



Introduction to OmniTrace

Gina Sitaraman, Suyash Tandon, George Markomanolis,
Jonathan Madsen, Austin Ellis, Bob Robey

AMD Profiling Training, Pawsey
April 26, 2023

AMD
together we advance_

[Public]

Background – AMD Profilers

ROC-profiler (rocprof)

Hardware Counters	Raw collection of GPU counters and traces	
	Counter collection with user input files	Counter results printed to a CSV

Traces and timelines	Trace collection support for			
	CPU copy	HIP API	HSA API	GPU Kernels

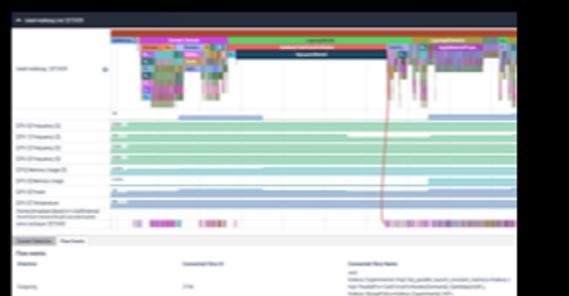
Visualisation	Traces visualized with Perfetto			

Name	Cells	Total	Avg	Percentile
hipAllocMemory	39	3.22e+09	3.27e+09	44.14872
hipAllocHostMemory	389	1.25e+09	3.22e+08	2.22e+08
hipFreeMemory	87	7.92e+09	8.92e+08	12.86953
hipEventCreate	9	5.62e+09	6.23e+08	7.415398
hipEventDestroy	28	1.32e+09	47.00e+08	1.805115
hipDeviceSynchronize	17	1.09e+09	65.63e+08	1.439504
hipHostFree	41	8.10e+08	1.979e+08	5.113310
hipLaunchKernel	1850	1.60e+09	8.67e+08	1.00e+09
hipLaunchKernelCuda	2	4.63e+08	2.32e+08	0.063625
hipLaunchCooperative	2	1.89e+08	94.93e+07	0.029584
hipLaunchKernelDestroy	2	2.15e+08	79.61e+07	0.029828
hipFree	38	2.09e+08	5.37e+07	0.013384
hipLaunchRecord	350	2.52e+08	7.48e+07	0.003457
hipLaunchConfigurable	30	1.48e+08	4.93e+07	0.003007
hipLaunchCooperativeConfigurable	388	1.00e+08	2.67e+07	0.003114
hipPushCallConfigurable	1856	2.24e+07	1.20e+06	0.003008
hipLaunchConfigurable	3494	1.00e+08	6.7e+07	0.003138
hipEventCreate	330	7.66e+07	2.32e+07	0.003035
hipEventDestroy	330	6.60e+07	1.98e+07	0.003035
hipLaunchConfigurable	47	5.18e+07	1.10e+07	0.003035
hipLaunchService	84	3.00e+07	3.61e+07	0.003035
hipService	5	4.01e+07	4.02e+07	5.300e-07
hipServiceCount	5	2.00e+07	2.00e+07	5.021e-07

