GAME OPTIMIZATION WITH THE AMD RADEON™ DEVELOPER TOOL SUITE

CHRIS HESIK (AMD)
AMIT MULAY (AMD)
Agenda

• Overview of the tools and new features
• Introduction to GPU Reshape
WHAT’S NEW?

GDC 2023

- **RGP 1.15**
  - Redesigned ISA disassembly view
  - Initial work graph support

March

- **RGD 1.0**
  - DirectX® 12 Support

April

- **RGD 1.1**
  - Vulkan® Support

May

- **RGA 2.8**
  - Support for AMD Radeon™ RX 7800/7700 XT
  - VGPR pressure GUI
  - AMD offline LLVM pipeline compiler

June

- **RGA 2.9**
  - Analyze pre-compiled Code Object binaries.
  - Analyze pre-compiled HIP binaries for the MI-200 architecture

July

- **RGA 2.0**
  - Redesigned Wavefront occupancy UI
  - Dark mode support
  - Raytracing pipeline thread divergence

August

- **RGA 2.9**
  - Analyze pre-compiled Code Object binaries.
  - Analyze pre-compiled HIP binaries for the MI-200 architecture

September

- **RMV 1.8**
  - Improvements to Resource Usage size timeline

October

- **RGA 2.9**
  - Analyze pre-compiled Code Object binaries.
  - Analyze pre-compiled HIP binaries for the MI-200 architecture

November

- **RG 1.0**
  - DirectX® 12 Support

December

- **RG 1.0**
  - DirectX® 12 Support

GDC 2024

- **RGP 1.16**
  - ISA disassembly view improvements
  - Quality of life improvements

- **RRA 1.3**
  - Ray Visualization feature
  - Persistent UI state

- **RMV 1.6**
  - Improved history resource table
  - Loading of RGD crash dump files
  - Resource name and implicit buffer fixes

- **RGD 1.0**
  - DirectX® 12 Support
Redesigned user interface offers

- Improved experience for new users
- Simplified user workflows for setting capture options
- Persistence of all settings across invocations of the panel
- Support for new features without increasing complexity
RADEON

GPU Profiler
RGP 1.16 WAVEFRONT OCCUPANCY LAYOUT

Previous layout

• UI controls above each row

• Legends below each row
RGP 2.0 WAVEFRONT OCCUPANCY LAYOUT

New layout

- UI controls and Legends are moved to the left of each row
- More vertical screen real estate allocated to the data views
RGP 2.0 WAVEFRONT OCCUPANCY LAYOUT

New layout

- The new left hand side panel can be hidden
RGP – CUSTOMIZE WAVEFRONT OCCUPANCY LAYOUT

New layout

• Individual rows can be hidden
RGP – CUSTOMIZE WAVEFRONT OCCUPANCY LAYOUT

New layout

- Individual rows can be hidden
- Raytracing counters hidden
**RGP – CUSTOMIZE WAVEFRONT OCCUPANCY LAYOUT**

**New layout**

- Individual rows can be hidden
- Raytracing counters hidden
- Hidden rows can be shown again
RGP – CUSTOMIZE WAVEFRONT OCCUPANCY LAYOUT

New layout

• The position of individual rows can be changed
New layout

- The position of individual rows can be changed
- Click and drag a view to reposition it
RGP – CUSTOMIZE WAVEFRONT OCCUPANCY LAYOUT

New layout

- The position of individual rows can be changed
- Click and drag a view to reposition it
- Drop it in the new position
New layout

• The default view can be restored
RGP – CUSTOMIZE WAVEFRONT OCCUPANCY LAYOUT

New layout
• The default view can be restored
RGP – DARK MODE

- RGP can be set to use Dark Mode or Light mode
- Or it can be set to follow the host operating system
RGP – DARK MODE

• RGP can be set to use Dark Mode or Light mode
• Or it can be set to follow the host operating system
• Use the new “Color Theme” setting on the “Themes and Colors” page
RGP: RAY TRACING THREAD DIVERGENCE

