APPLICATION AND MICROBENCHMARK STUDY USING MVAPICH2 AND MVAPICH2-GDR ON SDSC COMET AND EXPANSE

MVAPICH USER GROUP MEETING
AUGUST 26, 2020

MAHIDHAR TATINENI
SDSC

EXPANSE
COMPUTING WITHOUT BOUNDARIES

SAN DIEGO SUPERCOMPUTER CENTER

NSF Award 1928224

Overview

- History of InfiniBand based clusters at SDSC
- Hardware summaries for Comet and Expanse
- Application and Software Library stack on SDSC systems
- Containerized approach – Singularity on SDSC systems
- Benchmark results on Comet, Expanse development system - OSU Benchmarks, NEURON, RAxML, TensorFlow, QE, and BEAST are presented.




InfiniBand and MVAPICH2 on SDSC Systems

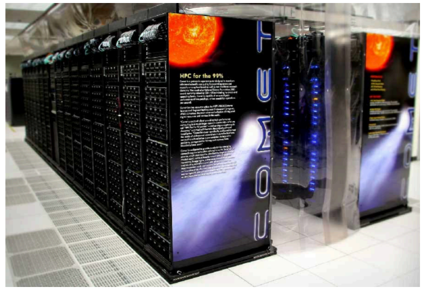
Trestles (NSF)
2011-2014

- *324 nodes, 10,368 cores*
- *4-socket AMD Magny-Cours*
- *QDR InfiniBand*
- *Fat Tree topology*
- *MVAPICH2*

Gordon (NSF)
2012-2017
GordonS
(Simons Foundation)
2017- 2020

- *1024 nodes, 16,384 cores*
- *2-socket Intel Sandy Bridge*
- *Dual Rail QDR InfiniBand*
- *3-D Torus topology*
- *300TB of SSD storage - via iSCSI over RDMA (iSER)*
- *MVAPICH2 (1.9, 2.1) with 3-D torus support*

COMET (NSF)
2015-Current



- *1944 compute, 72 GPU, and 4 large memory nodes.*
- *2-socket Intel Haswell*
- *FDR InfiniBand*
- *Fat Tree topology*
- *MVAPICH2, MVAPICH2-X, MVAPICH2-GDR*
- *Leverage SRIOV for Virtual Clusters*

Expanse (NSF)
Production Fall 2020

- ***728 compute, 52 GPU, and 4 large memory nodes.***
- ***2-socket AMD EPYC 7742, HDR100 InfiniBand***
- ***GPU nodes with 4 V100 GPUs + NVLINK***
- ***HDR200 Switches, Fat Tree topology with 3:1 oversubscription***
- ***MVAPICH2, MVAPICH2-X, MVAPICH2-GDR***

Comet: System Characteristics

- Total peak flops ~2.76 PF
- Dell primary integrator
  - *Intel Haswell processors w/ AVX2*
  - *Mellanox FDR InfiniBand*
- 1,944 standard compute nodes (46,656 cores)
  - *Dual CPUs, each 12-core, 2.5 GHz*
  - *128 GB DDR4 2133 MHz DRAM*
  - *2*160GB GB SSDs (local disk)*
- 72 GPU nodes
  - *36 nodes with two NVIDIA K80 cards, each with dual Kepler3 GPUs*
  - *36 nodes with 4 P100 GPUs each*
- 4 large-memory nodes
  - *1.5 TB DDR4 1866 MHz DRAM*
  - *Four Haswell processors/node; 64 cores/node*
- Hybrid fat-tree topology
  - *FDR (56 Gbps) InfiniBand*
  - *Rack-level (72 nodes, 1,728 cores) full bisection bandwidth*
  - *4:1 oversubscription cross-rack*
- Performance Storage (Aeon)
  - *7.6 PB, 200 GB/s; Lustre*
  - *Scratch & Persistent Storage segments*
- Durable Storage (Aeon)
  - *6 PB, 100 GB/s; Lustre*
- Home directory storage
- Gateway hosting nodes
- Virtual image repository
- 100 Gbps external connectivity to Internet2 & ESNet

Expanse Overview

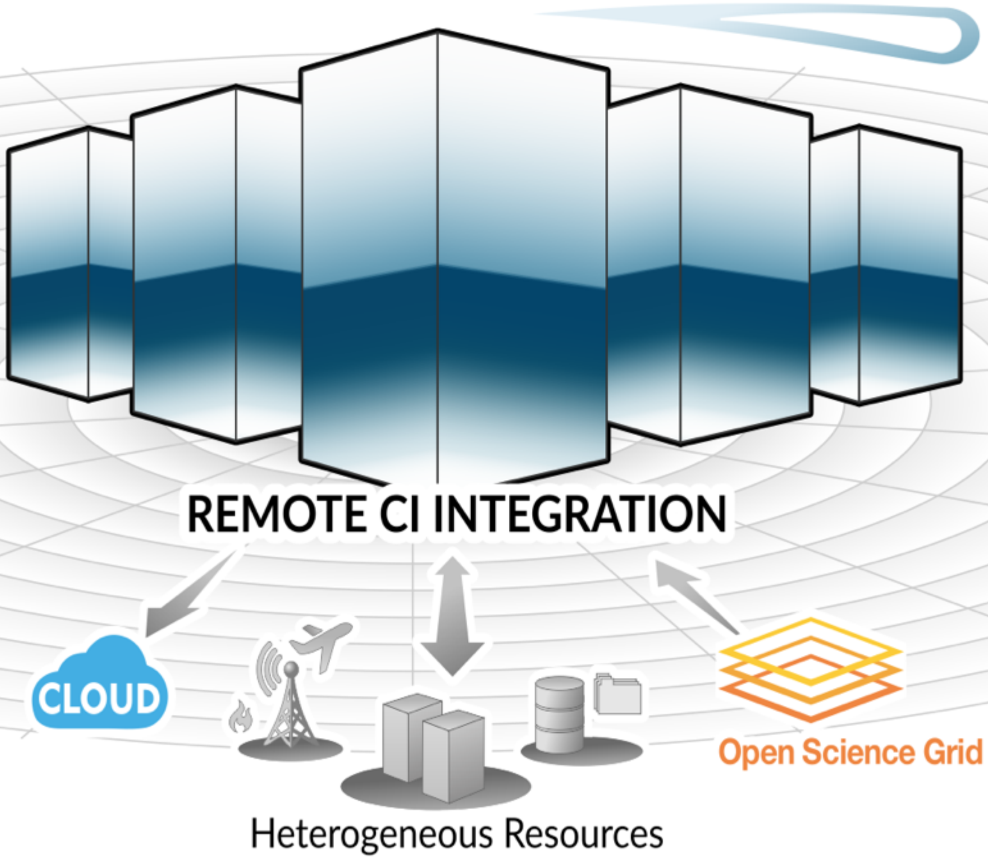
- Category 1: Capacity System, NSF Award # 1928224
- PIs: Mike Norman (PI), Ilkay Altintas, Amit Majumdar, Mahidhar Tatineni, Shawn Strande
- Based on benchmarks we've run, we expect > 2x throughput over Comet, and 1-1.8x per-core performance over Comet's Haswell cores
- SDSC team has compiled and run many of the common software packages on AMD Rome based test clusters and verified performance.
- Expect a smooth transition from Comet and other systems

