

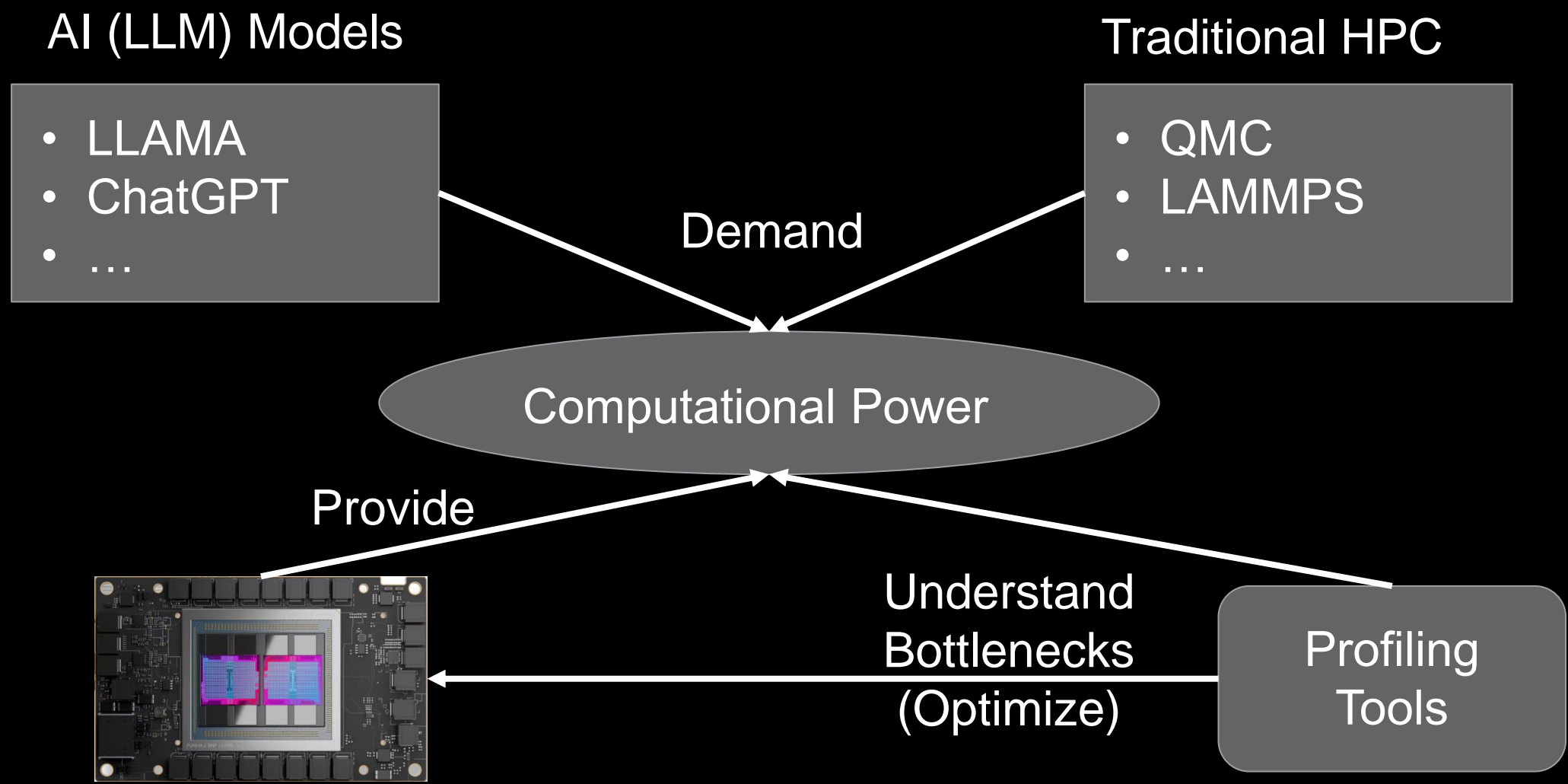
The Road to PC Sampling on AMD Instinct™ MI200 GPU Series

