:125], s[20:21]		232
v_fma_f64	v[168:169], v[4:5], v[208:209], v[168:169]		232
v_fma_f64	v[142:143], v[8:9], v[194:195], v[142:143]		232
v_fma_f64	v[192:193], v[50:51], v[184:185], v[192:193]		230
v_mul_f64	v[124:125], v[148:149], v[124:125]		230
v_fma_f64	v[120:121], -v[120:121], v[188:189], v[232:233]		228
v_mul_f64	v[138:139], v[48:49], v[190:191]		228
v_mul_f64	v[2:3], v[96:97], v[2:3]		228
v_mul_f64	v[190:191], v[6:7], v[190:191]		228
v_fma_f64	v[94:95], v[8:9], v[184:185], v[94:95]		228
v_fma_f64	v[130:131], v[80:81], v[130:131], v[234:235]		228
v_mul_f64	v[0:1], v[132:133], v[0:1]		226
s_waitcnt	vmcnt(7)		226
v_fma_f64	v[148:149], v[52:53], v[196:197], v[250:251]		228

# RADV & STEAM DECK SUPPORT

- RADV is a Vulkan® driver for AMD Radeon™ GPUs written for Linux® by the community as part of the Mesa project
- Radeon tools support in RADV enabled via environment variables for Steam Deck device
  - RGP support
    - **RADV\_THREAD\_TRACE**=*frameindex*
    - **RADV\_THREAD\_TRACE\_TRIGGER**=*filename*
    - “RGP Capture” button in the SteamOS Devkit client
  - RRA support
    - Added in Mesa 22.3
    - **RADV\_RRA\_TRACE**=*frameindex*
    - **RADV\_RRA\_TRACE\_TRIGGER**=*filename*
  - RMV support
    - Added in Mesa 23.0 (also requires Linux kernel 5.10 or newer)
    - **MESA\_VK\_MEMORY\_TRACE**=*frameindex*
    - **MESA\_VK\_MEMORY\_TRACE\_TRIGGER**=*filename*

