Performance Evaluation of I/O Traffic and Placement of I/O Nodes on a High Performance Network

Salvador Coll*†, Fabrizio Petrini*, Eitan Frachtenberg* and Adolfy Hoisie*

*CCS-3 Modeling, Algorithms, and Informatics
Los Alamos National Laboratory
†Digital Systems Design and Parallel Architectures Groups
Technical University of Valencia - SPAIN
scoll@lanl.gov
Outline

- Introduction
- Quadrics network design overview
- Experimental framework
- Experimental results
- Conclusions
Introduction

- Common trend in large-scale clusters: high performance data networks
- I/O can be limited by the interconnect performance
Introduction

- Common trend in large-scale clusters: high performance data networks
- I/O can be limited by the interconnect performance
- Open problems:
  - influence of the I/O servers placement
  - effect of using dedicated or shared I/O servers
  - potential interference of background I/O traffic with computation
Introduction

- Some of the most powerful systems in the world use the Quadrics interconnection network:
- The Terascale Computing System (TCS) at the Pittsburgh Supercomputing Center – the second most powerful computer in the world
Introduction

- Some of the most powerful systems in the world use the Quadrics interconnection network:

- The Terascale Computing System (TCS) at the Pittsburgh Supercomputing Center – the second most powerful computer in the world

- ASCI Q machine, currently under development at Los Alamos National Laboratory (30 TeraOps, expected to be delivered by the end of 2002)
Introduction

- Some of the most powerful systems in the world use the Quadrics interconnection network:
- The Terascale Computing System (TCS) at the Pittsburgh Supercomputing Center – the second most powerful computer in the world
- ASCI Q machine, currently under development at Los Alamos National Laboratory (30 TeraOps, expected to be delivered by the end of 2002)
- Objective: experimental evaluation of a Quadrics-based cluster under I/O traffic
Quadrics Network Design Overview

- Fat-tree
- Based on 4x4 switches
- Wormhole switching
- 2 virtual channels per physical link
- Adaptive routing
Quadrics Network Design Overview

- Fat-tree
- Based on 4x4 switches
- Wormhole switching
- 2 virtual channels per physical link
- Adaptive routing

Some of the most important aspects of this network are:

- the integration of the local memory into a distributed virtual shared memory,
- the support for zero-copy remote DMA transactions and
- the hardware support for collective communication.
Experimental Framework

- The experimental results are obtained on a 64-node cluster of Compaq AlphaServer ES40s running Tru64 Unix.
- Each Alphaserver is attached to a quaternary fat-tree of dimension three through a 64 bit, 33 MHz PCI bus using the Elan3 card.
- In order to expose the real network performance, we place the communication buffers in Elan memory.
Experimental Results

- We present:
  - unidirectional and bidirectional ping results, as a reference, and
  - single hot-spot
  - multiple hot-spots
  - combined traffic: I/O plus uniform traffic
Unidirectional Ping

Ping Bandwidth

- **Peak data bandwidth (Elan to Elan)** of 335 MB/s $\approx$ 396 MB/s
- **Main to main memory asymptotic bandwidth** of 200 MB/s
Unidirectional Ping

Latency (µs)

Message Size (bytes)

- Latency of 2.4 µs up to 64-byte messages (Elan to Elan memory)
- Higher MPI latency due to message tag matching
Bidirectional Ping

Bidirectional Ping Bandwidth

- Peak data bandwidth (Elan to Elan memory) of **280 MB/s**
- Main to main memory asymptotic bandwidth of **80 MB/s**
Bidirectional Ping

Bidirectional Ping Latency

Latency $\mu$s

- MPI
- Elan3, Elan to Elan
- Elan3, Main to Main

Latency of 4 $\mu$s up to 64-byte messages (Elan to Elan memory)
Ping Summary

### Ping Bandwidth

<table>
<thead>
<tr>
<th>Message Size (bytes)</th>
<th>Bandwidth (MB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>50</td>
</tr>
<tr>
<td>16</td>
<td>100</td>
</tr>
<tr>
<td>64</td>
<td>150</td>
</tr>
<tr>
<td>256</td>
<td>200</td>
</tr>
<tr>
<td>1K</td>
<td>250</td>
</tr>
<tr>
<td>4K</td>
<td>300</td>
</tr>
<tr>
<td>16K</td>
<td>350</td>
</tr>
<tr>
<td>64K</td>
<td>400</td>
</tr>
<tr>
<td>256K</td>
<td>450</td>
</tr>
<tr>
<td>1M</td>
<td>500</td>
</tr>
<tr>
<td>4M</td>
<td>550</td>
</tr>
</tbody>
</table>