Vladimir Indic
Timour Paltashev

15th International Parallel Tools Workshop
September 19-20, 2024

AMD 
together we advance_

Motivation



i.e., AMD Instinct™ MI250X

Profiling Tools on AMD GPUs

- ROCProf (V1/V3), HPCToolkit, Score-P, TAU, Caliper, PAPI, etc.
- Coarse-grained information about GPU kernel execution
- Host-side API tracing:
 - When kernel was launched
 - How long it executed
 - **No insights within kernel execution**
- (GPU) Hardware performance counter profiling:
 - Glance at potential cause of why kernel underutilize resources
 - **Requires kernel serialization**
 - Might require kernel/application replays

Goals to Achieve

- Detailed (as possible) insights within the kernel execution under reasonable overhead
- Monitoring execution of concurrent kernels
 - Avoid kernel serialization due to high overhead
- Avoiding kernel/application replays

Profiling Methods Borrowed from CPU Realm - Instrumentation

- Instrumentation modifies the original application source code or binary by injecting instrumenting code
- Pros:
 - Great level of details
 - Highly configurable
- Cons:
 - Interfering with application execution that can lead to a high overhead
 - Modifying the code might alter the application behavior which might result in an imprecise trace/profile

Profiling Methods Borrowed from CPU Realm – Statistical Sampling

- CPU sampling: probes the target program's call stack at regular intervals using OS interrupts*
- Results: histogram of samples that statistically approximates program execution
- Pros:
 - No changes of application source code or binary required
 - Reasonably low overhead (1-5%)
- Cons:
 - Lower level of details compared to the instrumentation
 - No time-based information

* [https://en.wikipedia.org/wiki/Profiling_\(computer_programming\)](https://en.wikipedia.org/wiki/Profiling_(computer_programming))

PC Sampling on AMD GPUs - Idea

- Periodically, choose an active wavefront (in a round robin manner), snapshot its state
 - At least program counter (PC) and the current timestamp
 - Other information if available
- The process takes place on every compute unit simultaneously
 - Which makes it device-wide across all shader engines
- Generates a histogram of samples that statistically approximates kernel execution
 - Flat samples due to lack of call stack on a GPU device
 - Call stack can be approximated*

* Zhou, K., Adhianto, L., Anderson, J., Cherian, A., Grubisic, D., Krentel, M., Liu, Y., Meng, X. and Mellor-Crummey, J., 2021. Measurement and analysis of GPU-accelerated applications with HPCToolkit. *Parallel Computing*, 108, p.102837.

PC Sampling on Other Vendors' Architectures (1)

- Nvidia GPUs contains hardware support that can provide information about whether sampled warp is issuing current instruction¹
 - If not, what's the stall reason
- Very useful for understanding stalls inside running kernels
- Limitations:
 - One process at a time can use GPU while PC sampling is enabled
 - Multiple MPI ranks cannot share the same GPU device under PC sampling²
 - To attribute samples to different launch instances of the same kernel (invocations), instances of the same kernel must not run concurrently^{3,4}
 - Important for the calling context sensitive analysis

¹ https://docs.nvidia.com/cupti/api/group_CUPTI_PCSAMPLING_API.html

² <https://journals.sagepub.com/doi/pdf/10.1177/10943420241277839>

³ <https://dl.acm.org/doi/pdf/10.1145/3524059.3532388>

⁴ https://docs.nvidia.com/cupti/api/structCUpti_PCSamplingPCData.html#cupti-pcsamplingpcdata

PC Sampling on Other Vendors' Architectures (2)

- Similarly, Intel GPUs provide stall reasons information^{1,2}
- To attribute samples to the kernel, one must either:
 - Serialized kernels or
 - Approximate the attribution of samples to a kernel instance using available information such as the kernel instance execution timestamps, sample time stamps, and/or kernel instance duration³

¹ <https://www.intel.com/content/www/us/en/docs/oneapi/optimization-guide-gpu/2024-1/vtune-stall-sampling.html>

² Zhiqiang Ma. Unified Tracing and Profiling Tool, Argonne A21 Community Tools Telecon, November 27, 2023.