EXPANSE

COMPUTING WITHOUT BOUNDARIES
5 PETAFLOP/S HPC and DATA RESOURCE

HPC RESOURCE
- 13 Scalable Compute Units
- 728 Standard Compute Nodes
- 52 GPU Nodes: 208 GPUs
- 4 Large Memory Nodes

DATA CENTRIC ARCHITECTURE
- 12PB Perf. Storage: 140GB/s, 200k IOPS
- Fast I/O Node-Local NVMe Storage
- 7PB Ceph Object Storage
- High-Performance R&E Networking



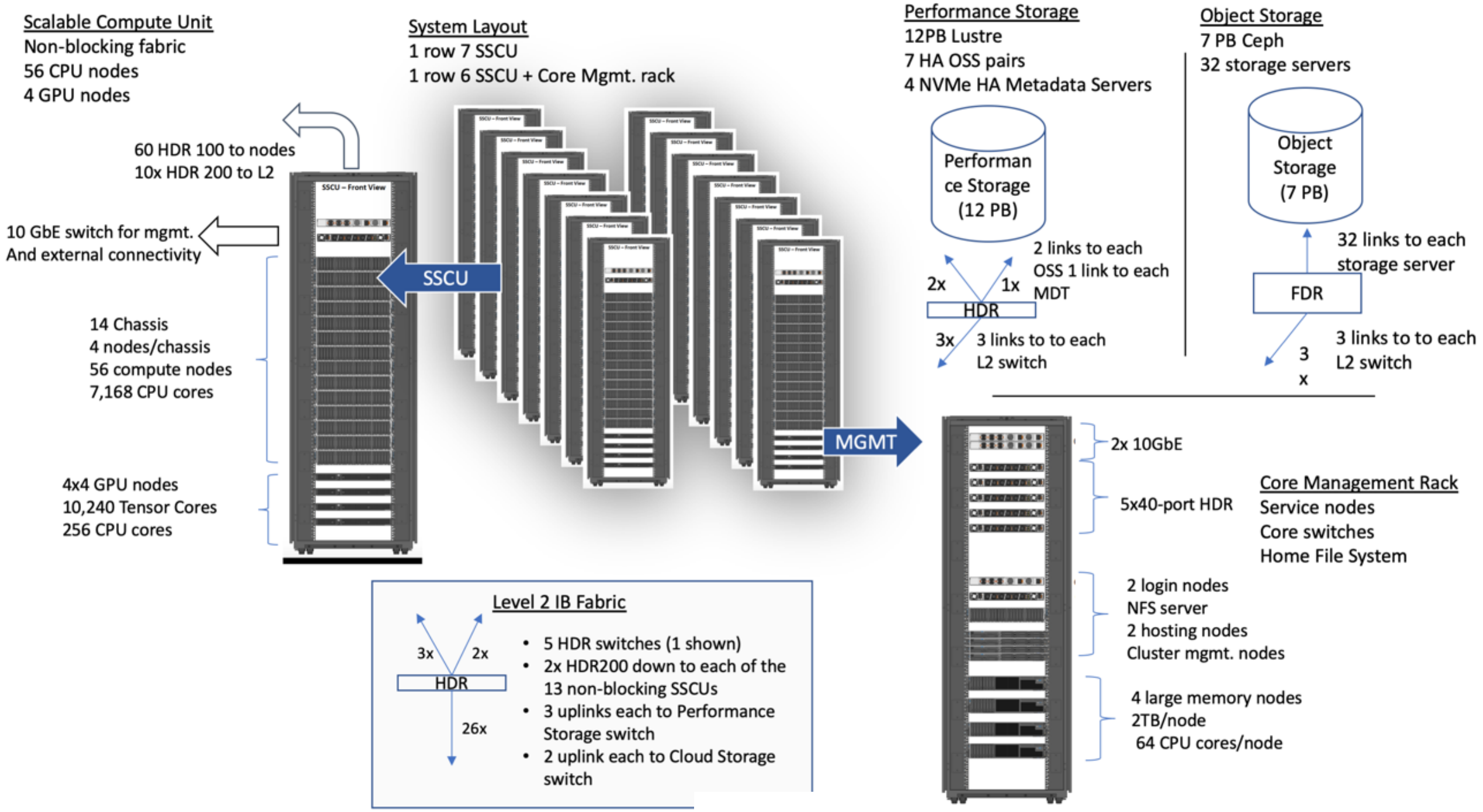
LONG-TAIL SCIENCE
- Multi-Messenger Astronomy
- Genomics
- Earth Science
- Social Science

INNOVATIVE OPERATIONS
- Composable Systems
- High-Throughput Computing
- Science Gateways
- Interactive Computing
- Containerized Computing
- Cloud Bursting

Expanse System Summary

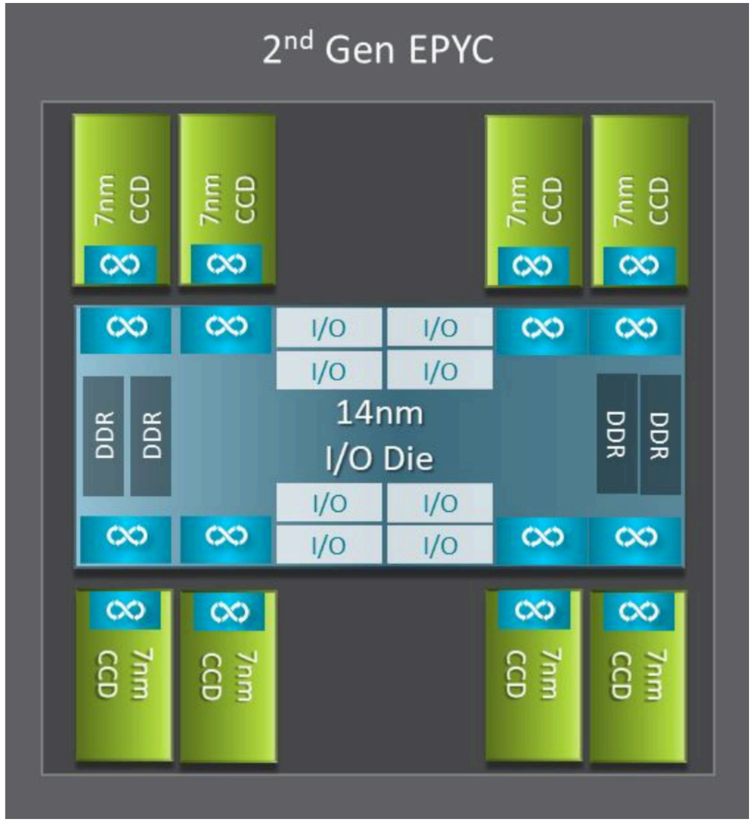
System Component	Configuration
AMD EPYC (Rome) 7742 Compute Nodes	
Node count	728
Clock speed	2.25 GHz
Cores/node	128
Total # cores	93,184
DRAM/node	256 GB
NVMe/node	1 TB
NVIDIA V100 GPU Nodes	
Node count	52
Total # GPUs	208
GPUs/node	4
GPU Type	V100 SMX2
Memory/GPU	32 GB
CPU cores; DRAM; clock (per node)	40; 384 GB; 2.5 GHz;
CPU	6248 Xeon
NVMe/node	1.6TB
Large Memory Nodes	
Number of nodes	4
Memory per node	2 TB
CPUs	2x AMD 7742/node;