Omnitrace

Trace collection	Comprehensive trace collection			
	CPU		GPU	
	CPU copy		HSA API	

Supports	CPU copy	HIP API	HSA API	GPU Kernels
	OpenMP®	MPI	Kokkos	p-threads



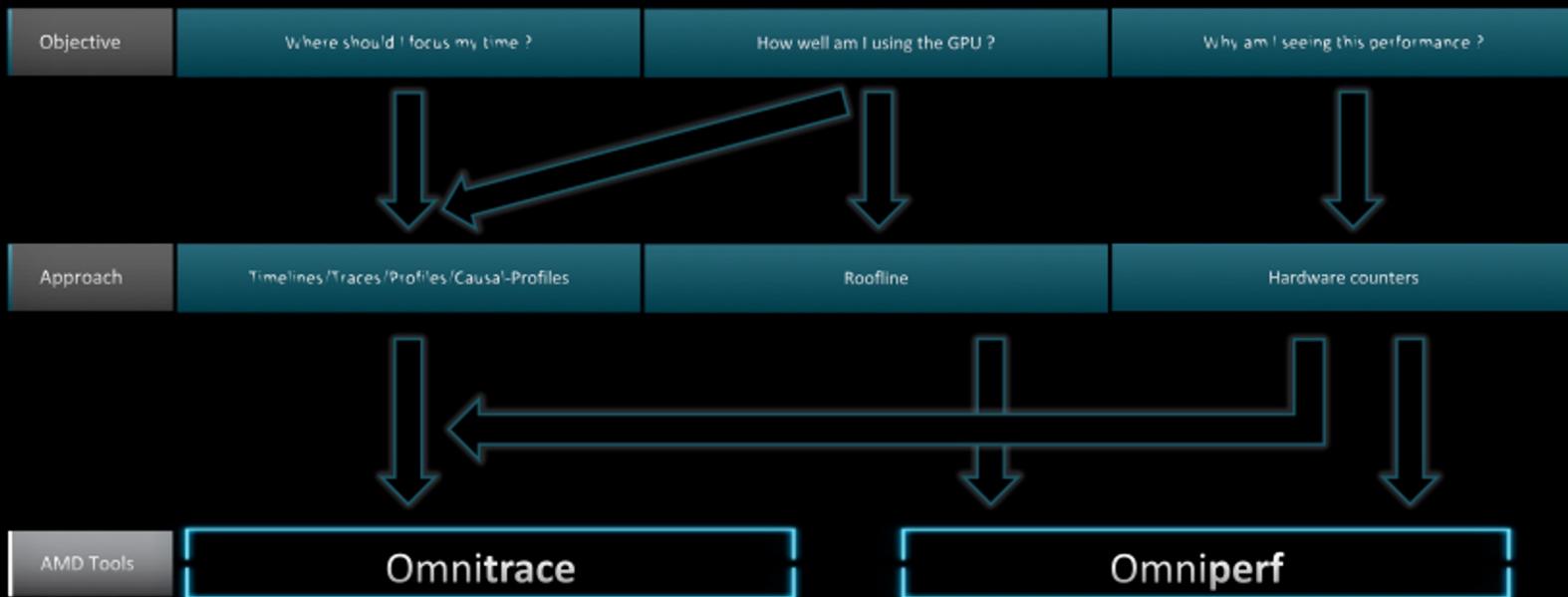
Omniperf

Performance Analysis	Automated collection of hardware counters			
	Analysis		Visualisation	
	Speed of Light		Memory chart	

Supports	Speed of Light	Memory chart	Rooflines	Kernel comparison
With Grafana or standalone GUI				



Background – AMD Profilers



[Public]

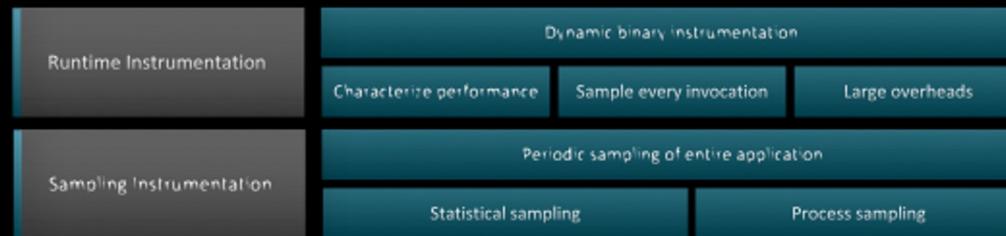
Omnitrace: Application Profiling, Tracing, and Analysis

AMD Research Tool	Repository: https://github.com/AMDRResearch/omnitrace					
	 Not part of ROCm stack					
Language Support	C/C++	Fortran	Python	OpenCL™		
Data Collection Modes	Dynamic instrumentation		Statistical/process sampling			
Data Analysis	High-level summary		Comprehensive trace			
Parallelism Support	MPI	OpenMP®	Pthreads	HIP	HSA	Kokkos
GPU Metrics	HW counters	HSA API	HIP API	HIP trace	HSA trace	Memory & thermal
CPU Metrics	HW counters	Timing metrics	Memory access	Network	I/O	more...

