Student Cluster Competition: Boston Green Team

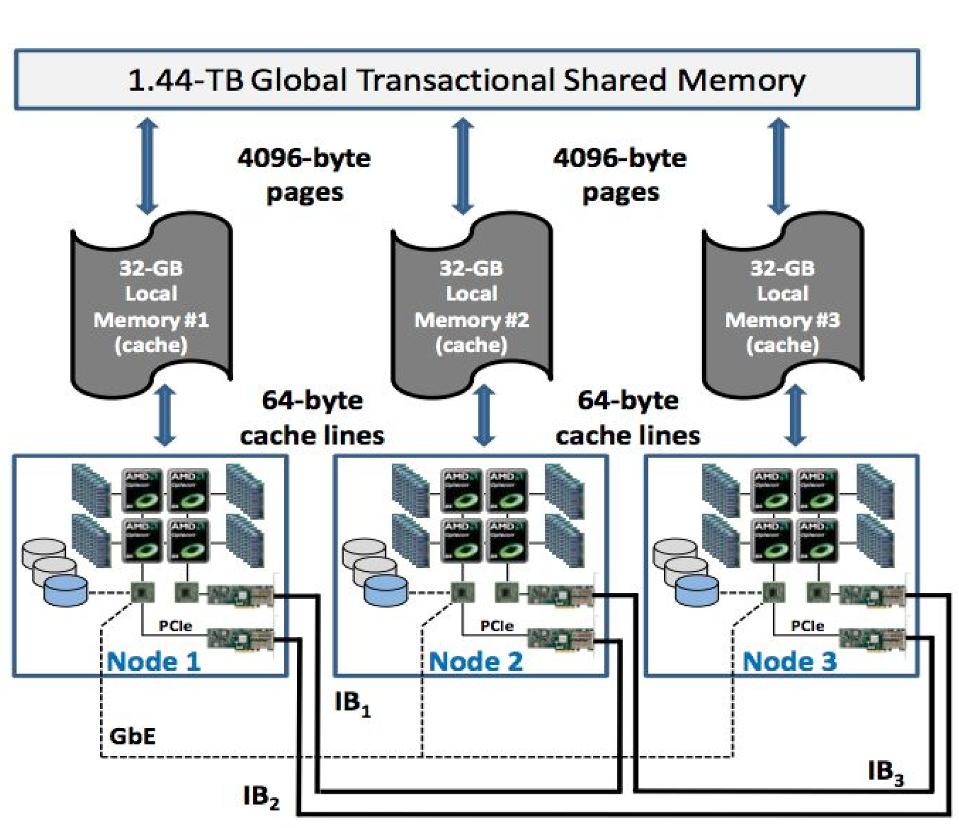
The Team

- The Boston Green Team represents a diverse group of students and an advisor from **three different Massachusetts universities**. Leveraging the numerous colleges within Massachusetts, the team has students from **Boston University and UMass Boston** and an advisor from **MIT**.
- The students have joined together as one unique team that values their one of a kind backgrounds and college educations regardless of their origins.

Hardware / Software

The team chose a Dual Node Symmetric Computing shared memory system. The team plans to take advantage of the massive amount of memory (1 TB) and shared memory architecture so that the processors can be as fast as memory accesses to a same location. The DSMP kernel module allows us to view the cluster as one machine. The idea is that we can use pthreads, openmp, and openmpi to spin up processes using the entire memory space, all the CPUs, and all the GPUs.

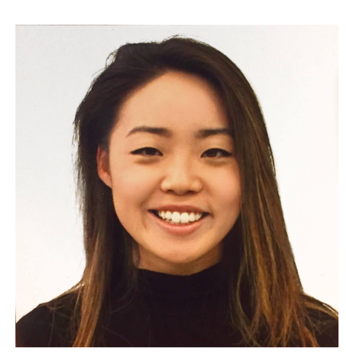
- AMD Opteron 6380 (4 per node), for a total of 128 Cores
- Infiniband QDR Interconnect
- CentOS 6.8
- Slurm
- OpenMPI (shared memory version)
- 2 NVIDIA P100
- 4 AMD Radeon R9 Nano Fury X
- DSMP - enables Distributed Shared Memory (DSM), or Distributed Global Address Space (DGAS), across an InfiniBand QDR connected cluster of homogeneous Symmetric Multiprocessing (SMP) nodes