Storage	
Lustre file system	12 PB (split between scratch & allocable projects)
Home File system	1 PB



AMD EPYC 7742 Processor Architecture

- 8 Core Complex Dies (CCDs).
- CCDs connect to memory, I/O, and each other through the I/O Die.
- 8 memory channels per socket.
- DDR4 memory at 3200MHz.
- PCI Gen4, up to 128 lanes of high speed I/O.
- Memory and I/O can be abstracted into separate quadrants each with 2 DIMM channels and 32 I/O lanes.



Reference: https://developer.amd.com/wp-content/resources/56827-1-0.pdf

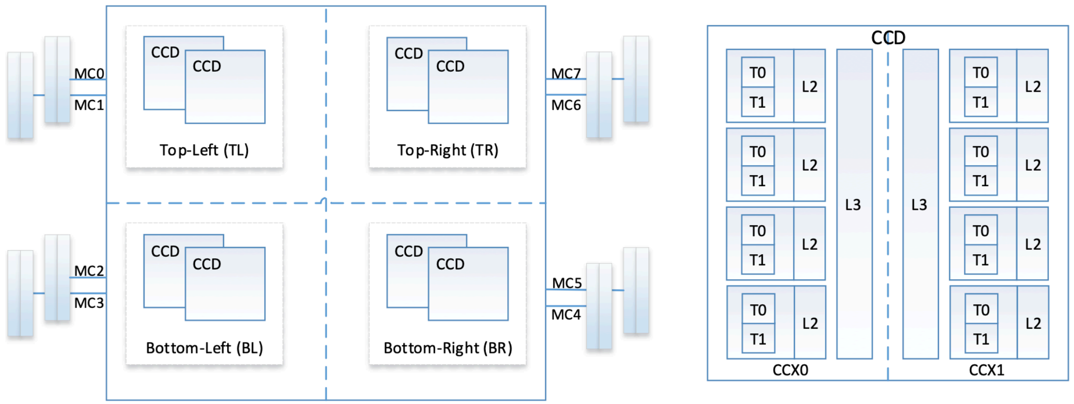
AMD EPYC 7742 Processor: Core Complex Die (CCD)

- 2 Core Complexes (CCXs) per CCD
- 4 Zen2 cores in each CCX shared a 16M L3 cache. Total of 16 x 16 = 256MB L3 cache.
- Each core includes a private 512KB L2 cache.

Reference: https://developer.amd.com/wp-content/resources/56827-1-0.pdf

AMD EPYC 7742 Processor : NUMA Nodes Per Socket

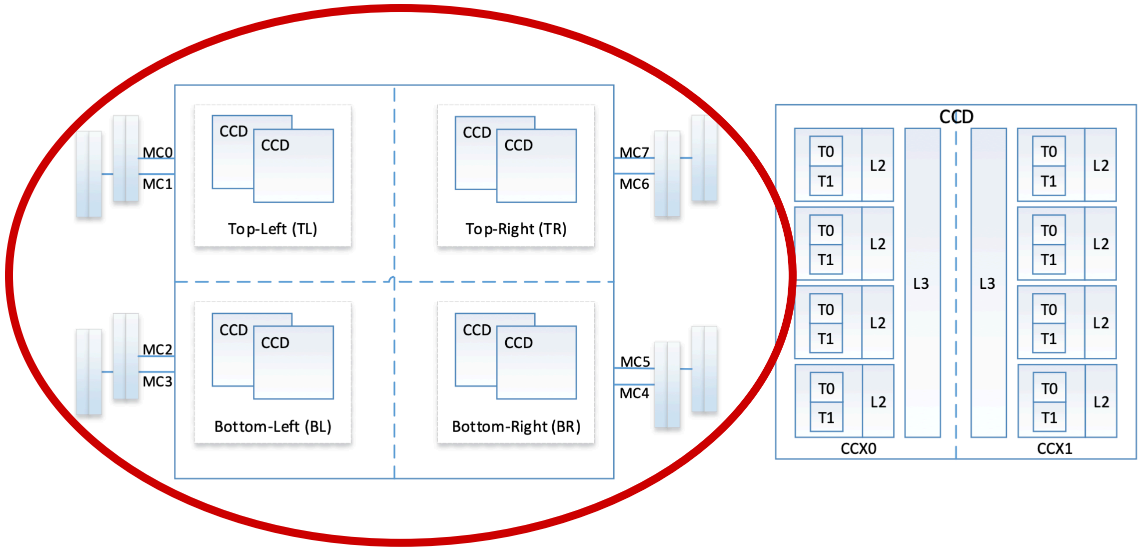
- The four logical quadrants allow the processor to be partitioned into different NUMA domains. Options set in BIOS.
- Domains are designated as NUMA per socket (NPS).
- **NPS4:** Four NUMA domains per socket is the typical HPC configuration.