• Learn about thread divergence in your ray tracing pipelines
• RGP reports the average number of active lanes upon entry of each function in the ray tracing pipeline
RGP: RAY TRACING THREAD DIVERGENCE

- Learn about thread divergence in your ray tracing pipelines
- New column reports the average number of active lanes upon entry of each function in the ray tracing pipeline
- Histogram shows the distribution across all invocations
RGP: RAY TRACING THREAD DIVERGENCE – RDP SUPPORT

- “Enable shader instrumentation” checkbox in RDP
- May add extra overhead which can affect runtime performance
RGP: WORK GRAPHS

- RGP’s event lists show individual sub-dispatches
- Shows how the work is broken down during graph execution
RGP: WORK GRAPHS

- RGP’s event lists show individual sub-dispatches
- Shows how the work is broken down during graph execution
- Coloring by event shows which waves come from which graph sub-dispatches
- More info on GPUOpen: https://gpuopen.com/learn/rgp-work-graphs/
AMD RADEON™ GPU ANALYZER

New mode
• Binary Analysis
RGA BINARY ANALYSIS MODE

- Drag & drop any pre-compiled AMD GPU Code Object binary
- Unlike other RGA modes, here we start with a pre-compiled Code Object binary file
RGA BINARY ANALYSIS MODE

- Shaders and kernels appear on the left pane
RGA BINARY ANALYSIS MODE

- Shaders and kernels appear on the left pane.
- Binary gets disassembled.
RGA BINARY ANALYSIS MODE

- Shaders and kernels appear on the left pane
- Binary gets disassembled
- VGPR pressure visualized
RGA BINARY ANALYSIS MODE

- RGA will detect modified binaries and reload the contents automatically.
- You can continue to use your normal workflows to edit and compile binaries – RGA will always show the latest updates.
PREVIEW: RGP/RGA INTEROP
PREVIEW: RGP/RGA INTEROP

- Any pipeline available in RGP can be automatically extracted and analyzed in RGA
- Use the new “Analyze pipeline in Radeon GPU Analyzer” menu item
PREVIEW: RGP/RGA INTEROP

- The selected pipeline will be extracted
- RGA will be launched, and the extracted pipeline will be loaded and analyzed
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

• Currently RGA works with complete pipelines
  • For graphics, that means you would need at least the following:
    • Vertex shader
    • Pixel shader
    • Graphics pipeline state object
    • Root signature

• What if you don't have a complete pipeline?
  • You are missing one of the shaders
  • Missing the graphics pipeline state object
  • Missing the root signature

• RGA will have a new mode that can autogenerate the missing pieces of the pipeline!
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

- Let's consider an example:
- You have a single pixel shader, but do not have a vertex shader, a root signature or a graphics pipeline state object.
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

- Compiling as usual in DX12 mode, providing only the pixel shader as an input.
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

- The vertex shader is auto generated

```c
// Auto-generated with Radeon GPU Analyzer (RGA).

struct VsInput
{
    float4 attribute0: POSITION0;
};

struct VsOutput
{
    float4 attribute0: SV_POSITION;
    float2 attribute1: TEXCOORD0;
};

void main(VsInput input, out VsOutput output)
{
    float4 result = float4(0.0, 0.0, 0.0, 0.0);
    result += float4(input.attribute0.xyzw);
    output.attribute0 = float4(result.xyzw);
    output.attribute1 = float2(result.xy);
}
```
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

• The root signature is auto generated

```c
#define RGA_ROOT_SIGNATURE 
"RootFlags( ALLOW_INPUTAssembler_INPUT_LAYOUT | DENY_HULL_SHADER_ROOT_ACCESS " 
"| DENY_DOMAIN_SHADER_ROOT_ACCESS | DENY_GEOMETRY_SHADER_ROOT_ACCESS ), " 
"DescriptorTable(Sampler(s0), visibility=SHADER_VISIBILITY_PIXEL), " 
"DescriptorTable(SRV(t0), visibility=SHADER_VISIBILITY_PIXEL)"
```
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

- The graphics pipeline state object is auto generated.
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

- RGA will output the pixel shader’s disassembly
PREVIEW: RGA DX12 SINGLE SHADER COMPILATION

PS.hlsl → Reflection → VS.hlsl → .gpso → AMD Compiler → Output files

- ISA disassembly
- VGPR analysis
- ...

