GADGET-2
Performance Benchmark and Profiling

May 2011
The following research was performed under the HPC Advisory Council activities
- Participating vendors: Intel, Dell, Mellanox
- Compute resource - HPC Advisory Council Cluster Center

The following was done to provide best practices
- GADGET-2 performance overview
- Understanding GADGET-2 communication patterns
- Ways to increase GADGET-2 productivity
- MPI libraries comparisons

For more info please refer to
- http://www.dell.com
- http://www.intel.com
- http://www.mellanox.com
- http://www.mpa-garching.mpg.de/gadget
• **GADGET-2**
  – “**GAlaxies with Dark matter and GasintEracT**”
  – Is a code for collisionless and gasdynamical cosmological simulations
  – Computes gravitational forces with a hierarchical tree algorithm
  – Used for studies of isolated systems, or for simulations that include the cosmological expansion of space, both with or without periodic boundary conditions
  – Follows the evolution of a self-gravitating collision-less N-body system, allows gas dynamics to be optionally included
Test Cluster Configuration

- **Dell™ PowerEdge™ M610 38-node (456-core) cluster**
  - Six-Core Intel X5670 @ 2.93 GHz CPUs
  - Six-Core Intel X5675 @ 3.06 GHz CPUs
  - Memory: 24GB memory, DDR3 1333 MHz
  - OS: RHEL 5.5, OFED 1.5.2 InfiniBand SW stack

- **Intel Cluster Ready certified cluster**

- **Mellanox ConnectX-2 InfiniBand adapters and non-blocking switches**

- **Storage**: InfiniBand-based Lustre Storage, Lustre 1.8.5

- **Compiler**: Intel Compiler 11.1

- **MPI**: Intel MPI 4.1, Open MPI 1.5.3 with KNEM 0.9.6, Platform MPI 8.1.1

- **Libraries**: FFTW-2.1.5, GNU Scientific Library (GSL) 1.13.3.el5

- **Application**: GADGET 2.0.7

- **Benchmark dataset**: small and large benchmark datasets
Intel® Cluster Ready

• Intel® Cluster Ready systems make it practical to use a cluster to increase simulation and modeling productivity
  – Simplifies selection, deployment, and operation of a cluster

• A single architecture platform supported by many OEMs, ISVs, cluster provisioning vendors, and interconnect providers
  – Focus on work productivity, spend less management time on the cluster

• Select Intel Cluster Ready
  – The cluster is delivered ready to run
  – Hardware and software are integrated and configured together
  – Applications are registered, validating execution on the Intel Cluster Ready architecture
  – Includes Intel® Cluster Checker tool, to verify functionality and periodically check cluster health
About Dell PowerEdge Servers

• **System Structure and Sizing Guidelines**
  – 38-node cluster build with Dell PowerEdge™ M610 blade servers
  – Servers optimized for High Performance Computing environments
  – Building Block Foundations for best price/performance and performance/watt

• **Dell HPC Solutions**
  – Scalable Architectures for High Performance and Productivity
  – Dell's comprehensive HPC services help manage the lifecycle requirements.
  – Integrated, Tested and Validated Architectures

• **Workload Modeling**
  – Optimized System Size, Configuration and Workloads
  – Test-bed Benchmarks
  – ISV Applications Characterization
  – Best Practices & Usage Analysis
GADGET-2 Performance – Results

• **Dataset: small problem**
  – Designed as a small test problem for brief tests of the setup
  – File size for the initial condition: 448MB
  – Memory requirement: 4.5GB (summed over all processes)

• **Dataset: large problem**
  – Designed as a test problem for current state of the art structure formation calculations
  – File size for the initial condition: 28GB
  – Memory requirement: 300GB (summed over all processes)
  – About 1.5GB or memory is used per CPU core for 192-process job

**GADGET-2 Benchmark**
(test_small)

<table>
<thead>
<tr>
<th>Number of CPU Cores</th>
<th>Seconds/Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>20</td>
</tr>
<tr>
<td>24</td>
<td>18</td>
</tr>
<tr>
<td>48</td>
<td>16</td>
</tr>
<tr>
<td>96</td>
<td>14</td>
</tr>
<tr>
<td>192</td>
<td>12</td>
</tr>
<tr>
<td>384</td>
<td>10</td>
</tr>
<tr>
<td>456</td>
<td>8</td>
</tr>
</tbody>
</table>

**GADGET-2 Benchmark**
(test_large)

<table>
<thead>
<tr>
<th>Number of CPU Cores</th>
<th>Seconds/Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>192</td>
<td>150</td>
</tr>
<tr>
<td>384</td>
<td>100</td>
</tr>
<tr>
<td>456</td>
<td>50</td>
</tr>
</tbody>
</table>

- **Total Time**
- **Tree Time**
- **PM Time**

*Lower is better

**InfiniBand QDR**
GADGET-2 Performance – Results

- **InfiniBand QDR enables higher performance and scalability for GADGET-2**
  - Gigabit Ethernet does not show work gain beyond 2 nodes
  - Achieved a 29-fold improvement over Gigabit Ethernet on a 16-node jobs

![GADGET-2 Benchmark (test_small)](chart)

*Higher is better*
GADGET-2 Performance – MPI

• **Platform MPI shows better performance for the small dataset**
  – Runs 12% more jobs compared Open MPI to Platform MPI at 38-node on small dataset

• **Intel MPI shows slightly better performance for the small dataset**
  – Runs 5% more jobs compared Open MPI to Intel MPI at 38-node on large dataset
GADGET-2 Performance – Open MPI