https://developer.amd.com/wp-content/resources/56338_1.00_pub.pdf

NPS1 Configuration

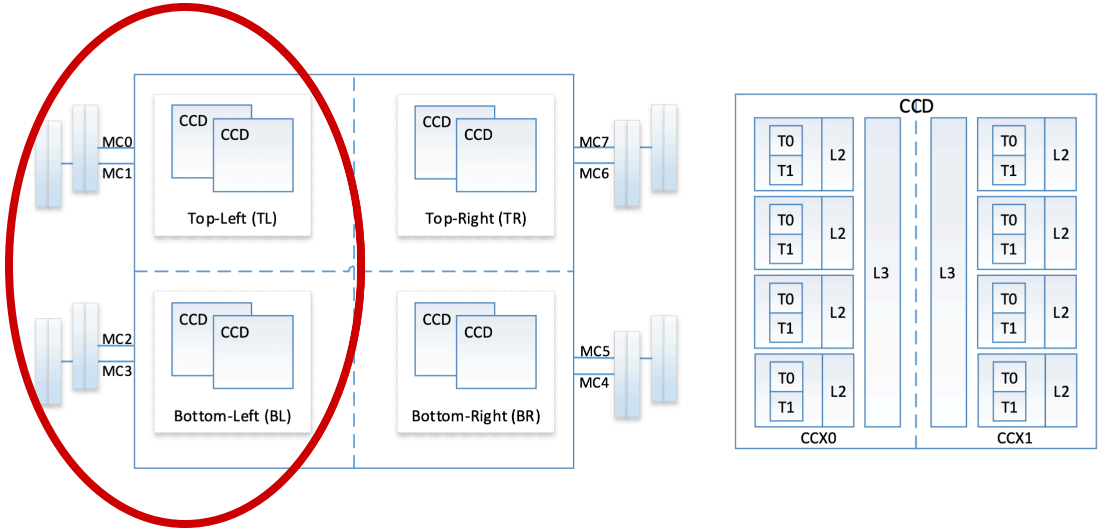
- **NPS1:** the processor is a single NUMA domain.
- Memory is interleaved across all 8 memory channels.
- Can try if workload is not very well NUMA aware



https://developer.amd.com/wp-content/resources/56338_1.00_pub.pdf

NPS2 Configuration

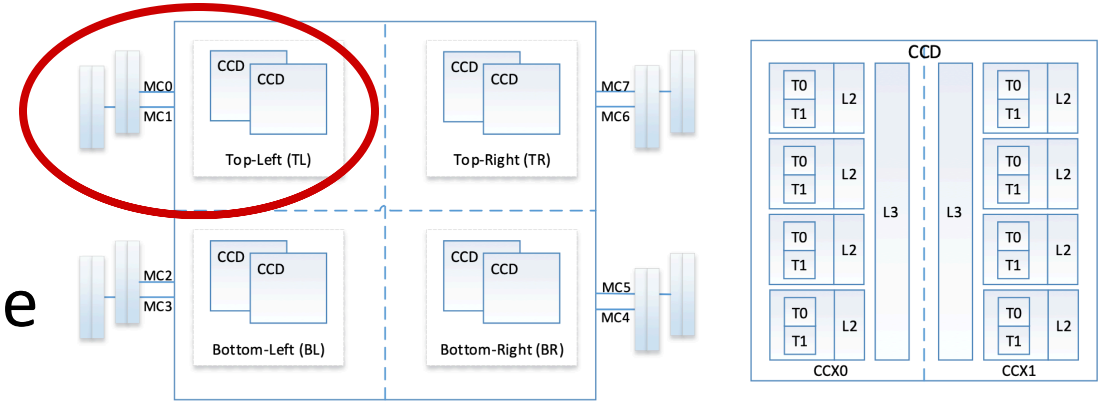
- Processor is partitioned into two NUMA domains in **NPS2** setting.
- Half the cores and half the memory channels connected to the processor are in one NUMA domain
- Memory is interleaved across the four memory channels



https://developer.amd.com/wp-content/resources/56338_1.00_pub.pdf

NPS4 Configuration

- The processor is partitioned into four NUMA domains.
- Each logical quadrant is a NUMA domain.
- Memory is interleaved across the two memory channels
- PCIe devices will be local to one of four NUMA domains (the IO die that has the PCIe root for the device)
- ***This is the typical HPC configuration*** as workload is NUMA aware, ranks and memory can be pinned to cores and NUMA nodes.



https://developer.amd.com/wp-content/resources/56338_1.00_pub.pdf

GPU Node Architecture

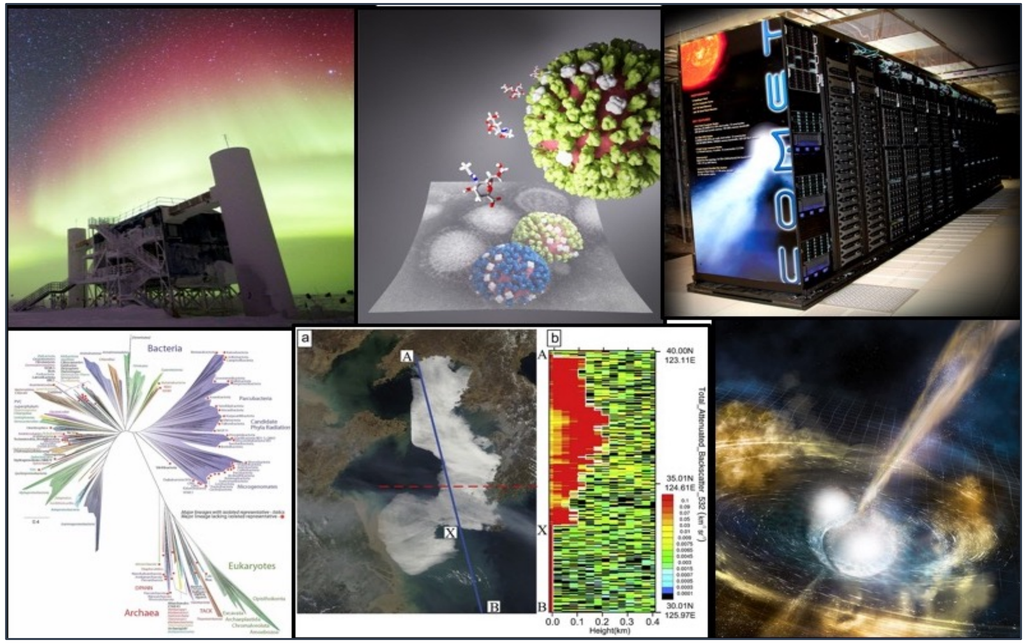
- 4 V100 32GB SMX2 GPUs
- 384 GB RAM, 1.6 TB PCIe NVMe
- 2 Intel Xeon 6248 CPUs
- Topology:

```
        GPU0    GPU1    GPU2    GPU3    mlx5_0  CPU Affinity
GPU0     X      NV2     NV2     NV2     SYS     0-0,4-4,8-8,12-12,16-16,20-20,24-24,28-28,32-32,36-36
GPU1    NV2      X      NV2     NV2     SYS     0-0,4-4,8-8,12-12,16-16,20-20,24-24,28-28,32-32,36-36
GPU2    NV2     NV2      X      NV2     SYS     1-1,5-5,9-9,13-13,17-17,21-21,25-25,29-29,33-33,37-37
GPU3    NV2     NV2     NV2      X      SYS     1-1,5-5,9-9,13-13,17-17,21-21,25-25,29-29,33-33,37-37
mlx5_0  SYS     SYS     SYS     SYS      X

Legend:

  X    = Self
  SYS  = Connection traversing PCIe as well as the SMP interconnect between NUMA nodes (e.g., QPI/UPI)
  NODE = Connection traversing PCIe as well as the interconnect between PCIe Host Bridges within a NUMA node
  PHB  = Connection traversing PCIe as well as a PCIe Host Bridge (typically the CPU)
  PXB  = Connection traversing multiple PCIe bridges (without traversing the PCIe Host Bridge)
  PIX  = Connection traversing at most a single PCIe bridge
  NV#  = Connection traversing a bonded set of # NVLinks
```

