Experiences in Optimizing Cluster Performance For Scientific Applications Controlling Configuration, Utilization, and Power Consumption


*Department of Computer Science, Illinois Institute of Technology
**Argonne Leadership Computing Facility, Argonne National Laboratory

{kbrandst, bwalter4, aballmer, ahaider3, adumitru, sdimitro, ayoung11}@hawk.iit.edu, {wscullin, bsallen}@alcf.anl.gov, iraicu@cs.iit.edu

Background

Abstract

In order to achieve the next level of performance from large-scale scientific computing, we explore ways to optimize cluster utility and applications in order to maximize performance, while minimizing power usage. To do this, we look at automated ways to manage cluster power consumption through management of CPU frequencies, fan speeds, and powering down of accelerators when not in use. In addition to hardware controls, we explore automating auto building multiple configurations of applications, and parameter sweeps in order to better predict ideal conditions for peak performance. We have implemented some of the practices in the recent student cluster competition at SC14, where we were required to optimize application performance of several scientific applications while remaining under a power constraint.

The Student Cluster Competition

The Student Cluster Competition is a competition aimed at high-school and undergraduate students, co-located with the IEEE/ACM Supercomputing/SC conference, which aims to bring in the best of the best students from all around the world competing for fame and glory in running 6 high-performance applications/benchmarks (HPHILP, WRF, Trinity, Repast-HPC, MTC, and MILC, and a mystery application) over a 48-hour period on hardware that they have built and configured with the help of sponsors (Argonne National Lab), *Intel*, and *Melanox*. The goal of the competition is for students to build and run a cluster that must remain within a power limit of 26 amps at 120 volts (3120 watts) while maximizing performance.

Team

Ben Walters (Team Captain) is a 2nd year undergraduate student in CS at IIT. He has worked in the Dababiy lab since June 2013. He was an official member of the SCC 2014 team. His puzzle includes WRIF and systems administration.

Alexander Ballmer is a 1st year CS student at IIT. He is a CAWRAS scholar with a full ride scholarship. He was an official member of the SCC 2014 team. His focus is on the HPCC Repast and systems administration.

Ariel Young is a 2nd year student in CS at IIT. She is interested in distributed systems and big data. She is part of monitoring and visualization.

Andrei Georgian Dumitru is a 1st year student studying CS. He has been working in the Dababiy lab since August 2014. He was a backup member of the SCC 2014 team. He is a part of the HPCC lab.

Adnan Haider is currently a 1st year student in CS. His research interests include distributed computing, architecture optimization, and parallel network simulation. He was a backup member of the SCC 2014 team. He is part of Trinity.

Serapheim Dimitropoulos is a 4th year student in CS at IIT. His duties in SCC will include tuning and porting applications to the Xeon Phi accelerators.

Applications

<table>
<thead>
<tr>
<th>Name</th>
<th>HPL-Linpack</th>
<th>WRF</th>
<th>Trinity</th>
<th>Repast-HPC</th>
<th>MILC</th>
<th>MLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain</td>
<td>Bench marking</td>
<td>Climate modeling</td>
<td>Bio-informatics</td>
<td>Decision modeling</td>
<td>High Energy Physics</td>
<td></td>
</tr>
<tr>
<td>Language</td>
<td>Fortran 90</td>
<td>Fortran 90</td>
<td>C++ and Java</td>
<td>C++</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>Algorithms</td>
<td>Linear Algebra</td>
<td>Navier-Stokes equation</td>
<td>Graph Problems</td>
<td>Agent-based Modeling</td>
<td>Monte Carlo, Molecular Dynamics, Heat Bath</td>
<td></td>
</tr>
<tr>
<td>Bottleneck</td>
<td>Everything but I/O</td>
<td>I/O</td>
<td>I/O</td>
<td>Network</td>
<td>Memory</td>
<td>Network</td>
</tr>
</tbody>
</table>

Scheduling and Resource Management

Most of the 48 hour competition is spent waiting for compute jobs to finish running. Thus, queueing jobs to run with the correct datasets would greatly reduce the time and stress needed to monitor jobs during the competition. To do this, a reliable resource manager is needed with the following characteristics:

- Flexibility in scheduling jobs on the same node
- Processor (or core) level granularity
- Support of pre-scripts and post-scripts to set up optimal environments for each job
- Ability to use separate management/compute networks

Challenges

<table>
<thead>
<tr>
<th>Name</th>
<th>HPL-Linpack</th>
<th>WRF</th>
<th>Trinity</th>
<th>Repast-HPC</th>
<th>MILC</th>
<th>MLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain</td>
<td>Bench marking</td>
<td>Climate modeling</td>
<td>Bio-informatics</td>
<td>Decision modeling</td>
<td>High Energy Physics</td>
<td></td>
</tr>
<tr>
<td>Language</td>
<td>Fortran 90</td>
<td>Fortran 90</td>
<td>C++ and Java</td>
<td>C++</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>Algorithms</td>
<td>Linear Algebra</td>
<td>Navier-Stokes equation</td>
<td>Graph Problems</td>
<td>Agent-based Modeling</td>
<td>Monte Carlo, Molecular Dynamics, Heat Bath</td>
<td></td>
</tr>
<tr>
<td>Bottleneck</td>
<td>Everything but I/O</td>
<td>I/O</td>
<td>I/O</td>
<td>Network</td>
<td>Memory</td>
<td>Network</td>
</tr>
</tbody>
</table>

Automated Testing

The process of determining the “best” possible combination of application build options, environmental settings, and system configurations to reduce time to solution while minimizing resource utilization and contention is laborious. By utilizing community tools like Spack, Jenkins, Sorkit Learn, HPCCheckit, Performance Co-Pilot, CPProfile, Tau, and WaveWatch, we can walk through many combinations and permutations without human input and assure high utilization.

Addressing I/O

We plan to focus on how to optimize the I/O performance of our cluster. In particular, we will focus on:

- The type of file system we will deploy
- Performance of different underlying storage devices.
- Parallel file system tuning options
- Stripe Size
- Data layout of stripes: in one data file, horizontally distributed or hybrid of the two

Performance Analysis

- Parallel I/O tracing
- Request Sizes
- Access Patterns

I/O Optimizations

SC15 Network Topology

We designed the mesh network interconnect in order to remove the need for a central switch. In our current design, each node in the cluster has a quad socket motherboard with one dedicated bus per socket. Our design for the network calls for four InfiniBand cards, arranged so that each card uses one dedicated bus. The end result is a mesh where each CPU has a zero hop link to one other node. Four CPUs give a maximum cluster size of five nodes.

Contact: Ioan Raicu, iraicu@cs.iit.edu; for more information, see http://sc15.supercomputing.org/conference-program/student-programs/sc15-student-cluster-competition