³ John Mellor-Crummey, Personal Communication

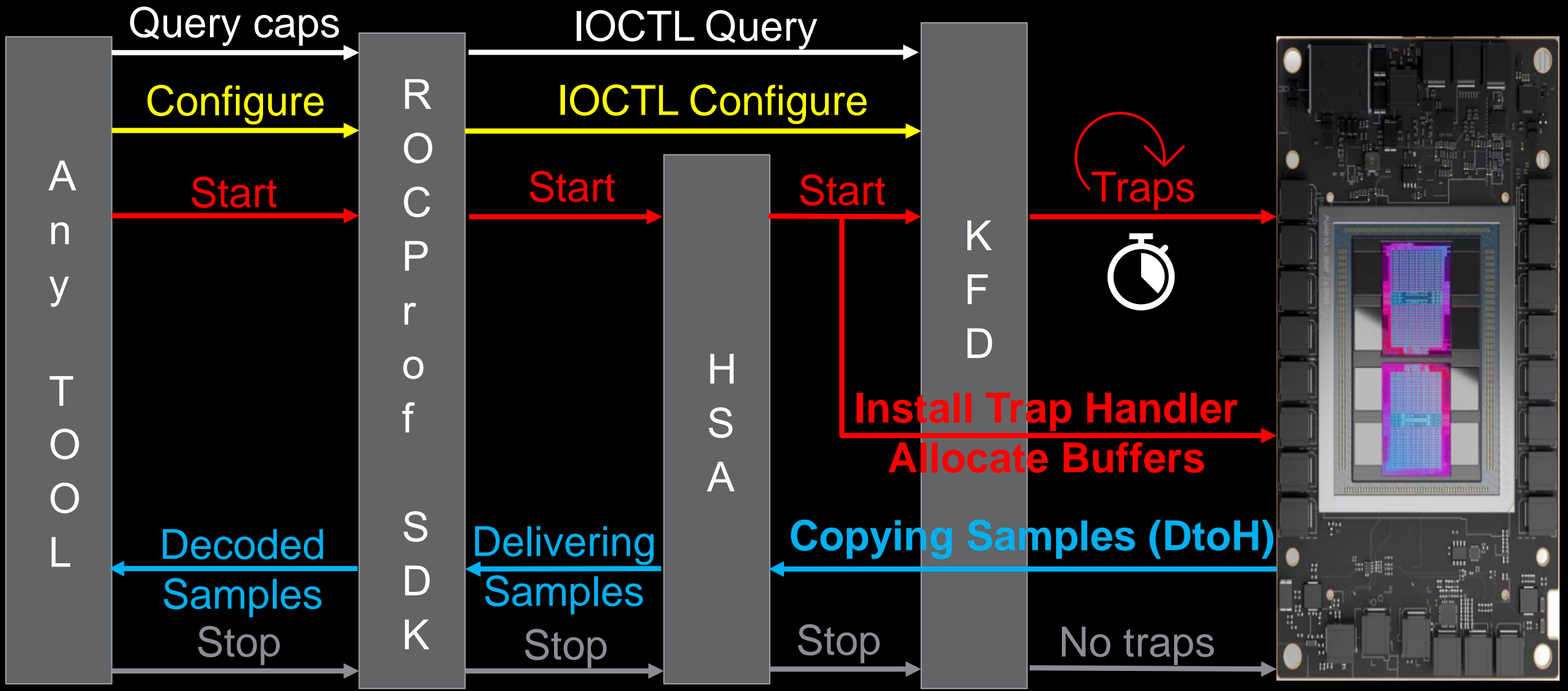
PC Sampling on AMD Instinct™ MI200 series

- No hardware support for PC sampling => using software to probe wavefronts
- Basic concepts:
 - Every X us, broadcast an instruction to all CUs for trapping active wavefronts with certain identifier (SIMD + wavefront slot)
 - Identifier is determined in a round-robin manner
 - Trapped wavefronts executes an alternative path called **trap handler**
 - Trap handler code captures wavefront's state in a GPU buffer
 - to form a PC sample
 - Other wavefronts remains intact
 - Once buffer is full, copy all PC samples to host