- **MPI**
- **Elan3, Elan to Elan**
- **Elan3, Main to Main**

### Bidirectional Ping Bandwidth

<table>
<thead>
<tr>
<th>Message Size (bytes)</th>
<th>Bandwidth (MB/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>50</td>
</tr>
<tr>
<td>16</td>
<td>100</td>
</tr>
<tr>
<td>64</td>
<td>150</td>
</tr>
<tr>
<td>256</td>
<td>200</td>
</tr>
<tr>
<td>1K</td>
<td>250</td>
</tr>
<tr>
<td>4K</td>
<td>300</td>
</tr>
<tr>
<td>16K</td>
<td>350</td>
</tr>
<tr>
<td>64K</td>
<td>400</td>
</tr>
<tr>
<td>256K</td>
<td>450</td>
</tr>
<tr>
<td>1M</td>
<td>500</td>
</tr>
<tr>
<td>4M</td>
<td>550</td>
</tr>
</tbody>
</table>

- **MPI**
- **Elan3, Elan to Elan**
- **Elan3, Main to Main**

### Performance Summary

<table>
<thead>
<tr>
<th>Memory Type</th>
<th>Unidirectional</th>
<th>Bidirectional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elan Memory</td>
<td>335 MB/s</td>
<td>280 MB/s</td>
</tr>
<tr>
<td>Main Memory</td>
<td>200 MB/s</td>
<td>80 MB/s</td>
</tr>
</tbody>
</table>

Workshop on Communication Architecture for Clusters ’02 – p.13
Hot-spot

Objective: analyze the behavior of a single I/O node
Hot-spot

Traffic: hot-spot - 1m bytes

- T-uniform, S-uniform
- T-uniform, S-exponential
- T-exponential, S-uniform
- T-exponential, S-exponential

Peak data bandwidth > 335 MB/s up to 32 nodes
Hot-spot

Bandwidth delivered to each node unevenly distributed
Multiple Hot-spots

Clustered Mapping

Distributed Mapping

8 compute nodes

I/O node

8 I/O nodes
Multiple Hot-spots

Clustered I/O mapping
Multiple Hot-spots

Distributed I/O mapping
Multiple Hot-spots

Objectives:

- behavior of multiple I/O nodes
- influence of the I/O node (hot-node) mapping: clustered and distributed
- effects of the application mapping: shared I/O and dedicated I/O
- influence of the traffic pattern: random and deterministic
- effect of the I/O read/write ratio
Multiple Hot-spots

I/O Traffic: random - 64 Nodes (8 I/O nodes - clustered)

- Asymptotic bandwidth delivered by each I/O node of 196 MB/s
Multiple Hot-spots

I/O Traffic: random - 64 Nodes (8 I/O nodes - distributed)

- Asymptotic bandwidth of 234 MB/s
Multiple Hot-spots

I/O Traffic: deterministic - 64 Nodes (8 I/O nodes - clustered)

- Asymptotic bandwidth of 320 MB/s
Multiple Hot-spots

I/O Traffic: deterministic - 64 Nodes (8 I/O nodes - distributed)

Offered Load per I/O Node (MB/s)

Accepted Load (MB/s)

- Shared I/O - T uniform
- Shared I/O - T exponential
- Dedicated I/O - T uniform
- Dedicated I/O - T exponential

Asymptotic bandwidth of 338 MB/s
Multiple Hot-spots Summary

<table>
<thead>
<tr>
<th></th>
<th>Clustered I/O</th>
<th>Distributed I/O</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Traffic</td>
<td>196 MB/s</td>
<td>234 MB/s</td>
</tr>
<tr>
<td>Deterministic Traffic</td>
<td>320 MB/s</td>
<td>338 MB/s</td>
</tr>
</tbody>
</table>

- Better results obtained with:
  - distributed I/O
  - deterministic traffic
- No significant effect of the application mapping
- Insensitive to read/write ratio
- Insensitive to time and message size distributions
Combined Traffic

Clustered Mapping

Dedicated I/O

Shared I/O

I/O or Uniform Traffic

Uniform Traffic or I/O

Distributed Mapping

Shared I/O

Dedicated I/O

I/O

8 compute nodes

I/O node

8 I/O nodes

Objective:

- interference of the I/O on a parallel job
Combined Traffic

Traffic pattern: uniform - 32 Nodes

Accepted Load (MB/s)

Offered Load (MB/s)

T-uniform, S-uniform
T-uniform, S-exponential
T-exponential, S-uniform
T-exponential, S-exponential

Uniform traffic with no background I/O. Results for 32 nodes.
Combined Traffic with Shared I/O

Clustered−1i Mapping

Clustered−1c Mapping

Distributed Mapping

8 compute nodes    I/O node    8 I/O nodes
Combined Traffic with Shared I/O

I/O load = 0.1

Combined Traffic - 64 Nodes

Accepted Load (MB/s) vs Offered Load (MB/s)

- clustered - 1i
- clustered - 1c
- distributed

Bandwidth delivered by each compute node.
Combined Traffic with Shared I/O

I/O load = 0.3

Bandwidth delivered by each compute node.
Combined Traffic with Shared I/O

I/O load = 0.5

Combined Traffic - 64 Nodes

Accepted Load (MB/s)