Comet advances science and engineering discovery for a broad user base => large application and software library stack



Clockwise from upper left: IceCube Neutrino Detection; Battling Influenza; Comet Surpasses 40,000 Users; Detecting Gravitational Waves; Predicting Sea Fog; Defining a New Tree of Life

In just over 4 years on Comet:

- 40,000+ Unique Users
- 1,200+ Publications
- ~2,000 Research, education and startup allocations
- 400+ Institutions
- Scientific discoveries and breakthroughs

Expanse will also support a broad user base and large application stack

Applications Stack on Comet

- Comet supports a wide array of applications and libraries as detailed below.
- Additionally, SDSC staff maintain a set of Singularity container images and provide the definition files for interested users.

Domain	Software
Biochemistry	APBS, Rosetta
Bioinformatics	BamTools, Bali-Phy, BCFtools, BEAGLE, BEAST, BEAST 2, bedtools, bioperl, Biopython, Bismark, BLAST, BLAT, Bowtie, Bowtie 2, BWA, Celera, Cufflinks, dDocent, DendroPy, Diamond, DPPDiv, Edena, FastQC, FastTree, FASTX-Toolkit, FSA, GARLI, GATK, GMAP-GSNAP, IDBA-UD, jModelTest2, MAFFT, Migrate, miRDeep2, MrBayes, PhyloBayes, Picard, PLINK, Pysam, QIIME, RAxML, RSeQC, SAMtools, SOAPdenovo2, SOAPsnp, SPAdes, Stacks, TopHat, Trimmomatic, Trinity, Velvet, ViennaRNA
Compilers	GNU, **Intel**, Mono, **PGI**
File format libraries	HDF4, HDF5, NetCDF
Interpreted languages	**MATLAB**, Octave, R, RStudio
Large-scale data-analysis frameworks	Hadoop 1, Hadoop 2 (with YARN), Spark, RDMA-Hadoop, RDMA-Spark
Molecular dynamics	Amber, Gromacs, LAMMPS, NAMD

Domain	Software
Computational Fluid Dynamics	OpenFOAM
MPI libraries	MPICH2, MVAPICH2, Open MPI, **IntelMPI**
Numerical libraries	ATLAS, FFTW, FSL, GDAL, GSL, JAGS, LAPACK, **MKL**, ParMETIS, PETSc, ScaLAPACK, SPRNG, Sundials, SuperLU, Trilinos
Predictive analytics	KNIME, Mahout, Weka
Machine Learning/Deep Learning	TensorFlow, Caffe, Torch, PyTorch, Deep-Torch
Profiling and debugging	**DDT**, **IDB**, IPM, mpiP, PAPI, TAU, Valgrind
Quantum chemistry	CPMD, CP2K, GAMESS, **Gaussian**, MOPAC, NWChem, **Q-Chem**, VASP, VASPsol, ORCA
Structural mechanics	**Abaqus**
Visualization	**IDL**, **ENVI**, VisIt, VMD

Libraries and Applications Software Current Approach on SDSC systems

- Users can manage environment via modules.
- Applications packaged into "Rocks Rolls" that can built and deployed on any of the SDSC systems. Benefits wider community deploying software on their Rocks clusters.
- Efficient system administration pooling software install/testing efforts from different projects/machines – Comet benefits from work done for Trestles, Gordon, and Triton Shared Computing Cluster (TSCC).
- Users benefit from a familiar applications environment across SDSC systems => can easily transition to Comet, TSCC from older systems.
- Rolls available for Rocks community (https://github.com/sdsc)

Motivation for Spack based approach

- SDSC supports many systems, with thousands of users, and a broad software stack with a small user support team. The motivation is to support a large, diverse software environment as efficiently as possible.
- Leverage work of the wider Spack community for installs.
- SDSC clusters feature a broad range of CPU and GPU architectures. Helps to have ability to have multiple installs – customizing and optimizing for specific targets.
- Easier for users to do customizations – chained Spack installs, environments.
- Systems like Expanse feature cloud integration, composable options. Spack based approach can help simplify the software management.

Compile and run time considerations

- Tested with AOCC, gnu, and Intel compilers. MPI versions include MVAPICH2, OpenMPI, and Intel MPI.

- Specific optimization flags:
  - AOCC, gnu: -march=znver2
  - Intel : -march=core-avx2

- Runtime considerations:
  - MPI: Use binding options such as --map-by core (OpenMPI); I_MPI_PIN, I_MPI_PIN_DOMAIN (Intel MPI); MVAPICH2 MAPPING/AFFINITY flags
  - Open MP: Use affinity options like GOMP_AFFINITY, KMP_AFFINITY
  - Hybrid MPI/OpenMP, MPI/Pthreads: Keep threads on same NUMA domain (or CCX) as parent MPI task using affinity flags or wrapped with taskset (in case of MPI/Pthreads; used in RAxML runs for example)

Singularity on SDSC systems

- Singularity has been available on Comet since 2016 and has become very popular on Comet. Expanse will also support Singularity based containers.
- Typically used for applications with newer library OS requirements than available on the HPC system – e.g. TensorFlow, PyTorch, Caffe2 (SDSC staff maintain optimal versions for Comet).
- Commercial application binaries with specific OS requirements.
- Importing singularity and docker images to enable use in a shared HPC environment. Usually this is entire workflows with a large set of tools bundled in one image.
- Training – encapsulate all the requirements in an image for workshops and SDSC summer institute. Also makes it easy for users to try out outside of the training accounts on Comet.