Refer to [current documentation](#) for recent updates

[Public]

Omnitrace instrumentation Modes



Basic command-line syntax:

```
$ omnitrace [omnitrace-options] -- <CMD> <ARGS>
```

For more information or help use -h/-help/? flags:

```
$ omnitrace -h
```

Can also execute on systems using a job scheduler. For example, with SLURM, an interactive session can be used as:

```
$ srun [options] omnitrace [omnitrace-options] -- <CMD> <ARGS>
```

For problems, create an issue here: <https://github.com/AMDRResearch/omnitrace/issues>

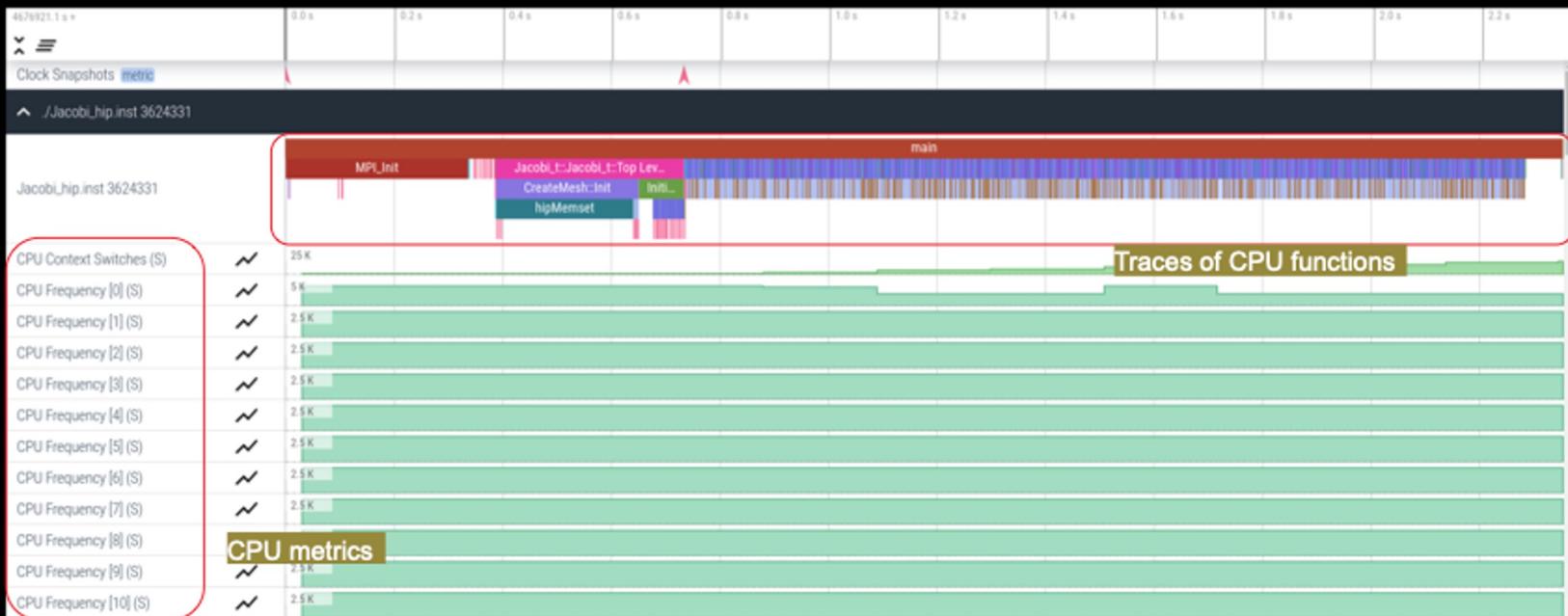
Documentation: <https://amdrresearch.github.io/omnitrace/>

[Public]

Visualizing Trace

Use Perfetto

Copy perfetto-trace-0.proto to your laptop, go to <https://ui.perfetto.dev/>, Click "Open trace file", select perfetto-trace-0.proto