Offered Load (MB/s)

Bandwidth delivered by each compute node.
Combined Traffic with Dedicated I/O

Dedicated I/O

I/O

Uniform Traffic

Clustered-1i Mapping

Dedicated I/O

Uniform Traffic

I/O

Clustered-1c Mapping

Dedicated I/O

I/O

Uniform Traffic

Distributed Mapping

8 compute nodes

I/O node

8 I/O nodes
Combined Traffic with Dedicated I/O

I/O load = 0.1

Bandwidth delivered by each compute node.
Combined Traffic with Dedicated I/O

I/O load = 0.3

Combined Traffic - 64 Nodes

Bandwidth delivered by each compute node.
Combined Traffic with Dedicated I/O

I/O load = 0.5

Bandwidth delivered by each compute node.
Combined Traffic Summary

Application Mapping

Shared I/O

Distributed

Sensitive to Background I/O Traffic

Clustered

Job Mapping

Compute Job Overlaps with I/O Nodes

I/O Job Overlaps with I/O Nodes

Dedicated I/O

InSensitive to Background I/O Traffic
Conclusions

- A single hot-node (I/O server) can handle, without performance degradation, traffic generated by up to 32 nodes.
Conclusions

- A single hot-node (I/O server) can handle, without performance degradation, traffic generated by up to 32 nodes.

- With multiple I/O servers it is more efficient to distribute them rather than cluster them, with a bandwidth increase of up to 20%.

- The performance is insensitive to both the fraction of I/O reads and writes and to the mapping of the parallel job.
Conclusions

- A single hot-node (I/O server) can handle, without performance degradation, traffic generated by up to 32 nodes.

- With multiple I/O servers it is more efficient to distribute them rather than cluster them, with a bandwidth increase of up to 20%.

- The performance is insensitive to both the fraction of I/O reads and writes and to the mapping of the parallel job.

- Multiple jobs can be run in parallel without interference, as long as these jobs are not mapped on the I/O nodes.

- The I/O job can interfere with the compute job when the latter is mapped on the I/O nodes.
Additional Information

http://www.c3.lanl.gov/~fabrizio/quadrics.html
Quadrics Network Design Overview

- QsNET provides an abstraction of distributed virtual shared memory
- Each process can map a portion of its address space into the global memory
- These address spaces constitutes the virtual shared memory
- This shared memory is fully integrated with the native operating system
- Based on two building blocks:
  - a network interface card called Elan
  - a crossbar switch called Elite
Elan

- SDRAM I/F
- Thread Processor
- \( \mu \) code Processor
- DMA Buffers
- Link Mux
- FIFO 0
- FIFO 1
- Inputter
- 4 Way Set Associative Cache
- MMU & TLB
- Table Walk Engine
- Clock & Statistics Registers
- PCI Interface
- 400 MB/s Bidirectional 200MHz / 10 bits

- Data Bus
- 100 MHz
- 100 MHz
- 66 MHz
- 64 MHz
- 64
- 32
- 28
- 72
- 64
- 64

Los Alamos National Laboratory

Workshop on Communication Architecture for Clusters ’02 – p.42
Elan

Thread Processor
Runs Communication Protocols
32-bit SPARC-based

PCI Interface

Clock & Statistics Registers

Table Walk Engine

MMU & TLB

4 Way Set Associative Cache

Data Bus

100 MHz

66MHz

Thread Processor

Processors

Inputter

PCI Interface

SDRAM I/F

100 MHz

66MHz

200MHz

72

64

64

64

32

28

64

Workshop on Communication Architecture for Clusters ’02 – p.44
Elan

SDRAM I/F
Thread Processor
μcode Processor
DMA Buffers
Inputter

72
64
64
28
28

4 Way Set Associative Cache
MMU & TLB
Table Walk Engine
Clock & Statistics Registers

PCI Interface

100 MHz
66 MHz

TLB Synchronized with Host

10
200 MHz
10

Workshop on Communication Architecture for Clusters '02 – p.45
Elite

- 8 bidirectional links with 2 virtual channels in each direction
- An internal 16x8 full crossbar switch
- 400 MB/s on each link direction
- Packet error detection and recovery, with routing and data transactions CRC protected
- 2 priority levels plus an aging mechanism
- Adaptive routing
- Hardware support for broadcast
Network Topology: Quaternary Fat-Tree
Network Topology: Quaternary Fat-Tree
Network Topology: Quaternary Fat-Tree
Packet Format

- 320 bytes data payload (5 transactions with 64 bytes each)
- 74-80 bytes overhead
Programming Libraries

- Elan3lib
  - event notification
  - memory mapping and allocation
  - remote DMA
- Elanlib and Tports
  - collective communication
  - tagged message passing
- MPI, shmep

User Applications

```
user space

  shmep
  mpi
  elanlib
  tport

kernel space

  elan3lib
  system calls
  elan kernel comms
```