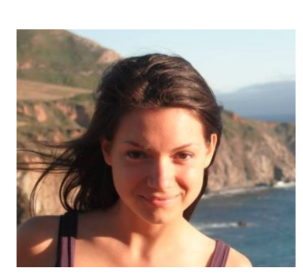
A Symmetric Computing DSMP enabled cluster

Sean Smith (CS '17) is a Senior in the combined BA/MS program in Computer Science Boston University. His research interests include Cloud computing and IOT for HPC. He is in charge of HPL and helped on ParConnect.

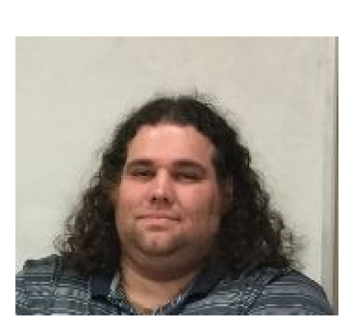
Hannah Gibson (BU CE'18) is a Junior studying Computer Engineering at Boston University. She has experience in Student Cluster Competitions, attending ASC in 2016. She is in charge of ParaView.



Satoe Sakuma (BU CS/IR '17) is a Computer Science and International Relations double major at Boston University. Her research interests are cybersecurity policies. She is in charge of ParConnect.



Masha Vasilenka (UMB CS'17) is a Senior studying Computer Science at Umass Boston. Her research interests are Artificial Intelligence and Big Data Analysis. She worked on HPCG



Evan Donato UMB Math/CS '19 is a Computer Science and Mathematics dual major who spent 8 years in industry as a systems engineer prior to college. His research interests are machine learning and security. He worked on Distributed Password Cracking.

Strategy

For LINPACK and HPCG, we're planning on running them on the entire machine, taking advantage of CUDA optimized versions to parallelize the work on the P100s. To win the ParConnect application, we're using a shared memory flag on openMPI that allows us to take advantage of DSMP's I/O performance increases and caching capabilities. Password cracking takes advantage of the superior single precision of the 4 R9 Nanos. The power of P100s on the benchmarks and shared memory architecture for the applications give us the edge to win the competition.

Applications

- **LINPACK**: a linear algebra library used to benchmark top systems. The team's running an optimized version of the NVIDIA Kepler architecture. For configurations, we experimented with the size of the problem using small deviations of 80% of our total RAM. Our benchmarking yielded a reasonable percentage of our theoretical peak performance. A key part of our experimentation was adjusting block sizes. Block sizes too small limited our performance because there was low data reuse whereas block sizes too large wasted space and extra computation.
- **HPCG**: a metric for ranking HPC systems driven by a multigrid preconditioned Conjugate Gradient algorithm that exercises the key kernels on a nested set of coarse grids. The team's running a CUDA optimized version. We experimented profusely with different matrix sizes each for 60 seconds to get a rough estimate on potential performance. For each result that seemed fruitful. we would run HPCG for 35 minutes and used the matrix size that would produce the most performance for our cluster.
- **Password Cracking:** Prepared multi-tiered approach using dictionary and rulesets to attack easy, likely hashes first. Dictionaries chosen are weighted based on frequency of usage. Rules increase in complexity to allow the picking of low hanging fruit. We benchmarked Jack the Ripper and hashcat on sample problems. We chose to use hashcat for this activity because we consistently saw a performance increase compared to Jack the Ripper.
- **ParaView**: a data visualization and analysis tool. The ParaView installation on our cluster uses Mesa3d with an llvm backend for rendering, OpenMP to parallelize data filtering and shared memory OpenMPI to parallelize loading data.
- **ParConnect:** We were able to communicate directly from node to node using Open MPI's shared memory version. This reduces latency in communication and allows us to load in very large data sets directly into memory far reducing our IO latency.
- **Power Shutoff Activity:** The team discussed using general checkpointing, but decided not to because the duration of the computations would not be long enough to regret losing if a power loss occurred. hashcat has built-in checkpointing that we will leverage.

Sponsors

Symmetric Computing
Supercomputers for Innovation™