Experience with 100Gbps Network Applications

Mehmet Balman, Eric Pouyoul, Yushu Yao, E. Wes Bethel, Burlen Loring, Prabhat, John Shalf, Alex Sim, and Brian L. Tierney

DIDC – Delft, the Netherlands
June 19, 2012
Experience with 100Gbps Network Applications

Mehmet Balman
Computational Research Division
Lawrence Berkeley National Laboratory

DIDC – Delft, the Netherlands
June 19, 2012
Outline

• A recent 100Gbps demo by ESnet and Internet2 at SC11

• Two applications:
  • Visualization of remotely located data (Cosmology)
  • Data movement of large datasets with many files (Climate analysis)

Our experience in application design issues and host tuning strategies to scale to 100Gbps rates
The Need for 100Gbps networks

Modern science is **Data driven** and **Collaborative** in nature

- The largest collaborations are most likely to depend on distributed architectures.
  - **LHC** (distributed architecture) data generation, distribution, and analysis.
  - The volume of data produced by genomic sequencers is rising exponentially.
  - In **climate** science, researchers must analyze observational and simulation data located at facilities around the world.
ESG (Earth Systems Grid)

- Over 2,700 sites
- 25,000 users

- IPCC Fifth Assessment Report (AR5) 2PB
- IPCC Forth Assessment Report (AR4) 35TB
100Gbps networks arrived

- Increasing network bandwidth is an important step toward tackling ever-growing scientific datasets.

- 1Gbps to 10Gbps transition (10 years ago)
  - Application did not run 10 times faster because there was more bandwidth available

In order to take advantage of the higher network capacity, we need to pay close attention to the application design and host tuning issues
Applications’ Perspective

• Increasing the bandwidth is not sufficient by itself; we need careful evaluation of high-bandwidth networks from the applications’ perspective.

• Real time streaming and visualization of cosmology data
  • How high network capability enables remotely located scientists to gain insights from large data volumes?

• Data distribution for climate science
  • How scientific data movement and analysis between geographically disparate supercomputing facilities can benefit from high-bandwidth networks?
The SC11 100Gbps demo

- 100Gbps connection between ANL (Argon), NERSC at LBNL, ORNL (Oak Ridge), and the SC11 booth (Seattle)
Demo Configuration

RRT:

Seattle – NERSC  16ms
NERSC – ANL     50ms
NERSC – ORNL    64ms
Visualizing the Universe at 100Gbps

- **VisaPult** for streaming data
- **Paraview** for rendering

- For visualization purposes, occasional packet loss is acceptable: (using UDP)

- 90Gbps of the bandwidth is used for the full dataset
  - 4x10Gbps NICs (4 hosts)
- 10Gbps of the bandwidth is used for 1/8 of the same dataset
  - 10Gbps NIC (one host)
Demo Configuration

The 1Gbps connection is used for synchronization and communication of the rendering application, not for transfer of the raw data.
UDP shuffling

- UDP packets include position (x, y, z) information (1024^3 matrix)

- MTU is 9000. The largest possible packet size under 8972 bytes (MTU size minus IP and UDP headers)

- 560 points in each packet, 8968 bytes
  - 3 integers (x,y,z) + a floating point value

In the final run n=560, packet size is 8968 bytes
Data flow

• For each time step, the input data is split into 32 streams along the z-direction; each stream contains a contiguous slice of the size $1024 \times 1024 \times 32$.
• 32 streams for 90Gbps demo (high-bandwidth)
• 4 streams for 10Gbps demo (low bandwidth, 1/8 of the data)
Performance Optimization

• Each 10Gbps NIC in the system is bound to a specific core.
• Receiver processes are also bound to the same core.
• Renderer same NUMA node but different core (accessing to the same memory region)
Network Utilization

- 2.3TB of data from NERSC to SC11 booth in Seattle in ≈ 3.4 minutes
- For each timestep, corresponding to 16GB of data, it took ≈ 1.4 seconds to transfer and ≈ 2.5 seconds for rendering before the image was updated.
- Peak ≈ 99Gbps.
- Average ≈ 85Gbps
Demo
Climate Data Distribution

- ESG data nodes
  - Data replication in the ESG Federation

- Local copies
  - data files are copied into temporary storage in HPC centers for post-processing and further climate analysis.
Climate Data over 100Gbps

- Data volume in climate applications is increasing exponentially.
- An important challenge in managing ever increasing data sizes in climate science is the large variance in file sizes.
  - Climate simulation data consists of a mix of relatively small and large files with irregular file size distribution in each dataset.
    - Many small files
Keep the data channel full

- Concurrent transfers
- Parallel streams
lots-of-small-files problem! file-centric tools?

