Summary

- Perlmutter is a heterogeneous CPU+GPU system designed to accelerate the diverse data-centric and computational workflows for thousands of NERSC users.
- First phase of Perlmutter with all ~6000 NVIDIA A100 GPUs has been delivered in NERSC’s data center and is undergoing integration and testing.
- The system will support a wide range of programming languages and models, ensuring that its broad workload will be able to use its GPUs effectively.
NERSC mission
NERSC is the mission computing facility for the DOE Office of Science

8,000 Users
800 Projects
700 Codes
2000 NERSC citations per year

Simulations at scale

Data analysis support for DOE’s experimental and observational facilities

Photo Credit: CAMERA
NERSC systems roadmap

NERSC-7: Edison
2.5 PF
Multi-core CPU
3 MW

NERSC-8: Cori
30 PF
Manycore CPU
4 MW

NERSC-9: Perlmutter
3-4x Cori
CPU and GPU nodes
>5 MW

NERSC-10 ExaSystem
~20 MW

2013
2016
2021
2024
Perlmutter hardware
Hardware overview

- CPU-only nodes
  - AMD EPYC™
  - Milan CPUs

- GPU-accelerated nodes
  - A100 “Ampere” NVIDIA GPUs
  - Tensor Cores

- All-Flash Platform
  - Integrated Storage
  - 35 PB, 5+ TB/s

- Workflow Nodes
  - High-memory Nodes

- User Access (Login) Nodes

- External Filesystems & Networks

- “Slingshot” Ethernet Compatible Interconnect

---

Phase 1
- Late 2020 - Early 2021

Phase 2
- Mid 2021

Partly Phase 1
- Partly Phase 2
GPU nodes in Perlmutter Phase 1

- AMD EPYC 7763
- NVIDIA A100
- DDR4

Connections:
- PCIe 4.0 x16 (32 GB/s/dir)
- NVLink (3rd gen) (25 GB/s/dir)
GPU nodes in depth

- 4x NVIDIA A100 GPUs
- 1 AMD EPYC 7763 CPU
- CPU connected to GPUs via PCIe 4.0
- NVLink connected A2A across GPUs, 4x bonded
- FP16, TF32, FP64 tensor cores on GPUs
- Multi-Instance GPU

<table>
<thead>
<tr>
<th></th>
<th>V100</th>
<th>A100</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FP64 Peak</strong></td>
<td>7.5 TF FMA</td>
<td>19.5 TF TC (9.7 TF FMA)</td>
</tr>
<tr>
<td><strong>FP16 Peak</strong></td>
<td>125 TF TC</td>
<td>312 TF TC</td>
</tr>
<tr>
<td><strong>SMs</strong></td>
<td>80</td>
<td>108</td>
</tr>
<tr>
<td><strong>Memory BW</strong></td>
<td>900 GB/s</td>
<td>1555 GB/s</td>
</tr>
<tr>
<td><strong>Memory Size</strong></td>
<td>16 GB (HBM2)</td>
<td>40 GB (HBM2)</td>
</tr>
<tr>
<td><strong>L2 Cache</strong></td>
<td>6 MB</td>
<td>40 MB</td>
</tr>
<tr>
<td><strong>Shared Mem. / SM</strong></td>
<td>96 KB</td>
<td>164 KB</td>
</tr>
</tbody>
</table>
High-speed network

- Perlmutter’s high-speed network (“Slingshot”) connects compute nodes, Lustre storage nodes, login nodes, workflow nodes, and other service nodes into a single network
- Uses a “dragonfly” topology, and is an evolution of the dragonfly topology used in Cray’s XC product
- Dragonfly is a hierarchy of A2A connections, with each group acting as a “virtual” high-radix router, to minimize the diameter of the network while also minimizing the use of active optical cables (which are very expensive)

Fig. 1: A dragonfly group with all-to-all local connections. Boxes are routers, circles are nodes, solid lines are electrical local links, and dashed lines are optical global links.

Aries HSN on Cray XC systems

- Each compute blade has 4 nodes and 1 Aries ASIC; 16 blades form a chassis
- Each of the 16 Aries ASICs in a chassis are connected A2A with a backplane
- Each chassis (3 per cabinet) is connected A2A to the other 5 chassis in a 2-cabinet group with electrical cables (short, fast, cheap)
- Each group connected A2A to every other group with active optical cables (long, fast, expensive)
- Aries HSN protocol is specialized and proprietary - only special types of nodes can use it

Slingshot HSN on Shasta systems

- Compute blades and Slingshot switch blades are oriented at 90 deg
  - Compute blades are vertical, switch blades horizontal
  - Enables A2A connectivity to compute nodes in a group without a backplane
- Slingshot is Ethernet-compatible - enables high-bandwidth connectivity to many types of network endpoints, not just compute nodes and specialized service nodes
- Lustre storage nodes connected directly to HSN - no more LNet routers

