ROCm Tools Ecosystem

Vladimir Indic (AMD)

Motivation

- Collect useful feedback from tooling folks
- Better understanding of what tools need
- Deliver stable ROCm stack that is foundation for delivering remarkable 3rd party tools

Support third-party performance tools

- **PC sampling on AMD GPUs**
  - How will HIP/OpenMP/PSTL C++ cooperate
- **When profiling how to attribute metrics to**
  - Specific GPU on a system, GPU Model, and/or GPU binary
- **Need a clear contract between a tool and ROCm**
  - The right way/order of using API functions

Support for XML ISA

- **Provides information sufficient for instruction decoding**
- **Need HW description**
  - Provide a description of registers in AMD architecture as ARM does
  - Need correct instruction semantic information for dataflow analysis (WIP)
  - Right now, DynInst infers architecture description from things referenced in the instructions

Support control of running kernels under program control

- **Who?**
  - **Gdb**
  - **Dyninst (ProcControlAPI)**
- **What?**
  - Intercept and control launch of GPU kernels
  - Stop a kernel execution at any point
    - Inspect state in the GPU
  - Be able to configure, start, stop and inspect performance counters

Problems with ROCm

- **Need for a stable API** (hopefully V2)
- **Need to establish the internal processes** (e.g., bug reporting, regression testing, ABI testing versus lib versions, etc,)
- **Lack of transparency**
  - Before any major change, inform the tooling folks what to expect
  - Items in internal changelog should be visible in public Release notes
  - Why is ATT hidden, what information does it provide
- **Lack of documentation and samples**
- Other **technical bugs/questions**