- Not necessarily high-speed (same distance)
  - Latency is still a problem

100Gbps pipe  10Gbps pipe

request a dataset

send data
Block-based

Front-end threads (access to memory blocks)

memory caches are logically mapped between client and server
Moving climate files efficiently
Advantages

• Decoupling I/O and network operations
  • front-end (I/O, processing)
  • back-end (networking layer)

• Not limited by the characteristics of the file sizes
  On the fly tar approach, bundling and sending many files together

• Dynamic data channel management
  Can increase/decrease the parallelism level both in the network communication and I/O read/write operations, without closing and reopening the data channel connection (as is done in regular FTP variants).
Demo Configuration

Disk to memory / reading from GPFS (NERSC), max 120Gbps read performance
The SC11 100Gbps Demo

- CMIP3 data (35TB) from the GPFS filesystem at NERSC
  - Block size 4MB
  - Each block’s data section was aligned according to the system pagesize.
  - 1GB cache both at the client and the server

- At NERSC, 8 front-end threads on each host for reading data files in parallel.
- At ANL/ORNL, 4 front-end threads for processing received data blocks.
- 4 parallel TCP streams (four back-end threads) were used for each host-to-host connection.
83Gbps throughput
MemzNet: memory-mapped zero-copy Network channel

memory caches are logically mapped between client and server
ANI 100Gbps testbed

ANI Middleware Testbed

Note: ANI 100G routers and 100G wave available till summer 2012; Testbed resources after that subject funding availability.
ANI 100Gbps testbed

SC11 100Gbps demo
ANI testbed 100Gbps (10x10NICs, three hosts): Throughput vs the number of parallel streams [1, 2, 4, 8, 16, 32, 64 streams - 5min intervals], TCP buffer size is 50M
ANI testbed 100Gbps (10x10NICs, three hosts): Interface traffic vs the number of concurrent transfers [1, 2, 4, 8, 16, 32, 64 concurrent jobs - 5min intervals], TCP buffer size is 50M
Performance at SC11 demo

TCP buffer size is set to 50MB
Performance results in the 100Gbps ANI testbed
Host tuning

With proper tuning, we achieved 98 Gbps using only 3 sending hosts, 3 receiving hosts, 10 10GE NICS, and 10 TCP flows

<table>
<thead>
<tr>
<th>Host Name</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>nersc-diskpt-1-v4012</td>
<td>1179.1875 MB / 1.00 sec = 9891.8010 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-1-v4013</td>
<td>1179.2500 MB / 1.00 sec = 9888.4787 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-1-v4014</td>
<td>1179.1875 MB / 1.00 sec = 9891.1482 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-1-v4015</td>
<td>1179.1250 MB / 1.00 sec = 9891.1581 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-2-v4012</td>
<td>1179.2500 MB / 1.00 sec = 9891.9494 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-2-v4013</td>
<td>1179.0625 MB / 1.00 sec = 9891.1580 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-2-v4014</td>
<td>1179.3750 MB / 1.00 sec = 9893.1365 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-2-v4015</td>
<td>1179.1250 MB / 1.00 sec = 9891.0690 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-3-v4014</td>
<td>1121.8750 MB / 1.00 sec = 9410.9602 Mbps</td>
<td>0 retrans</td>
</tr>
<tr>
<td>nersc-diskpt-3-v4015</td>
<td>1121.8750 MB / 1.00 sec = 9410.9884 Mbps</td>
<td>0 retrans</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Octets</td>
<td>18462079</td>
</tr>
<tr>
<td>Packets</td>
<td>184615</td>
</tr>
<tr>
<td>Errors</td>
<td>0</td>
</tr>
<tr>
<td>Utilization (% of port capacity)</td>
<td>0.17</td>
</tr>
</tbody>
</table>
NIC/TCP Tuning

• We are using Myricom 10G NIC (100Gbps testbed)
  • Download latest drive/firmware from vendor site
    • Version of driver in RHEL/CentOS fairly old
  • Enable MSI-X
  • Increase txgqueuelen
    
    /sbin/ifconfig eth2 txgqueuelen 10000
  • Increase Interrupt coalescence
    
    /usr/sbin/ethtool -C eth2 rx-usecs 100

• TCP Tuning:
  net.core.rmem_max = 67108864
  net.core.wmem_max = 67108864
  net.core.netdev_max_backlog = 250000
100Gbps = It’s full of frames!

- **Problem:**
  - Interrupts are very expensive
  - Even with jumbo frames and driver optimization, there is still too many interrupts.

- **Solution:**
  - Turn off Linux irqbalance (chkconfig irqbalance off)
  - Use `/proc/interrupt` to get the list of interrupts
  - Dedicate an entire processor core for each 10G interface
  - Use `/proc/irq/<irq-number>/smp_affinity` to bind rx/tx queues to a specific core.
Host Tuning Results

- **Interrupt coalescing (TCP)**: 24% → 36.8%, % Improvement: 53.3333333
- **Interrupt coalescing (UDP)**: 21.1% → 38.8%, % Improvement: 83.8862559
- **IRQ Binding (TCP)**: 30.6% → 36.8%, % Improvement: 20.2614379
- **IRQ Binding (UDP)**: 27.9% → 38.5%, % Improvement: 37.9928315
Conclusion

• Host tuning & host performance
• Multiple NICs and multiple cores
• The effect of the application design

• TCP/UDP buffer tuning, using jumbo frames, and interrupt coalescing.
• Multi-core systems: IRQ binding is now essential for maximizing host performance.
Acknowledgements

Peter Nugent, Zarija Lukic, Patrick Dorn, Evangelos Chaniotakis, John Christman, Chin Guok, Chris Tracy, Lauren Rotman, Jason Lee, Shane Canon, Tina Declerck, Cary Whitney, Ed Holohan, Adam Scovel, Linda Winkler, Jason Hill, Doug Fuller, Susan Hicks, Hank Childs, Mark Howison, Aaron Thomas, John Dugan, Gopal Vaswani
Questions?

Contact:

• Mehmet Balman mbalman@lbl.gov
• Brian Tierney bltierney@es.net