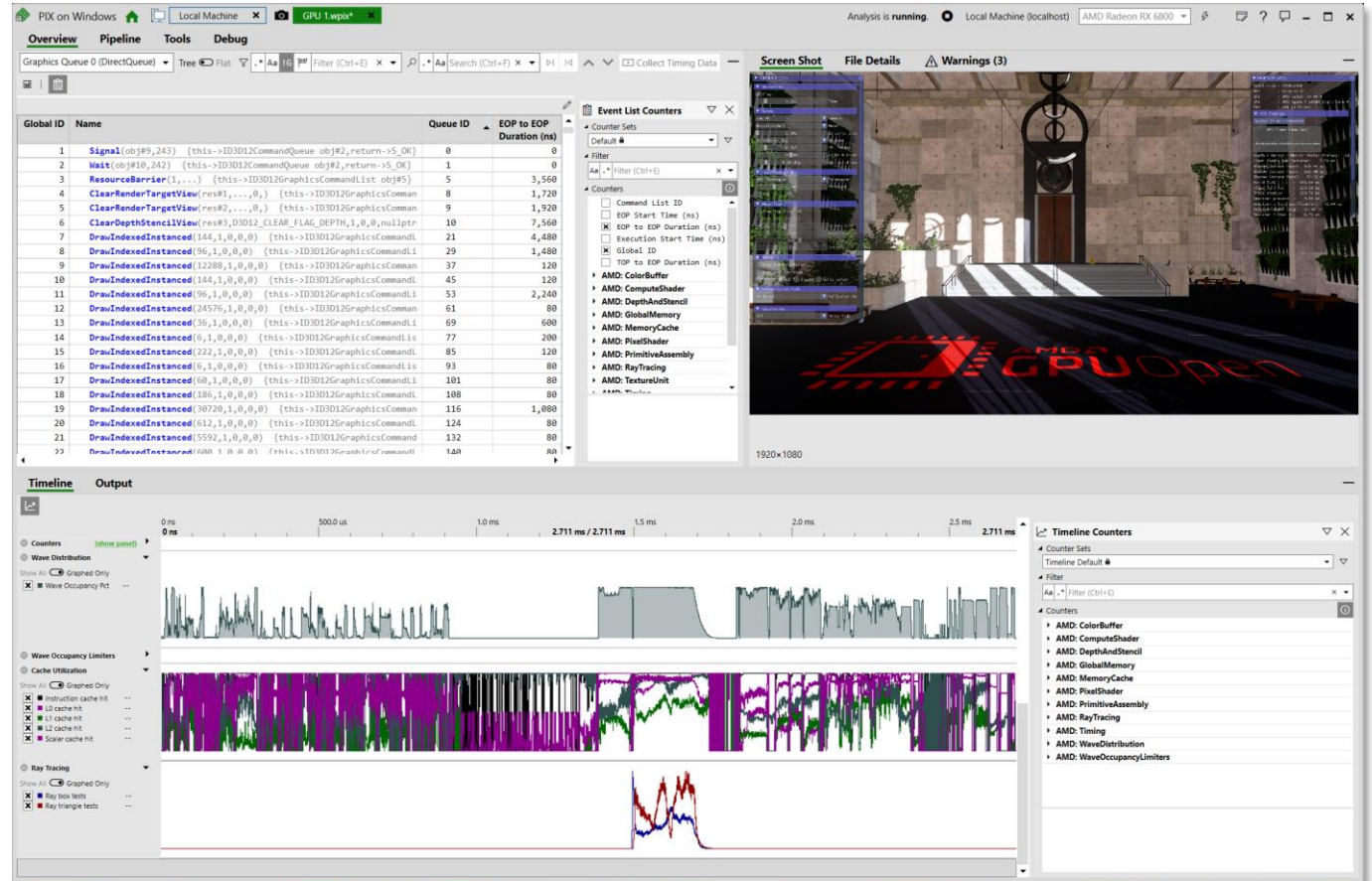


# THIS PRESENTATION

- Part 1: Overview of the Radeon™ Developer Tool Suite
  - What is in it
  - What's new since GDC 2022
  - Dive into new features and improvements
  - RADV & Steam Deck support
- **Part 2: Collaboration with external tool developers**
  - Microsoft® PIX on Windows
  - RenderDoc
  - GFXReconstruct

