Ateneo de Manila University

Development of a University-Based High Performance Computing System of PC Clusters

Ateneo High Performance Computing Group
8 August 2000
http://www.math.admu.edu.ph/ahpc/
william.s.yu@ieee.org
Section I

Introduction
Supercomputing

★ the ability to perform computational tasks that are far beyond the normal scope of workstations and PCs

★ Usually one generation faster than the most commonly available high performance machines in production

★ able to tackle the so called Grand Challenge Problems that are not widely solved due to the lack of computing resources
Classes of Supercomputers

★ Computer Systems with cutting edge microprocessor technology (SMP machines and High Performance Processors-SPARC, Alpha, PA-RISC)

★ Massively Parallel Systems (Cray, Origin 2000, RS6000SP)

★ Cluster/Network of Workstations
Clustering

★ connecting two or more computers to perform a single task or solve an single problem

★ provide parallel computing muscle to solve compute intensive problems that are parallelizable

★ provides a single system image for users and developers
Current Trends

★ Supercomputing is limited to a few powerful nations
   ★ low availability
   ★ extremely high cost
   ★ lack of expertise
   ★ lack of infrastructure

★ Growing number of compute intensive applications

★ Increasing in micro-processing power following Moore’s Law

★ Lowering of costs for high bandwidth networking hardware
History

- Summer of 1994, Thomas Sterling, Don Becker and team at CESDIS located at the Goddard Space Flight Center, NASA
- 16 486 DX4 processors w/ channel bonded 10mbps Ethernet
- solving n-body gravitational problems
Beowulf Defined

- Multicomputer Architecture
  - scalable
  - low cost
- High Speed Network
- Commodity Off-the-Shelf
- Free and Open Source Software
- Use of Message Passing Libraries
Section II

Applications
Education

Scientific Computing Cluster

★ Parallel Programming in the curriculum
★ Supporting research that require high performance computing
★ Venue for hands-on cutting edge research with distributed computing paradigms
★ Solve compute intensive problems from different fields
Industry

Scientific Computing Cluster

★ reduce of R&D costs by modeling

★ Aspects:
  ★ neural networks
  ★ ray tracing/graphics rendering
  ★ fluid dynamics

★ used for testing and simulation
Commerce

High-Availability Cluster

- High Availability Systems
- Task Distribution
- Fail-Operability, Fault-Tolerance
- Load Balancing
- Distributed I/O Subsystems
Scientific Computing

Scientific Computing Cluster

★ Linear Algebra

★ Queueing Theory

★ Fluid Dynamics

★ Neural Networks
Disclaimer

★ not all problems are parallelizable

★ do not expect all serial code to run parallelly on a cluster

★ coding in a parallel machine is not the same as coding in a serial machine
Section II

Beowulf Architecture
Hardware Component

★ Node Hardware - CPU, Motherboard, Memory, HDD

★ Network Hardware - Topology, NICs, Switches, Hubs

★ Cluster Layout - Connections, Maintainance, Ventilation, Power
Hardware Notes

★ consider price/performance ratio

★ better to add more memory to prevent swapping

★ dependent on application to be developed on the cluster
Software Component

★ Operating System - Open Source, Stable

★ Message Passing Libraries - shields developers from network layer

★ Scientific Libraries - shields developers from the message passing layer for commonly used mathematical and scientific formulas

★ Compilers - GNU tools, Portland Group, NAG

★ Other - NFS, BOOTP, PXE
Support Issues

- Configuration
- Upgradability
- Manageability
- Cluster Binary Propagation
Programming in Parallel

★ w/ BSD and Network Sockets
★ w/ Remote Shell(rsh)
★ w/ Message Passing Libraries
★ w/ Scientific Computing Libraries
Some Problems w/

★ w/ BSD and Network Sockets
★ w/ Remote Shell(rsh)
★ No Network Abstraction
★ No standard for porting code to different systems
★ No Heterogeneous System Support
Message Passing

★ completely separate address space and namespaces

★ library handles all network reliability, transmission, handshaking issues

★ provides a simple programming interface

★ ability to utilized user defined data types and other features
MPI

★ Message Passing Interface
★ standard for creating message passing applications
★ aims to be a practical, portable, efficient, and flexible standard

Goals and Aims of MPI
★ design an application programming interface
★ allow reliable and efficient communications features
★ allow operations in a heterogeneous environment
★ allow convenient bindings to C and Fortran 77
★ define an interface that is not too different from existing standards
★ define an interface that can be implemented on multiple vendor platforms
History

★ Workshop on Standards for Message Passing in a Distributed Memory Environment, sponsored by the Center for Research on Parallel Computing, held April 29-30, 1992, in Williamsburg, Virginia o 60 people from 40 different organizations