RGA

Root signature
AMD RADEON
Raytracing Analyzer
RRA 1.3 - NEW RAY FEATURES

- New ability to capture raytracing dispatches
- Captured through RDP by enabling "Collect ray dispatch data" option
- A buffer size option is needed to ensure all the data can be captured
RRA 1.3 - NEW RAY FEATURES

- New ability to capture raytracing dispatches
- Captured through RDP by enabling "Collect ray dispatch data" option
- A buffer size option is needed to ensure all the data can be captured
RRA 1.3 - DISPATCH

- Dispatch can be visualized into heatmap of various types if the dispatch shape is 2D or 3D
- Can also be mapped from 1D dispatches, must not be sparse
- Allows for exact traversal cost analysis since all the shadow and reflection rays will be included
- Select a "pixel" to inspect it
RRA 1.3 - RAY INSPECTOR

- View each ray for a given dispatch index
- See the cost and arguments for each ray
RRA 1.4

- New ray directions feature on the ray dispatches
- Quickly find and identify shadow and reflection areas
- Bugfixes and quality of life updates
PREVIEW: RRA 1.5

- New ray hierarchy in Ray Inspector
- Quickly verify sampling rate and recursive calls
RMV TIMELINE IMPROVEMENTS

- Better memory visualization
- Unbound memory is now shown on timeline
- Usage sizes for aliased resources properly calculated
- Implicit resources filtered from calculations
RMV RESOURCE OVERVIEW IMPROVEMENTS

- Usage sizes for aliased resources properly calculated and shown in tooltip
- Named allocations
- Improved size range filter
OTHER RMV IMPROVEMENTS

- Heap overview pane contains additional information
- Device configuration pane shows system memory and driver info
- Support added for file format which supports compression
- New time unit format
- Expanded history resource columns
AMD RADEON™ GPU DETECTIVE (RGD)

- Newest member of AMD Radeon™ Developer Tool Suite (https://gpuopen.com/tools/)

Overview:

- Tool for post-mortem analysis of GPU crashes
- Sets driver to Crash Analysis mode before reproducing crash
- Developers capture AMD GPU Crash Dump files upon crash
- Produces concise crash analysis report in Text/JSON formats
- Report helps narrow down the search for the crash root cause
AMD RADEON™ GPU DETECTIVE (RGD)

- Newest member of AMD Radeon™ Developer Tool Suite ([https://gpuopen.com/tools/](https://gpuopen.com/tools/))

Requirements:

- OS: Windows 10 or 11
- GPU: AMD Radeon™ RX 7000 Series (RDNA™ 3) or AMD Radeon™ RX 6000 Series (RDNA™ 2)
- Driver: AMD Software: Adrenalin Edition 23.12.1 or newer
- Graphics API used by the crashing application: **DirectX 12** or **Vulkan**
UNVEILING THE TOOL: WORKFLOW AND FUNCTIONAL INSIGHTS

1. Crashing application
2. Crash dump file
3. Crash analysis summary

Auto-generated (no action required)
UNVEILING THE TOOL: WORKFLOW AND FUNCTIONAL INSIGHTS

- Curious about the newest addition to our tool suite?
- Stay tuned for the next presentation

POST-MORTEM GPU CRASH ANALYSIS WITH AMD RADEON™ GPU DETECTIVE (RGD)

ADAM SAWICKI (AMD)
AMIT MULAY (AMD)
MARCO BOUTERSE (NIXXES SOFTWARE)
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

© 2024 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 Linus Torvalds. OpenCL is a trademark of Apple, Inc. used by permission from The Khronos Group. LLVM is a trademark of LLVM Foundation. 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.