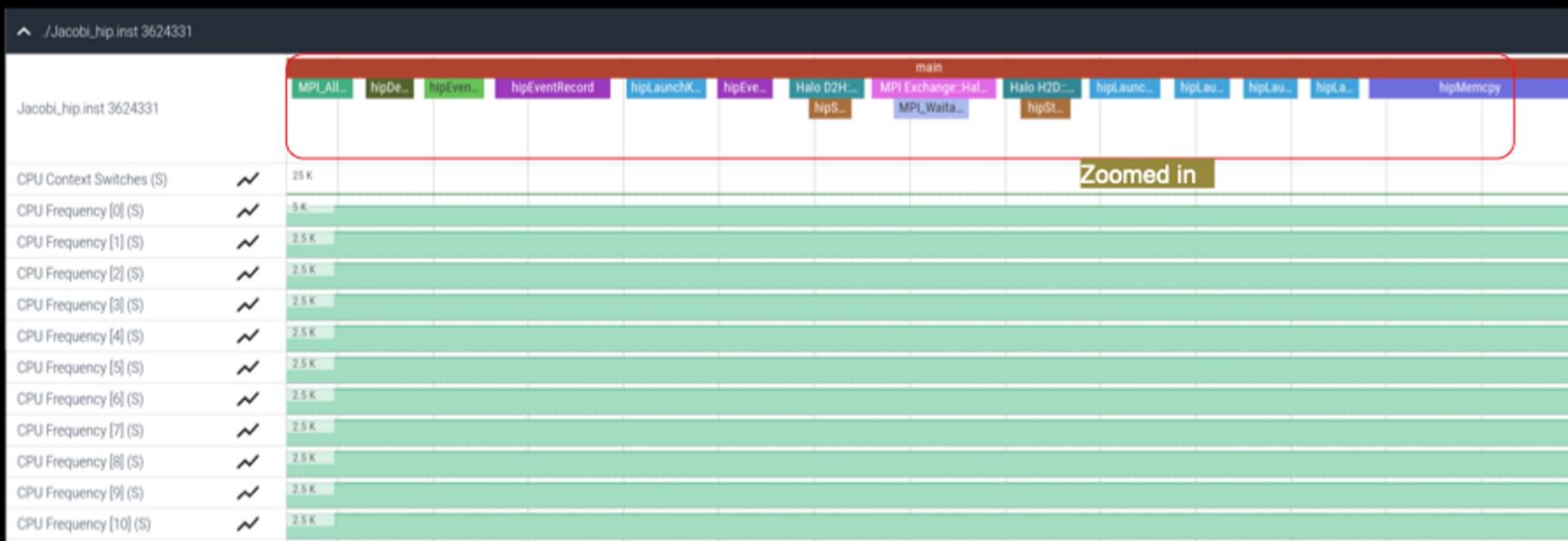


[Public]