MVAPICH2-GDR via Singularity Containers

- Installed in Singularity Container
  - NVIDIA driver, CUDA 9.2 (this can alternately be pulled in via the --nv flag)
  - Mellanox OFED stack
  - gdrcopy library - *kernel module is on the host system*.
  - MVAPICH2-GDR (w/o slurm)
  - TensorFlow (conda install)
  - Horovod (pip installed)

- Other modifications:
  - Wrap ssh binary in Singularity container to run remote commands via image environment (more on this next slide)

MVAPICH2-GDR Job Launch w/ Singularity Containers

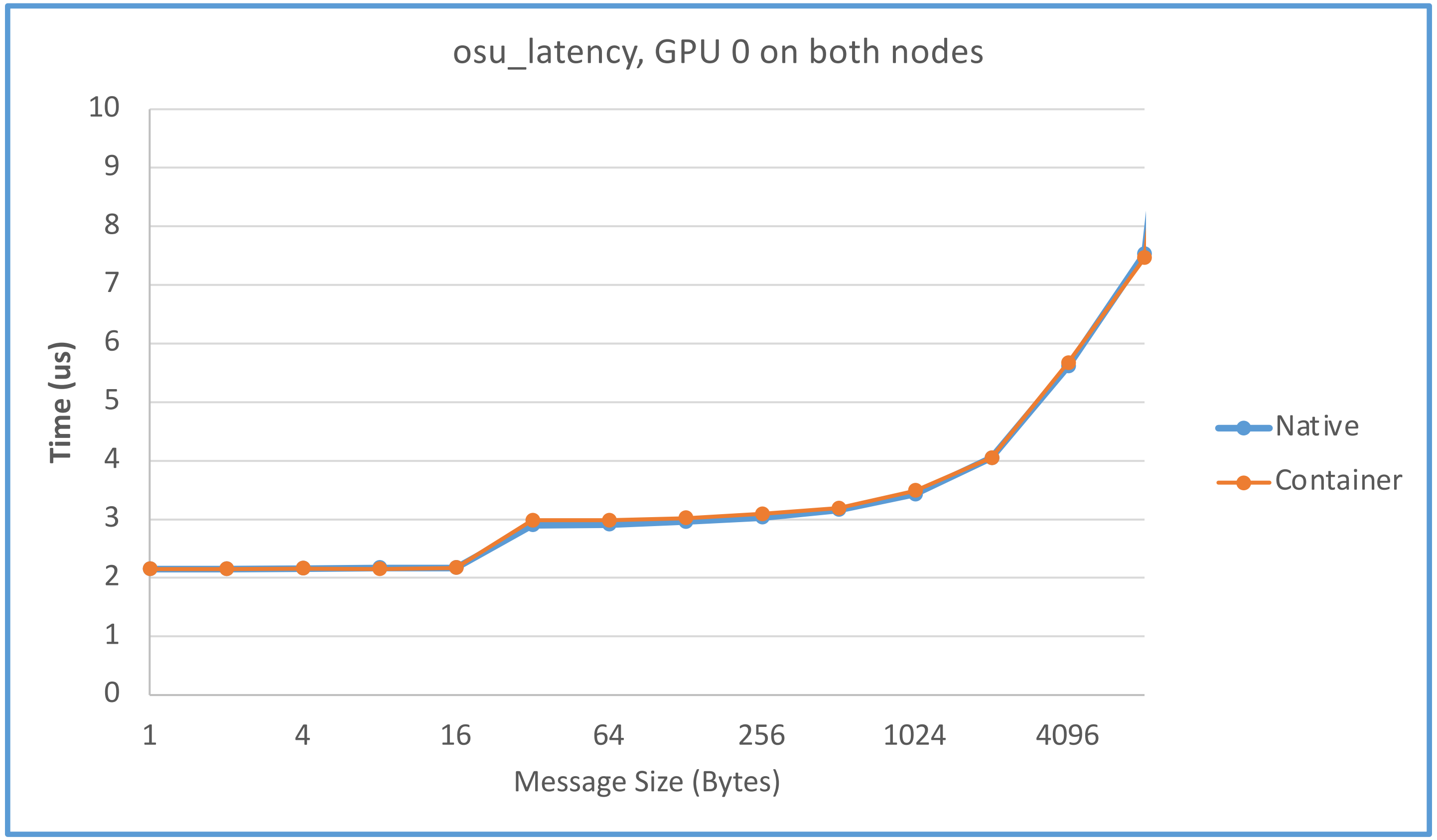
- Use mpirun/mpirun_rsh on the host (external to the image) and wrap the executable/script in Singularity ”exec” command.

- Launch using mpirun_rsh within the Singularity image.
  - Needs ssh to be wrapped so that the remote command is launching in ssh environment
  - ssh binary was moved in container, and then wrapped ssh is used (to point to ssh + singularity command).

Applications and Microbenchmarks

- Typical microbenchmarks include OSU Benchmarks, IOR, STREAM, FIO, and DGEMM.
- CPU applications include GROMACS, NAMD, NEURON, OpenFOAM, Quantum Espresso, RAxML, and WRF. These applications are among the most commonly used on Comet.
- GPU benchmarks include AMBER, NAMD, BEAST, GROMACS, MXNET, PyTorch, and TensorFlow.
- ***Preliminary results*** for some of the benchmarks from the Expanse development system are presented. Some prior results from Comet are presented for comparison.

Comet OSU Latency Results from prior testing: MVAPICH2-GDR (v2.3.2) using Containerized Approach



TensorFlow Benchmark (tf_cnn_benchmarks)

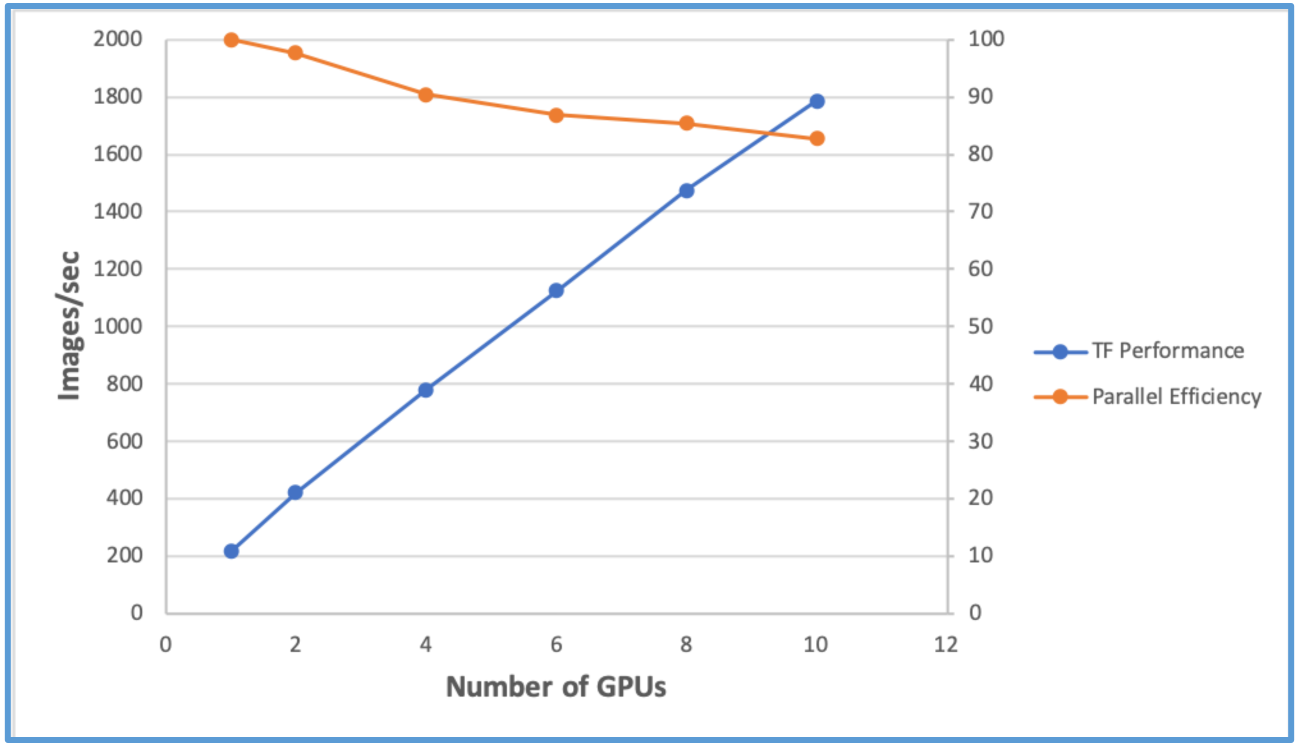
- Interactive access to resources using ”srun”
- Get an interactive shell in Singularity image environment