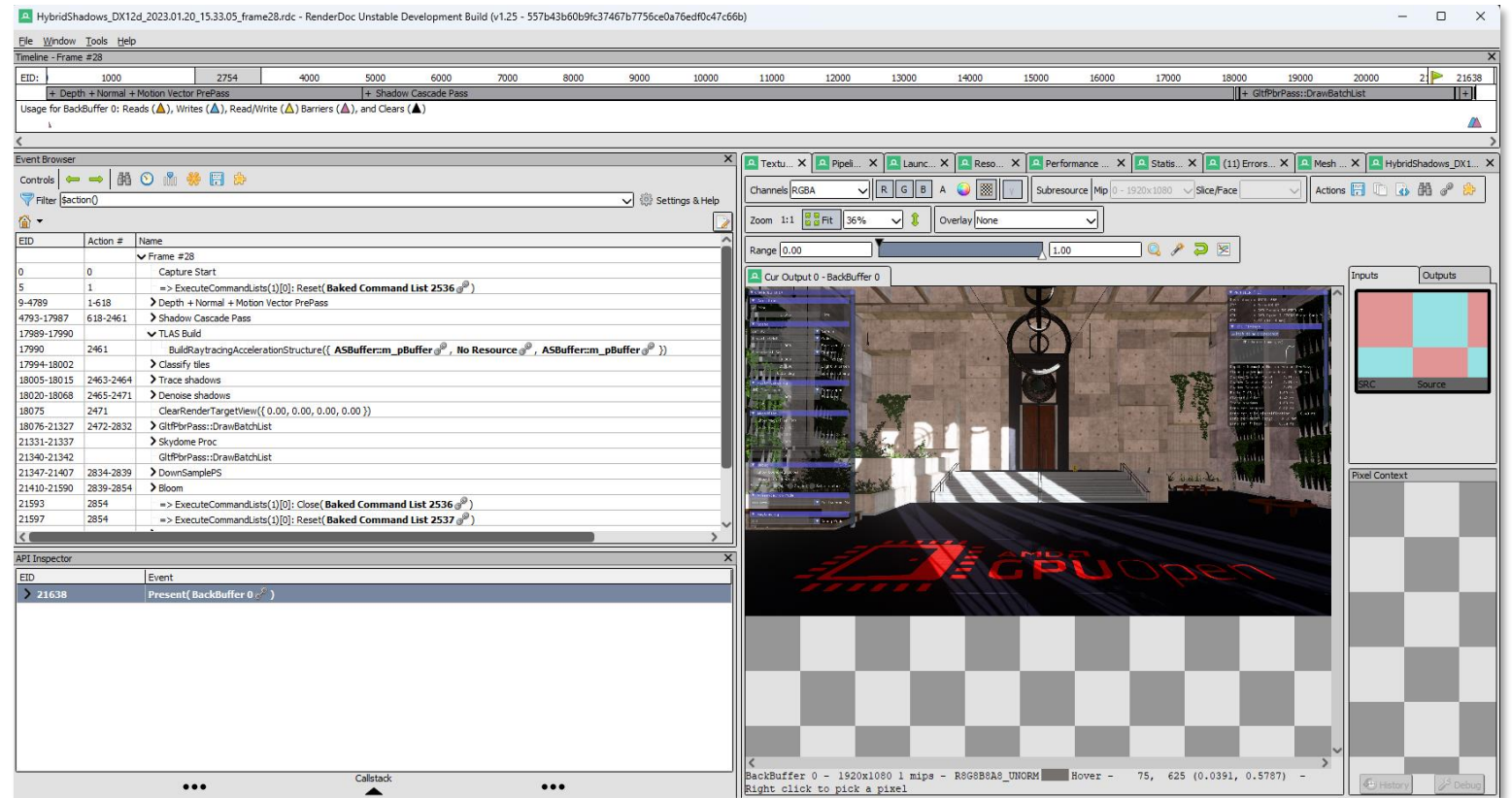
# MICROSOFT PIX ON WINDOWS

- Debugging and performance analysis
- Event List Counters
- High frequency counter graphs
  - Wave distribution
  - Wave occupancy limiters
  - Cache utilization – **NEW!**
    - Scalar cache hit
    - Instruction cache hit
    - L0/L1/L2 cache hit
  - Raytracing – **NEW!**
    - Ray box tests
    - Ray triangle tests
- <https://devblogs.microsoft.com/pix/>
- DirectX Discord server: discord.gg/directx