“HPE Cray EX Liquid-Cooled Cabinet for Large-Scale Systems brochure” (hpe.com)
All-flash file system

- **Fast across many dimensions**
  - > 4 TB/s sustained bandwidth
  - > 7,000,000 IOPS
  - > 3,200,000 file creates/sec
- **Usable for NERSC users**
  - > 30 PB usable capacity
  - Familiar Lustre interfaces
  - New data movement capabilities
- **Optimized for data workloads**
  - NEW small-file I/O improvements
  - NEW features for high IOPS, non-sequential I/O
Data Movement

- Project file system replaced with Community File System
- NERSC-HPE collaboration will simplify data motion between Perlmutter & CFS
- Bandwidth and capacity are competing resources - tiered storage enables NERSC to spend $ on each where it is most critical
Software configuration and user environment
## Perlmutter Programming Environments

<table>
<thead>
<tr>
<th></th>
<th>GPU Support</th>
<th>Fortran/C/C++</th>
<th>OpenACC 2.x</th>
<th>OpenMP 5.x</th>
<th>CUDA</th>
<th>Kokkos, RAJA</th>
<th>Cray MPI</th>
<th>HIP</th>
<th>DPC++ / SYCL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVIDIA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GNU</td>
<td></td>
<td></td>
<td>(Community Effort)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LLVM</td>
<td></td>
<td></td>
<td></td>
<td>(Community Effort)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Vendor Supported**
- **NERSC Supported**
GPUDirect RDMA enables GPUs on different compute nodes to share data without copying the data first to the host CPU.

Removing CPU host memory from the data motion path improves performance due to reduced number of trips along PCIe.

Implemented in network drivers and kernel module - no user intervention required.

Most CUDA-aware MPI implementations already “do the right thing” to take advantage of GPUDirect RDMA.

Figure 1. GPUDirect RDMA communication model

“Mellanox OFED GPUDirect RDMA Software Product Brief” (mellanox.com)
CUDA-aware MPI

- Cray MPI for Perlmutter is “CUDA-aware”: the programmer may put pointers to GPU memory in many MPI function calls.
- CUDA-aware programming has performance and convenience features:
  - No manual cudaMemcpy() required before calling MPI function.
  - MPI library may avoid cudaMemcpy() altogether and use GPUDirect RDMA where appropriate.
- Relying on CUDA-aware MPI also has pitfalls:
  - Code may crash if running on a system which does not have CUDA-aware MPI.

Without CUDA-aware MPI:
```c
cudaMemcpy(buf_h, buf_d, size, cudaMemcpyDeviceToHost);
MPI_Send(buf_h, size, MPI_CHAR, 1, 100, MPI_COMM_WORLD);
```

With CUDA-aware MPI:
```c
MPI_Send(buf_d, size, MPI_CHAR, 1, 100, MPI_COMM_WORLD);
```
CUDA Unified Memory

- GPU compute nodes on Perlmutter support CUDA Unified Memory
  - CPU and GPU see a common address space, can interact with memory without explicit copies between CPU <-> GPU
  - CUDA runtime automatically migrates unified memory between CPU <-> GPU
- UM provides **convenience** and **consistency**, but not always **performance**

```c
cudaMallocManaged(&x, ...);
gpu_kernel<<<nblk, blksz>>>(x, ...);
cpu_function(x, ...);
```
OpenMP NRE partnership with NVIDIA

• Agreed upon subset of OpenMP features to be included in the NVIDIA HPC SDK compiler
• OpenMP test suite created with micro-benchmarks, mini-apps, and the ECP SOLLVE V&V suite
• 5 NESAP application teams partnering with NVIDIA to add OpenMP target offload directives
• NVIDIA HPC SDK compiler versions >= 20.11 include the OpenMP offload capability developed as part of this NRE
Analytics and Workflow Integration

- Software
  - Optimized analytics libraries, includes Cray Analytics stack
  - Collaboration with NVIDIA for Python-based data analytics support
  - Support for containers
- Perlmutter will aid complex end-to-end workflows
  - Slurm co-scheduling of multiple resources and real-time/deadline scheduling
  - Workflow nodes: container-based services
    - Connections to scalable, user workflow pool (via Spin) with network/scheduler access
  - High-availability workflow architecture and system resiliency for real-time use-cases
What is in Perlmutter’s future?
Summary

• Perlmutter is a heterogeneous CPU+GPU system designed to accelerate the diverse data-centric and computational workflows for thousands of NERSC users
• First phase of Perlmutter with all ~6000 NVIDIA A100 GPUs has been delivered in NERSC’s data center and is undergoing integration and testing
• The system will support a wide range of programming languages and models, ensuring that its broad workload will be able to use its GPUs effectively