AMD GPU servers

- Overview
- Hardware info
    - Architecture specs
- Resources
    - ROCm GPU-enabling framework
    - Training Guides
    - Command-line diagnostics
- Sharing Resources

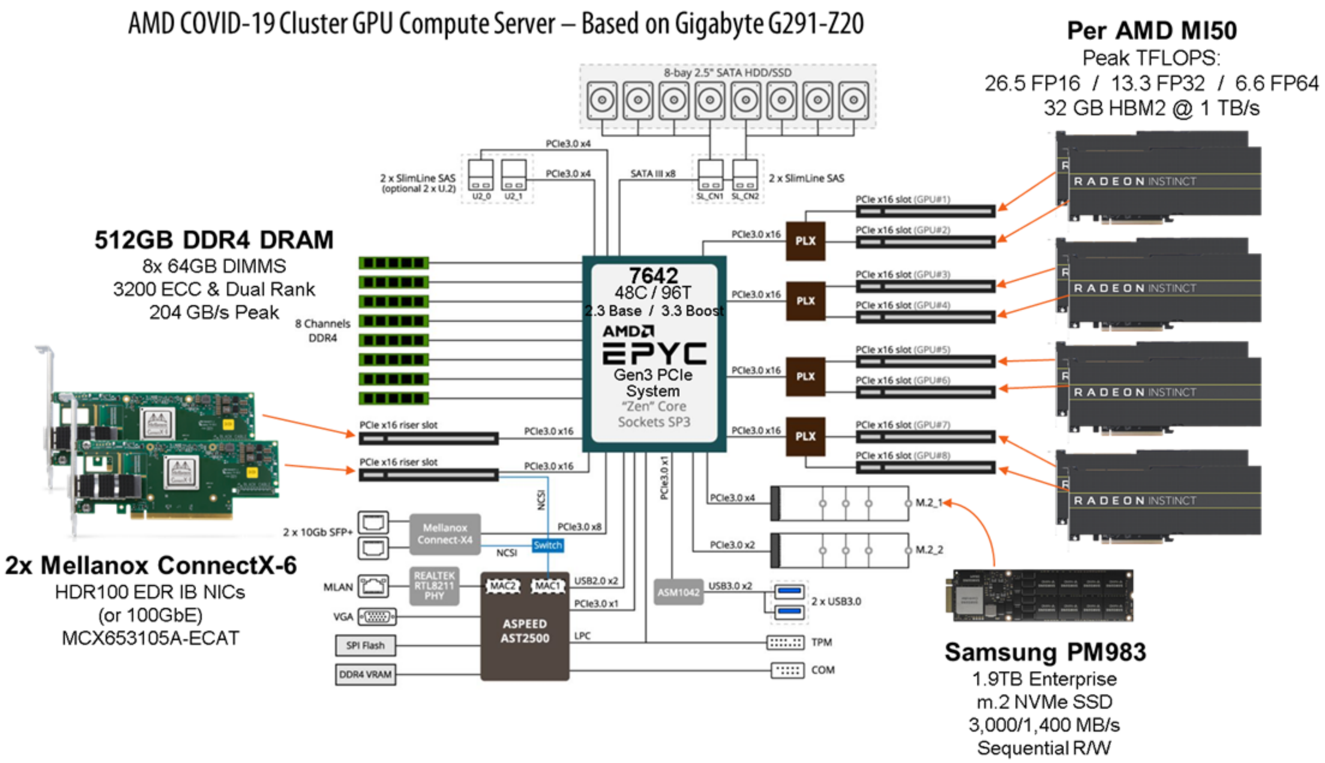
Overview

The Hopefog and Livestong PODs both have two AMD GPU servers, which enable powerful Machine Learning (ML) workflows.

Hardware info

Architecture specs

- 2U, Gigabyte G291 enclosures https://www.gigabyte.com/GPU-Server/G291-Z20-rev-100#ov
- 8 AMD Radeon Instinct MI50 GPUs (https://www.amd.com/en/products/professional-graphics/instinct-mi50)
    - each with 32 GB HBM2 memory
- 48-core/96-hyperthread EPYC 7642 CPU
- 512 GB RAM
- 1.9 TB Samsung NVMe SSD



Resources

ROCm GPU-enabling framework

Best starting places:

- **ROCm Video series**
    - https://community.amd.com/t5/instinct-accelerators-blog/rocm-open-software-ecosystem-for-accelerated-compute/ba-p/418720
    - Especially the **Introduction to AMD GPU Hardware**: Link

        - Provides hardware background and terminology used throughout other guides
    - Also
        - **GPU Programming Concepts**
            - Part 1 - HIP framework (like NVIDIA CUDA): Link
            - Part 2 - Device management, synchronization, MPI programming: Link
            - Part 3 - Device code, shared memory & thread synchronization: Link
        - **GPU Programming Software** (compilers, libraries & tools): Link
- **AMD ROCm resources Learning Center**: https://developer.amd.com/resources/rocm-resources/rocm-learning-center/
    - Especially:
        - Introduction to ROCm (Video, PDF)
        - Introduction to HIP (Video, PDF)
        - Introduction to Deep Learning on ROCm (Video, PDF)

Training Guides

1. Introduction_to_AMD_7002_processor.pdf
2. Radeon_Instinct_HPC_Training_2020.pdf
3. Radeon_Instinct_ML_Training_2020.pdf

Command-line diagnostics

- GPU usage: **rocm-smi**
- CPU and GPU details: **rocminfo**
- What ROCm modules are installed: **dpkg -l | grep rocm**
- GPU GPU/CPU communication bandwidth test
    - between GPU2 and CPU: **rocm-bandwidth-test -b2,0**
    - between GPU3 and GPU4: **rocm-bandwidth-test -b3,4**

Sharing Resources

Since there's no batch system on BRCF POD compute servers, it is important for users to monitor their resource usage and that of other users in order to share resources appropriately.

- Use **top** to monitor running tasks (or **top -i** to exclude idle processes)
    - commands while top is running include:
    - **M** - sort task list by memory usage
    - **P** - sort task list by processor usage
    - **N** - sort task list by process ID (PID)
    - **T** - sort task list by run time
    - **1** - show usage of each individual hyperthread
        - they're called "CPUs" but are really hyperthreads
        - this list can be long; non-interactive **mpstat** may be preferred
- Use **mpstat** to monitor overall CPU usage
    - **mpstat -P ALL** to see usage for all hyperthreads
    - **mpstat -P 0** to see specific hyperthread usage
- Use **free -g** to monitor overall RAM memory and swap space usage (in GB)
- Use **rocm-smi** to see GPU usage