# RENDERDOC - OVERVIEW

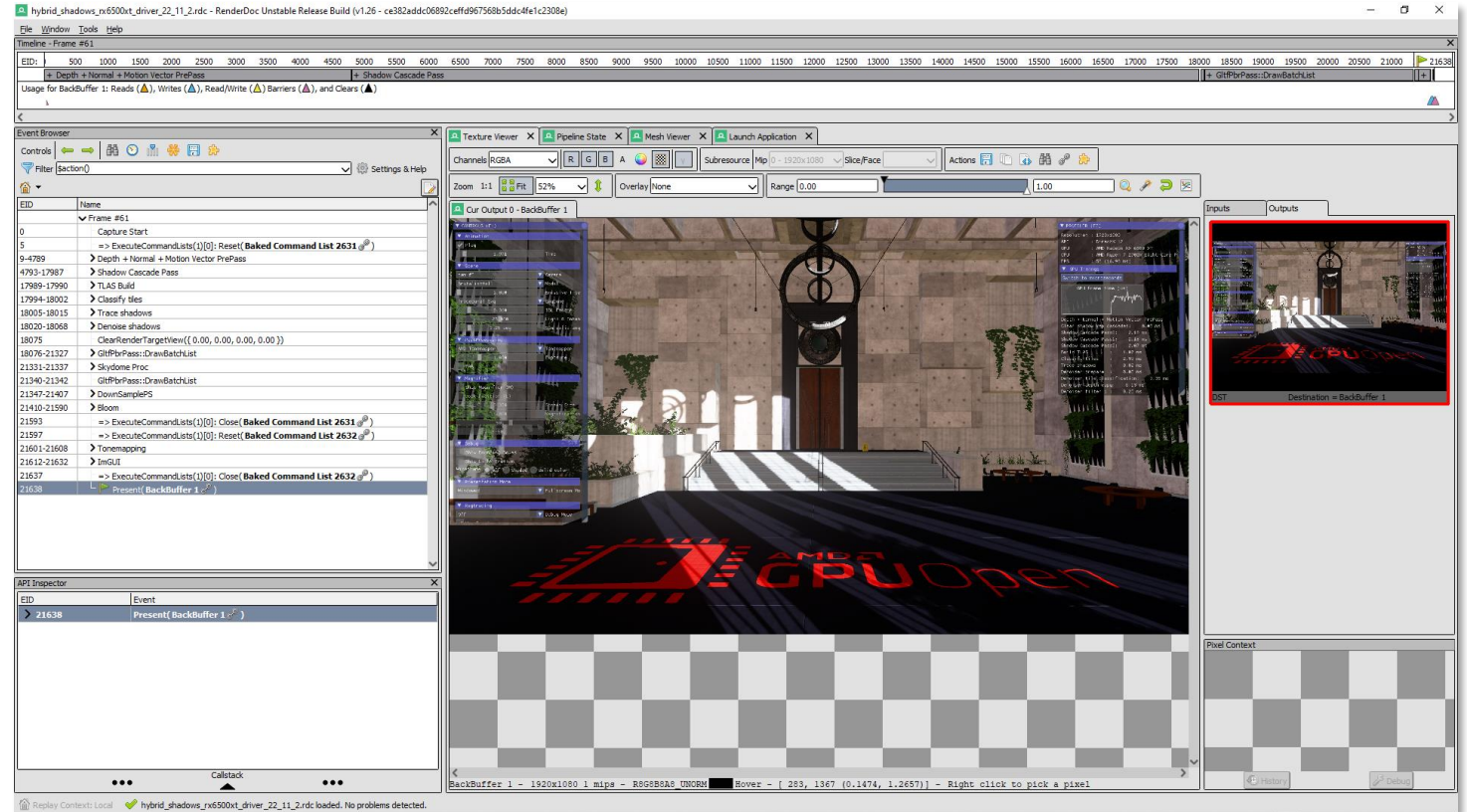
- RenderDoc is a free MIT licensed stand-alone graphics debugger
- Allows quick and easy single-frame capture and detailed introspection
- Applications using Vulkan, D3D11, OpenGL & OpenGL ES or D3D12 across Windows, Linux, Android
- RDNA3 GPA Counter support – **NEW!**