  singularity shell ./centos7mv2gdr.img

- Run benchmark using hosts (get list from Slurm)

```
export MV2_PATH=/opt/mvapich2/gdr/2.3.2/mcast/no-openacc/cuda9.2/mofed4.5/mpirun/gnu4.8.5
export MV2_USE_CUDA=1
export MV2_USE_MCAST=0
export MV2_GPUDIRECT_GDRCOPY_LIB=/opt/gdrcopy/lib64/libgdrapi.so
export CUDA_VISIBLE_DEVICES=0,1
export MV2_SUPPORT_TENSOR_FLOW=1
$MV2_PATH/bin/mpirun_rsh -export -np 4 comet-34-16 comet-34-16 comet-34-17 comet-34-17 python tf_cnn_benchmarks.py --model=resnet50 --variable_update=horovod > TF_2NODE_4GPU.txt
```

TensorFlow Benchmark Results from prior testing on Comet (GPU 0,1 on each P100 node)



Results for ResNet-50 Benchmark

TensorFlow with MVAPICH2-GDR (v2.3.2) on Popeye

- *Resource for Simons Foundation hosted at SDSC*
- *360 compute nodes and 16 GPU nodes with EDR InfiniBand*
- *GPU Nodes: 4 NVIDIA V100 GPUs along with Intel Skylake processors.*
- ***Expanse GPU nodes are projected to show similar performance***

FP16, Batch Size: 256

GPUs	Images /sec	Scaling
1	775	1
2	1597	2.06
4	2833	3.66
8	5357	6.91

FP32, Batch Size: 256

GPUs	Images /sec	Scaling
1	383	1
2	758	1.98
4	1505	3.93
8	2995	7.82

FP32, Batch Size: 128

GPUs	Images /sec	Scaling
1	360	1
2	732	2.03
4	1421	3.95
8	2757	7.66

Results for ResNet-50 Benchmark

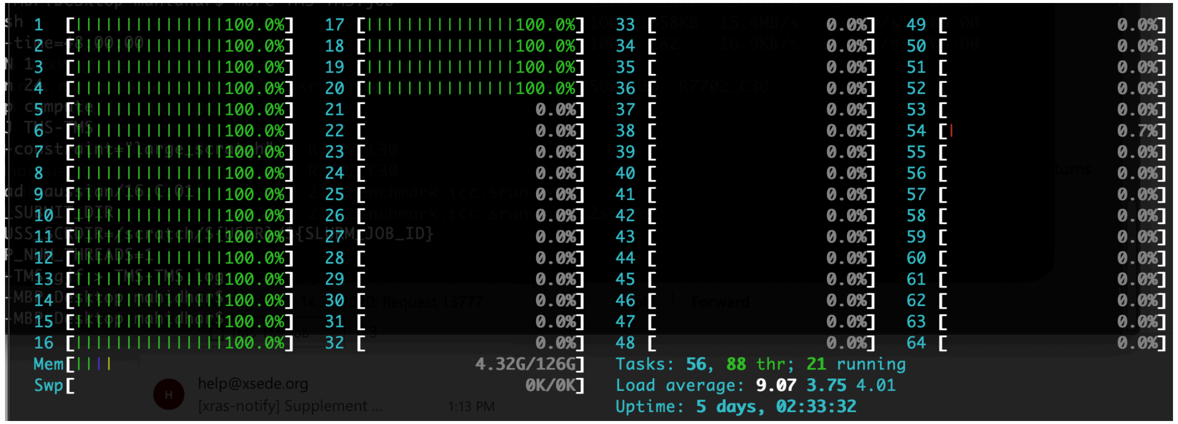
Initial Benchmarks of Applications on AMD Rome Hardware

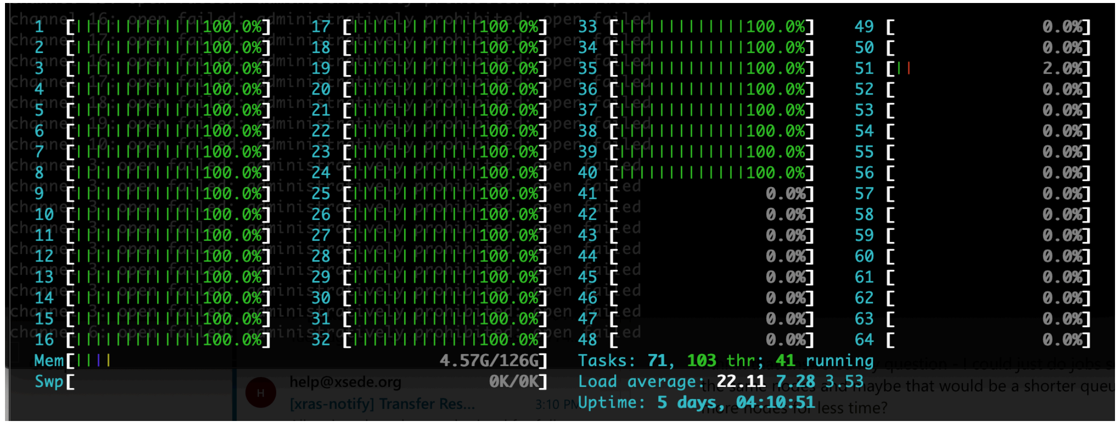
- Benchmarked CPU Applications: GROMACS, NAMD, NEURON, OpenFOAM, Quantum Espresso, RAxML, WRF, and ASTRAL.
- MPI, Hybrid MPI/OpenMP, and Hybrid MPI/Pthreads cases. Compilers used included AOCC, gnu, and Intel.
- Early results on test clusters show performance ranges from matching on a per core basis to 1.8X faster on a per core basis compared to Comet.
- Overall throughput is expected to be easily more than 2X of Comet.
- ***Expanse hardware is currently being installed at SDSC - more benchmarks will be performed in the near future!***
- ***Results from Expanse development node testing are presented in the next few slides. Single socket AMD EPYC 7742 processors + HDR100 on node + HDR200 switch.***