Visualizing Trace

Use Perfetto

Zoom in to investigate regions of interest

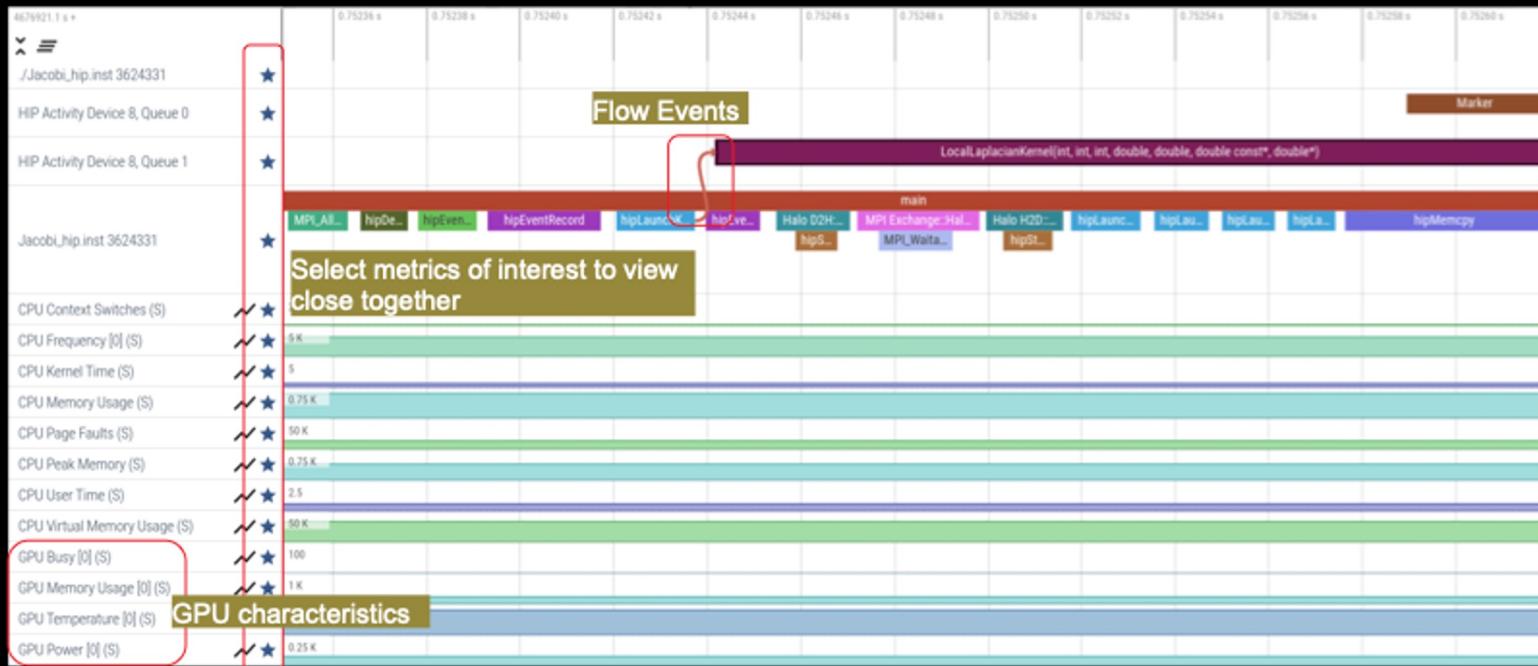


[Public]

Visualizing Trace

Use Perfetto

Zoom in to investigate regions of interest



Commonly Used GPU Counters

VALUUtilization	The percentage of ALUs active in a wave. Low VALUUtilization is likely due to high divergence or a poorly sized grid
VALUBusy	The percentage of GPUTime vector ALU instructions are processed. Can be thought of as something like compute utilization
FetchSize	The total kilobytes fetched from global memory
WriteSize	The total kilobytes written to global memory
L2CacheHit	The percentage of fetch, write, atomic, and other instructions that hit the data in L2 cache
MemUnitBusy	The percentage of GPUTime the memory unit is active. The result includes the stall time
MemUnitStalled	The percentage of GPUTime the memory unit is stalled
WriteUnitStalled	The percentage of GPUTime the write unit is stalled

Modify config file

Create a config file in \$HOME:

```
$ omnitrace-avail -G $HOME/.omnitrace.cfg
```

Modify the config file \$HOME/.omnitrace.cfg to add desired metrics and for concerned GPU#ID:

```
...
OMNITRACE_ROCM_EVENTS = GPUBusy:device=0,
Wavefronts:device=0, MemUnitBusy:device=0
...
```

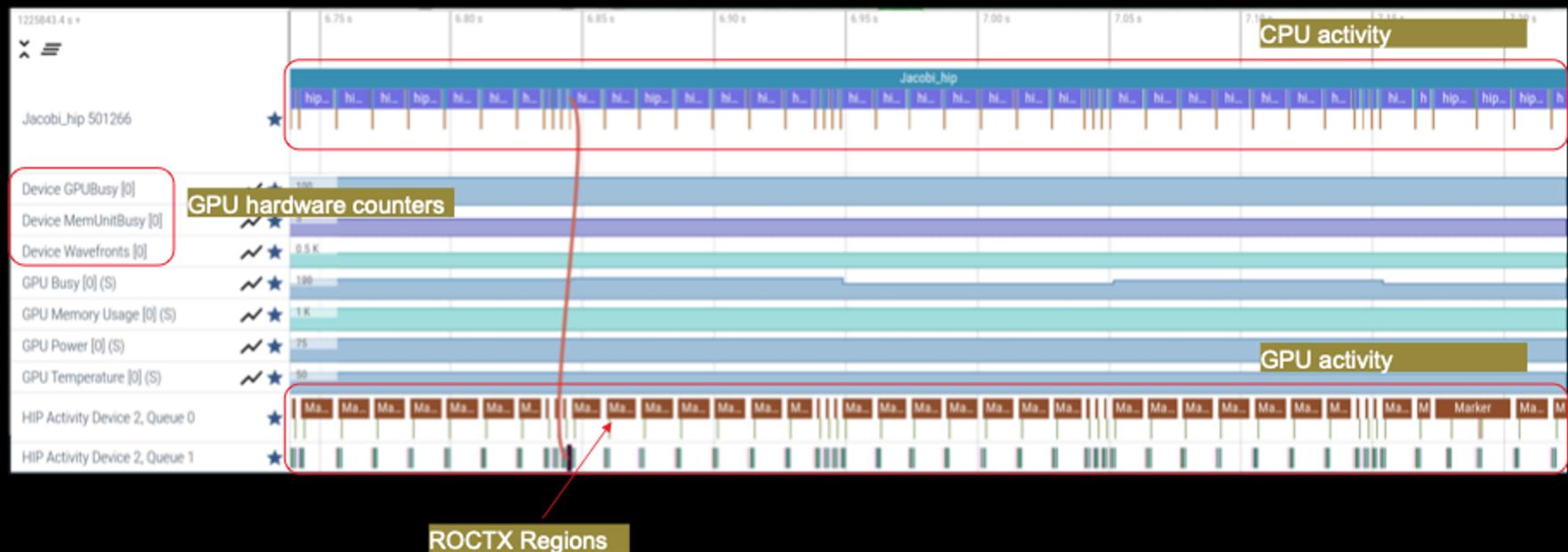
To profile desired metrics for all participating GPUs:

```
...
OMNITRACE_ROCM_EVENTS = GPUBusy, Wavefronts,
MemUnitBusy
...
```