Glossary

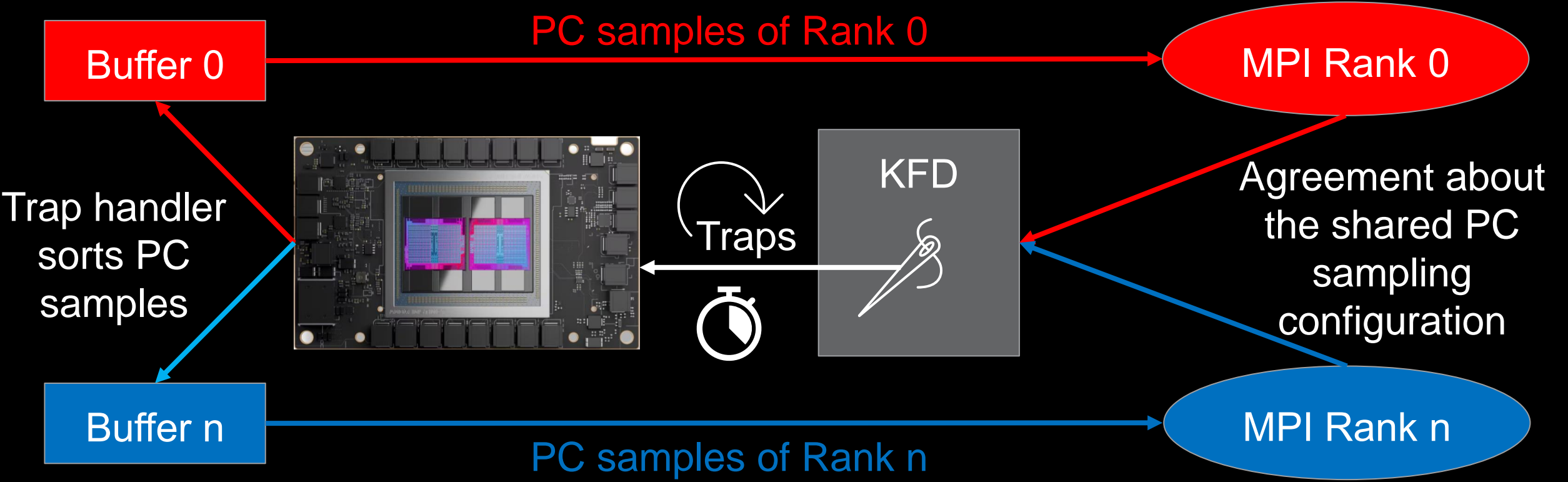
- ROCProfiler-SDK
 - Brand new profiling library that provides an API for: tracing, counter collection, PC sampling, etc.
 - Used by ROCProf(V3) and other 3rd Party Tools
- HSA Runtime
 - Operates with GPU agents in user-level mode:
 - Memory copies, kernel dispatches, etc.
 - HIP, OpenMP, and OpenCL rely on HSA runtime
- AMDGPU driver (KFD)
 - Manages GPU agents in kernel mode
 - Direct access to GPU HW
 - Implements IOCTL operations for communicating with GPU agents
 - Used by ROCProfiler-SDK, HSA runtime, etc.