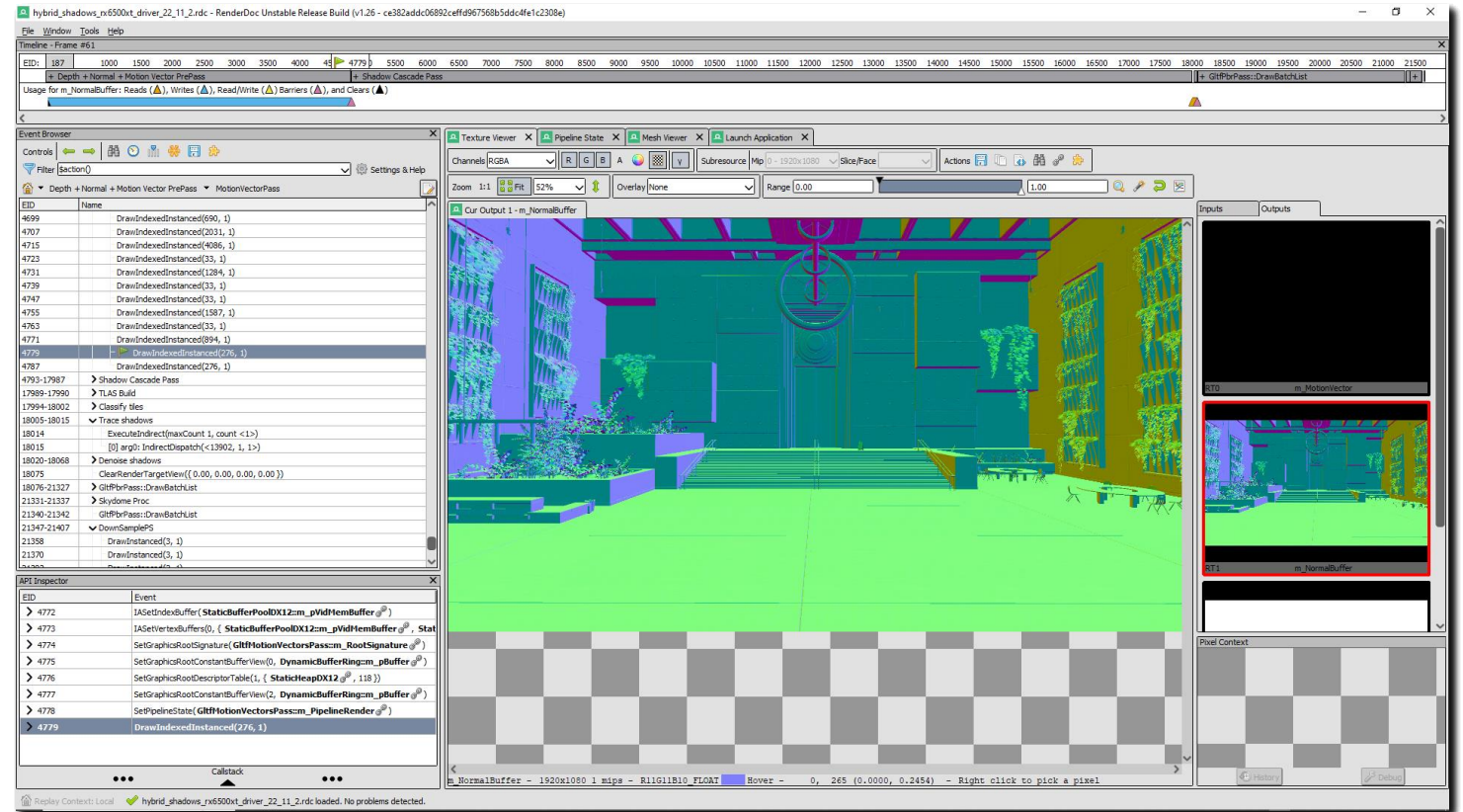
# RENDERDOC – WORK IN PROGRESS

- Collaboration between AMD and Baldur Karlsson to support capture of DXR applications
- DXR Support
  - Capture/replay DXR workload



# RENDERDOC – RGP INTEROP

- Generate RGP profile from capture file
- Profile opened in RGP
- Select event in RenderDoc
- Jump to same event in RGP
- Select event in RGP
- Jump to same event in RenderDoc



# GFXRECONSTRUCT + D3D12 + DXR

- What is it?

- Graphics workload capture/replay software suite
- Replacement for vktrace
- Command line based
- Added D3D12 + DXR support in joint venture between LunarG and AMD

- Highlights

- Supports ray tracing
- Fully open-source
- Extensible software architecture
- Can capture/replay thousands of frames
- Lightweight

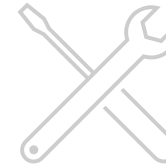




# GFXRECONSTRUCT + D3D12 + DXR

- **Provided components**

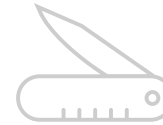
- Writing: DLLs to capture a GPU workload
- Reading: executables to replay and process the captured workload