- Processor binding improves performance in Open MPI
  - Up to 47% improvement seen for 38-node when using "--bind-to-core" flag
- Running with KNEM does not show much of an improvement on GADGET-2
  - KNEM typically improves shared memory communications of MPI messages by using RDMA for intra-nodal communications on large messages
GADGET-2 Performance – Process Per Node

- Running with all CPU cores enables higher system utilization
  - Provides up to 83% better performance compared to 6ppn at 38-node
  - Provides up to 46% better performance compared to 8ppn at 38-node

![GADGET-2 Benchmark](image)

Steps/Hour

<table>
<thead>
<tr>
<th>Nodes</th>
<th>16</th>
<th>24</th>
<th>32</th>
<th>38</th>
</tr>
</thead>
<tbody>
<tr>
<td>6ppn</td>
<td></td>
<td></td>
<td>83%</td>
<td></td>
</tr>
<tr>
<td>8ppn</td>
<td></td>
<td></td>
<td></td>
<td>46%</td>
</tr>
<tr>
<td>12ppn</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Higher is better

InfiniBand QDR
Higher CPU frequency provides higher performance
- Seen a 2-4% in work improvement by using CPUs with 3.07GHz vs 2.93GHz
GADGET-2 Profiling – MPI/User Time Ratio

- MPI communication time dominates as cluster scales
  - Reflects that more time spent on message passing communications
  - Percentage of computation is much less on the small dataset

![GADGET-2 Profiling (test_small) MPI/User Time Ratio](image1)

- Number of Messages
- Percentage of Messages

![GADGET-2 Profiling (test_large) MPI/User Time Ratio](image2)

- Number of Messages
- Percentage of Messages
GADGET-2 Profiling – Number of MPI Calls

- Almost the same number of calls for both small and large datasets
- The biggest number of MPI calls is MPI_Sendrecv at 38-node
  - MPI_Sendrecv(85%), MPI_Barrier(4%), MPI_Ssend(3%), MPI_Recv(3%)
- The number of MPI_Sendrecv increases dramatically
  - increases by more than a double when the number of node doubles
GADGET-2 Profiling – MPI Size Distribution

- **MPI_Sendrecv**: Over 50% of the calls begins at 4KB to 16KB
- **MPI_Ssend & MPI_Recv**: Majority are between 64KB to 4MB
- **MPI_Alltoallv**: happens between 16MB and 64MB
- **The rest of the calls have small message sizes**
GADGET-2 Profiling – Time Spent by MPI

- **MPI_Barrier** is the biggest time consumer for large dataset at 38 node
  - MPI_Barrier(38%), MPI_Isend(21%), MPI_Sendrecv(18%), MPI_Recv(9%)
- **For small dataset, MPI_Init** is the leader in MPI time consumer
  - Since less time is spent for MPI communication overall
- **For large dataset, time spent collectively for MPI_Barrier stays constant**
  - However, each MPI rank spends less time in MPI_Barrier individually

![GADGET-2 Profiling (test_small) Time Spent of MPI Calls](chart1)

![GADGET-2 Profiling (test_large) Time Spent of MPI Calls](chart2)
- **Majority of the MPI communications time**
  - Is spent between 1MB to 4MB range
  - Communications include MPI_Sendrecv, MPI_Ssend, MPI_Recv
• **Majority of MPI messages are small messages**
  – In the range of 0 to 64 bytes

• **Same message distribution patterns are seen for both datasets**
  – With the exception that more large message sizes are seen for large dataset
• Data transferred to each process gradually drops as processes increase
  – 40-60MB of data transferred for small dataset for each MPI rank
  – 2-3GB of data transferred for large dataset for each MPI rank on 16-node
  – 800-1500MB of data transferred for large dataset for each MPI rank on 32- and 38-node
• Same data transfer pattern is seen for 16 nodes and 32 nodes
• Data send and data receive are symmetrical
• Data transfers are about halved between 16 nodes and 32 nodes
  – Point-to-point max is 383MB at 32-node versus 155MB at 16-node

16 Node (test_large)

32 Node (test_large)
**GADGET-2 Profiling – Aggregated Transfer**

- **Aggregated data transfer refers to:**
  - Total amount of data being transferred in the network between all MPI ranks collectively

- **For small dataset:**
  - The total data transfer stays around 10GB for small node counts
  - The total data transfer jumps after larger to 20GB after 16 node

- **For large dataset:**
  - The total data transfer remains at around 600GB
GADGET-2 – Summary

- **GADGET-2** is the code for cosmological simulations of structure formation
  - Shows good scalability by using compute nodes to reduce runtime of GADGET-2
- **InfiniBand QDR enables better job productivity when running on multiple nodes**
  - Gigabit Ethernet shows no work gain beyond 2 nodes
- **Intel MPI and Platform MPI performs slight better than Open MPI**
  - Using KNEM in Open MPI does not show benefit for GADGET-2
  - Using “bind-to-core” shows improvement of 47% at 38-node
- **Using nodes with higher CPU frequency enables higher job productivity**
- **MPI Profiling**
  - MPI_Sendrecv has the most number of MPI calls
  - MPI_Barrier is the biggest time consumer in MPI calls
  - Majority of messages are small messages between 0 and 64 bytes
  - Majority of the message sizes for MPI data communications are between 1MB to 4MB
  - 2-3GB of data transfers between MPI ranks on 16-node
  - Data transfer spreads out as cluster scales
Thank You

HPC Advisory Council