PC Sampling Workflow on AMD Instinct™ MI200 Series



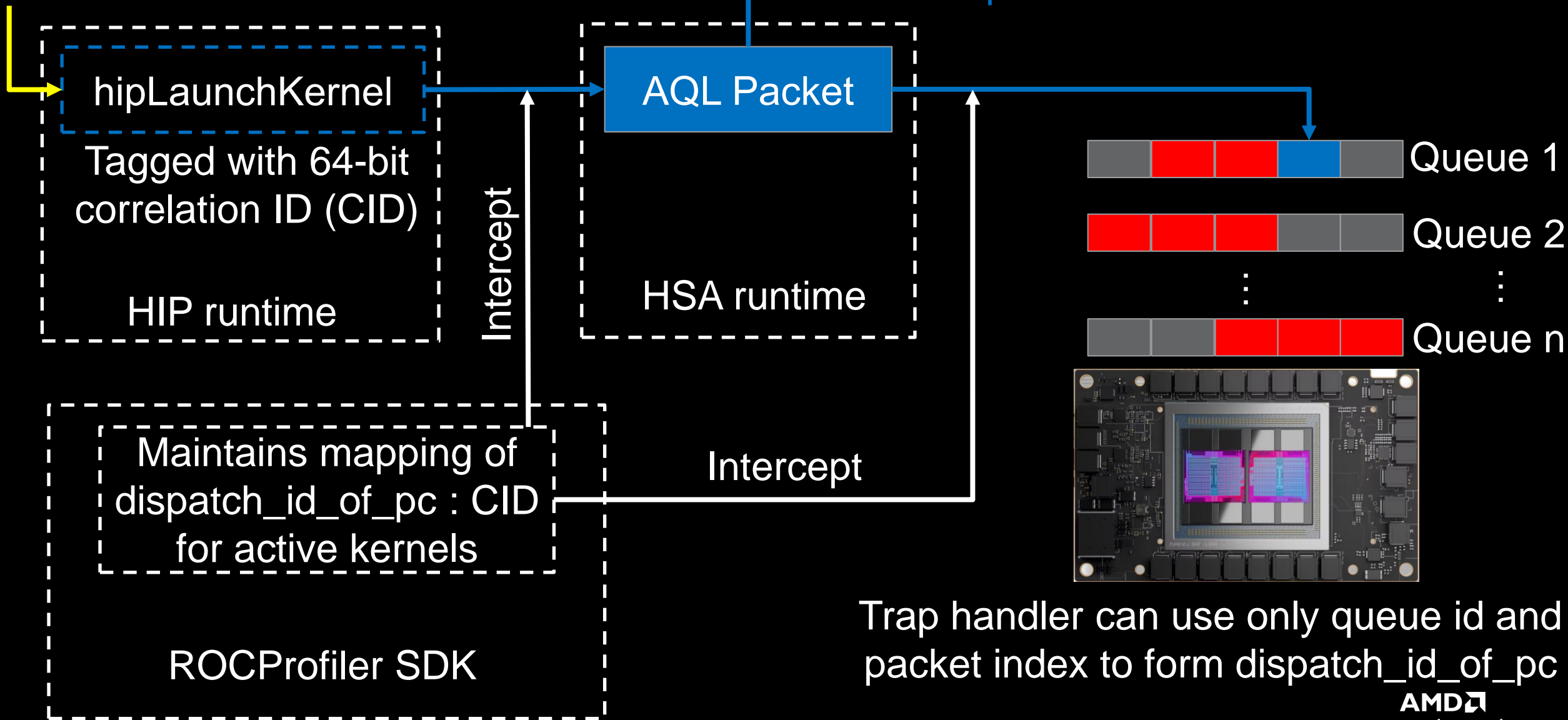
Multi-Processes (MPI) Support

- AMD Instinct™ MI2xx GPU is a shared resource
 - But it can support at most one PC sampling configuration at a time