- **Main tools:** gfxrecon-replay.exe and gfxrecon-optimize.exe

- **Other tools, that can:**

- Display the capture file metadata
- Change the compression algorithm
- Generate a human readable API dump

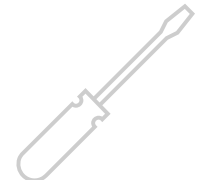


- **Use cases**

- Open-source toolkit
- Profiling specific frames
- Debugging visual artifacts
- Stress testing
- Experimentation platform

- **How to get it**

- Source code and binaries: <https://github.com/LunarG/gfxreconstruct>
- Blog: <https://gpuopen.com/learn/amd-lunarg-gfxreconstruct-dx12-dxr>



**Chris Hesik**

[christopher.hesik@amd.com](mailto:christopher.hesik@amd.com)

**Can Alper**

[can.alper@amd.com](mailto:can.alper@amd.com)

**Radeon Developer Tool Suite**

<https://gpuopen.com/tools/>

**Tool Suite Videos**

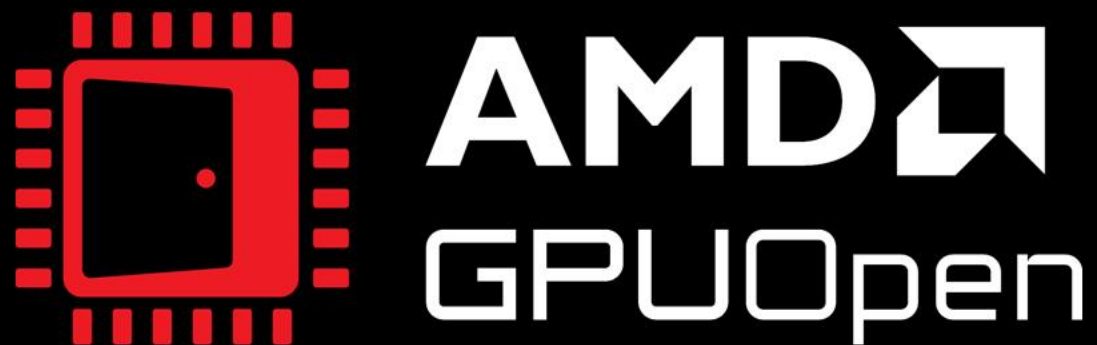
<https://gpuopen.com/videos/>

# DISCLAIMER

## GENERAL DISCLAIMER

The information contained herein is for informational purposes only and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale. GD-18

© 2023 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, Radeon and combinations thereof are trademarks of Advanced Micro Devices, Inc. DirectX is a registered trademark of Microsoft Corporation in the US and other jurisdictions. Linux is a registered trademark of Linux Torvalds. OpenGL® and the oval logo are trademarks or registered trademarks of Hewlett Packard Enterprise in the United States and/or other countries worldwide. OpenCL is a trademark of Apple, Inc. used by permission from The Khronos Group. SPIR, SPIR-V and the SPIR logo are trademarks of the Khronos Group Inc. Vulkan and the Vulkan logo are registered trademarks of the Khronos Group Inc. Windows is a registered trademark of Microsoft Corporation in the US and other jurisdictions. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.



**AMD**  
together we advance\_

**AMD**  
EPYC

**AMD**  
RYZEN

**AMD**  
RADEON

# ANY QUESTIONS?

## Part 1: Overview of the Radeon™ Developer Tool Suite

- What is in it
- What's new since GDC 2022
- Dive into new features and improvements
- RADV & Steam Deck support

## Part 2: Collaboration with external tool developers

- Microsoft® PIX on Windows
- RenderDoc
- GFXReconstruct

**Chris Hesik**

[christopher.hesik@amd.com](mailto:christopher.hesik@amd.com)

**Can Alper**

[can.alper@amd.com](mailto:can.alper@amd.com)

## Radeon Developer Tool Suite

<https://gpuopen.com/tools/>

## Tool Suite Videos

<https://gpuopen.com/videos/>