RAxML Benchmark: All-in-one analysis: 218 taxa, 2,294 DNA characters, 1,846 patterns, 100 bootstraps (MPI + Pthreads)

Build: Intel Compiler + MVAPICH2/2.3.4 (Spack installed)

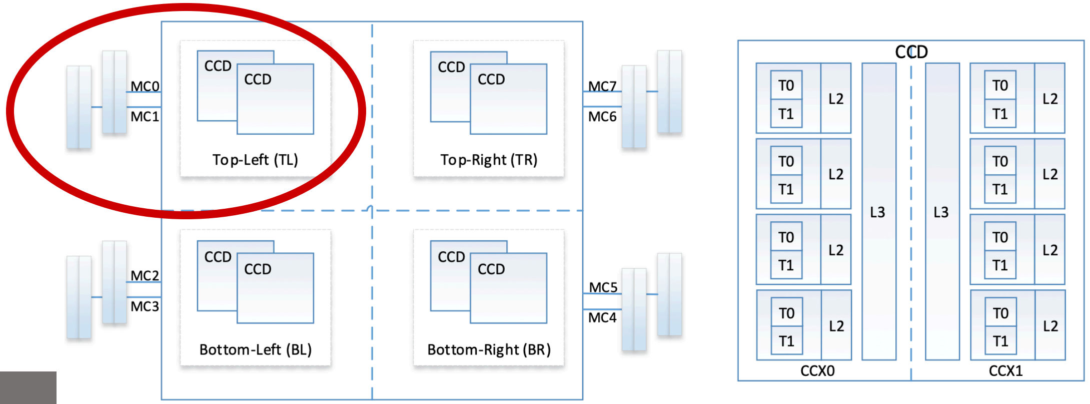
Total tasks	Comet (s)	Stampede2 (s)	Expanse-Dev (s)
10 (5 MPI x 2 Pthreads)	925	610	514
20 (5 MPI x 4 Pthreads)	542	363	292
30 (10 MPI x 3 Pthreads)	433	332	247
40 (10 MPI x 4 Pthreads)	341	300	201





NEURON Benchmark:
Large-scale model of olfactory bulb: 10,500 cells, 40,000 timesteps
Build: Intel + Intel MPI compilers

Total #MPI Tasks	Expanse-Dev (Compact)	Expanse-Dev (Best Memory BW)
16	5004	1781
32	2336	1321
64	1130	1130



NEURON Benchmark:
Large-scale model of olfactory bulb: 10,500 cells, 40K timesteps
Build: Intel + Intel MPI compilers, Results from Dell Test Cluster w/EDR IB

#MPI Tasks	Comet	Test Cluster AMD Rome, EDR IB
96	522 s	525 s
192	264 s	220 s
384	120 s	68 s
768	53 s	35 s

Quantum Espresso Benchmark

PSIWAT: gold + thiols + water 4k points, 586 atoms, 2,552 electrons, 5 iterations

Build: Intel + Intel MPI compilers, Results from Dell Test Cluster w/EDR IB

#MPI Tasks	Comet	Test Cluster AMD Rome, EDR IB
96	1498 s	1263 s
192	776 s	534 s
384	437 s	318 s

BEAST v1.8.2 + BEAGLE v3.1.2 (GPU)
104 taxa, 131,706 DNA characters, 83,144 patterns, 100k steps

Build: Intel Compiler + Threads + CUDA/10.2.89
Spack based build for BEAST and BEAGLE

GPUs	Comet (P100 nodes)	Expanse Development (V100s)
1	217.9 s	153.1 s
4	88.6 s	64.4 s

Important Events/Dates

- **First XRAC submissions complete. Review Aug 31, allocations start Oct 1, 2020.**
- **Upcoming XRAC Allocation submission period:** Sept 15 - Oct 15, 2020. Review of these submissions will be in early December and allocations will start January 1, 2021.
- **Summer 2020:** Hardware delivery, installation, application stack development, and initial testing (ongoing)
- **Expanse Early Access Period:** September 2020
- **Expanse 101: Accessing and running jobs:** Late September 2020
- **Training for Comet to Expanse transition:** October 2020
- **Start of Expanse production period:** October 2020

- Follow all things Expanse at https://expanse.sdsc.edu.

Allocations

- Three resources related to Expanse:
  - **Expanse**: For allocations on compute (AMD Rome) part of the system.
  - **Expanse GPU**: For allocations on the GPU (V100) part of the system.
  - **SDSC Expanse Projects Storage**: Allocations on Expanse projects storage space* (will be mounted on both compute and GPU part of system).
- Allocation request submission link:
  - https://portal.xsede.org/submit-request
  - Next allocation submission window: Sept 15 – Oct 15, 2020.

*Total space available will be 5PB (The 12 PB Lustre based filesystem will be split between projects and scratch areas)

Summary

- Expanse will provide a substantial increase in the performance and throughput compared to the highly successful, NSF-funded Comet supercomputer.
- Expanse features 728, 2-socket AMD-based compute nodes (2.25 GHz EPYC; 64-cores/socket) and 52 4-way GPU nodes based on V100 w/NVLINK.
- Based on benchmarks we've run, we expect > 2x throughput over Comet, and 1-1.8x per-core performance over Comet's Haswell cores.
- Big thanks to Dr. Panda and MVAPICH team for providing MPI implementations for various SDSC systems over the years – Trestles, Gordon, and Comet + ***Expanse upcoming!***

NSF Award# 1341698, Gateways to Discovery: Cyberinfrastructure for the Long Tail of Science
PI: Michael Norman Co-PIs: Shawn Strande, Amit Majumdar, Robert Sinkovits, Mahidhar Tatineni
SDSC Project in Collaboration with Indiana University (led by Geoffrey Fox)
NSF Award #1565336, SHF: Large: Collaborative Research: Next Generation Communication Mechanisms exploiting Heterogeneity, Hierarchy and Concurrency for Emerging HPC Systems
Collaborative project with OSU (Lead Institution, PI: DK Panda), OSC, SDSC, TACC
NSF Award # 1928224, Category 1: Capacity System: Computing Without Boundaries: Cyberinfrastructure for the Long Tail of Science
PI: Mike Norman, CoPIs: Ilkay Altintas, Amit Majumdar, Mahidhar Tatineni, Shawn Strande