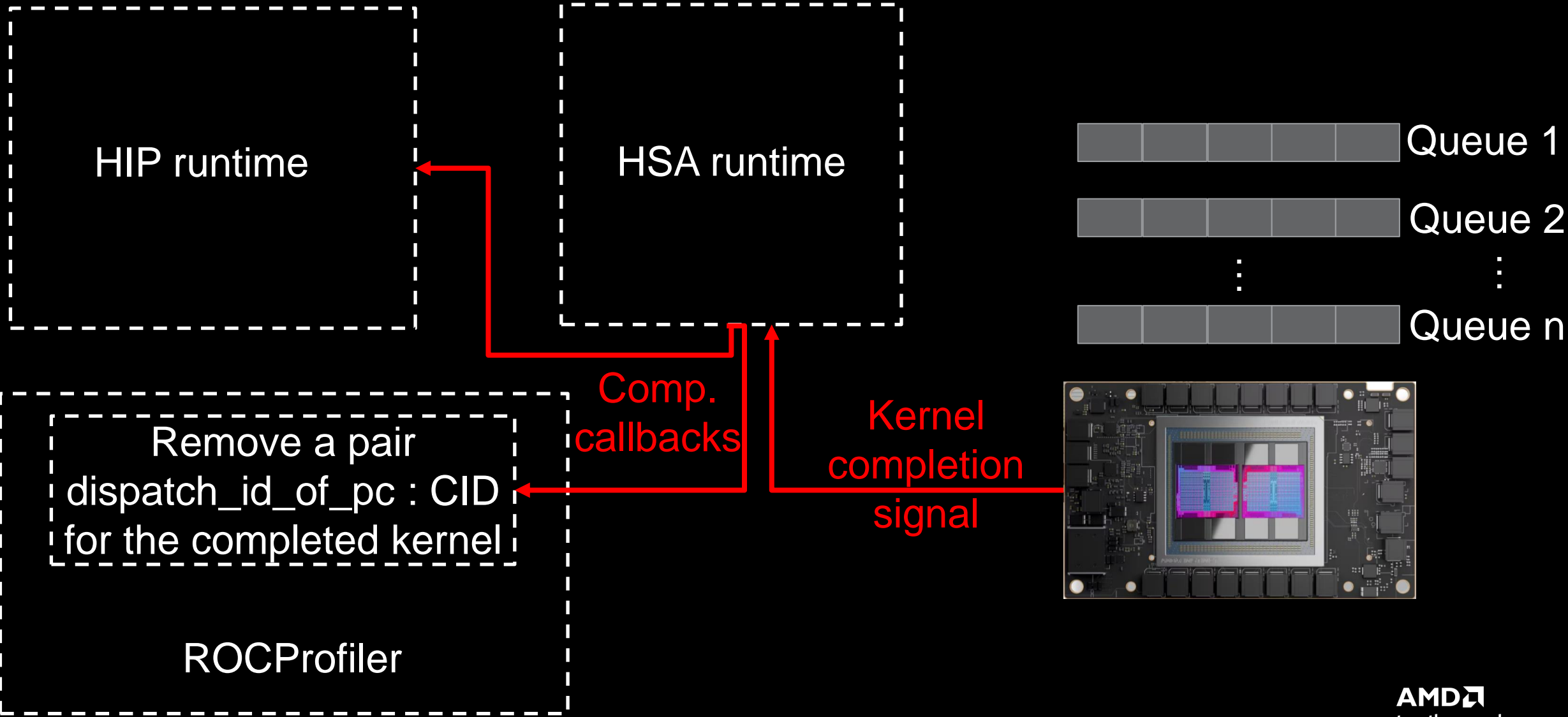
Correlating PC Samples to Kernel Dispatches (1)

HIP Application



Trap handler can use only queue id and packet index to form `dispatch_id_of_pc`

Correlating PC Samples to Kernel Dispatches (2)



Remaining Items

- Practical usage of PC sampling for profiling real world workload kernels on AMD Instinct™ MI200 series

Conclusion

- Importance of PC sampling profiling method
- Design concepts of PC sampling implementation on AMD Instinct™ MI200 GPU series
 - Delivering samples to originating processes (MPI ranks)
 - Attributing samples to different instances (launches) of the same kernel useful for calling context sensitive analysis
- Validation
 - ROCProfiler Team (internally)
 - HPCToolkit Team (Rice University, Houston, TX)
 - Other teams/tools are more than welcome! 😊

Copyright and disclaimer

- ▶ ©2024 Advanced Micro Devices, Inc. All rights reserved.
- ▶ AMD, the AMD Arrow logo, CDNA, EPYC, Instinct, Infinity Fabric, ROCm, Ryzen, 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.
- ▶ The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate releases, for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.
- ▶ 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

AMD 