Full list at: <https://github.com/ROCM-Developer-Tools/rocprofiler/blob/amd-master/test/tool/metrics.xml>

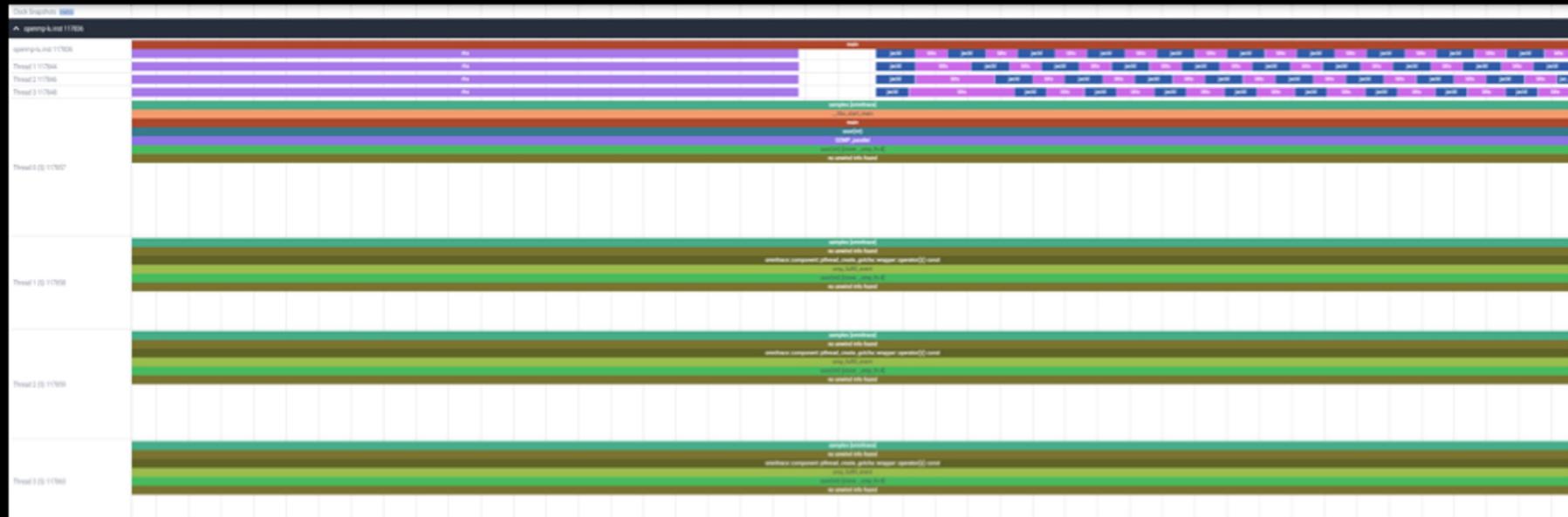
[Public]

Visualization with Hardware Counters





OpenMP® Visualization



Summary

- OmniTrace is a powerful tool to understand CPU + GPU activity
 - Ideal for an initial look at how an application runs
- Leverages several other tools and combines their data into a comprehensive output file
 - Some tools used are AMD uProf, rocprof, rocm-smi, roctracer, perf, etc.
- Easy to visualize traces in Perfetto
- Includes several features:
 - Dynamic Instrumentation either at Runtime or using Binary Rewrite
 - Statistical Sampling for call-stack info
 - Process sampling, monitoring of system metrics during application run
 - Causal Profiling
 - Critical Path Tracing

[Public]

DISCLAIMERS AND ATTRIBUTIONS

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

THIS INFORMATION IS PROVIDED 'AS IS.' AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

© 2023 Advanced Micro Devices, Inc. All rights reserved.

AMD, the AMD Arrow logo, Radeon™, Instinct™, EPYC, Infinity Fabric, ROCm™, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.