★ The workshop aims to develop a standard message passing interface

★ Dongarra, Hempel, Hey, and Walker proposed the first draft(v1.0) in November 1992

★ On June 1995, MPI v1.1 was released by the MPI Forum

★ MPI v2.0 meeting began on April 1995
Flavors of MPI

⭐ MPICH - is a free and portable implementation of the MPI standard developed by MSU/ANL

⭐ LAM/MPI - is another free implementation of the MPI standard and is being developed at ND

⭐ Other Commercial MPI Implementations - distributed usually by the supercomputer vendors (IBM, SUN, etc.)
Basic MPI concepts

- Message - data packet to be exchanged among compute nodes
- Process - computational task to be completed
- Rank - order of the node in the cluster
- Communicator - group of nodes
Scientific Computing Libraries

★ No need to recode commonly used mathematical and scientific routines

★ No need to deal with the low level message passing routines

★ Reduces coding time by promoting code reuse

★ More often than not contain optimized routines for certain tasks

★ PETSc - Portable Extensible Toolkit for Scientific Computing
Section IV

Current Status
Hardware Installed

Master Node Configuration(1):

★ AMD Athlon K7 600Mhz
★ Freetech K7M 200Mhz Motherboard
★ 256 MB SDRAM
★ 6.4 GB IDE Hard Drive
★ (2) Intel Ethernet Express Pro 100+ Network Interface Card
★ 21” Monitor
★ CDROM Drive

Processing Node Configuration(7):

★ AMD Athlon K7 600Mhz
★ Freetech K7M 200Mhz Motherboard
★ 128 MB SDRAM
★ 6.4 GB IDE Hard Drive
★ Intel Ethernet Express Pro 100+ Network Interface Card

Networking Hardware:
★ (1) 24 port 100mbps Intel Express Pro 410T Switch
★ Category 5 UTP cable and RJ-45 connectors

Others:
★ APC SmartUPS Pro 3000
★ 8-way Monitor-Keyboard-Mouse Sharing Switch
# Software Installed

<table>
<thead>
<tr>
<th>Software Used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux Distribution</td>
</tr>
<tr>
<td>Redhat 6.2</td>
</tr>
<tr>
<td>Linux Kernel</td>
</tr>
<tr>
<td>Kernel 2.2.16</td>
</tr>
<tr>
<td>Message Passing Libraries</td>
</tr>
<tr>
<td>PVM 3.4.3, MPICH 1.2.1, LAM 6.3.2</td>
</tr>
<tr>
<td>Scientific Libraries</td>
</tr>
<tr>
<td>LAPACK, SCALAPACK, PETSc</td>
</tr>
<tr>
<td>Benchmark Suites</td>
</tr>
<tr>
<td>HPL, NPB</td>
</tr>
<tr>
<td>Support Software</td>
</tr>
<tr>
<td>NFS, BOOTP, RSH, NTPD</td>
</tr>
<tr>
<td>Others</td>
</tr>
<tr>
<td>MPI-POVRAY</td>
</tr>
</tbody>
</table>

Table 1: Software Currently Installed in the AGILA Cluster
## Benchmarks

### AGILA Results:

<table>
<thead>
<tr>
<th>N</th>
<th>NB</th>
<th>P</th>
<th>Q</th>
<th>Gflops</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>1.02</td>
</tr>
<tr>
<td>5000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.22</td>
</tr>
<tr>
<td>8000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.88</td>
</tr>
<tr>
<td>10000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>3.16</td>
</tr>
</tbody>
</table>

Table 2.A: AHPC AGILA 8-node Athlon Cluster (128MB - Fast Ethernet)
## TORC Results:

<table>
<thead>
<tr>
<th>N</th>
<th>NB</th>
<th>P</th>
<th>Q</th>
<th>Gflops</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>1.76</td>
</tr>
<tr>
<td>5000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.32</td>
</tr>
<tr>
<td>8000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.51</td>
</tr>
<tr>
<td>10000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.58</td>
</tr>
<tr>
<td>15000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.72</td>
</tr>
<tr>
<td>20000</td>
<td>64</td>
<td>2</td>
<td>4</td>
<td>2.73</td>
</tr>
</tbody>
</table>

Table 2.B: UTK/ICL Torc 8-Dual Intel PIII 550Mhz(512MB - Myrinet)
Section V

Future Plans
★ improve network topology
★ develop techniques to ease cluster construction and maintenance
★ develop courseware for teaching parallel computing
★ develop parallel code for existing applications
★ enter into other fields of research such as weather modelling and traffic simulation
Optional Demo
Ateneo de Manila University

Development of a University-Based High Performance Computing System of PC Clusters

Ateneo High Performance Computing Group
8 August 2000
http://www.math.admu.edu.ph/ahpc/
william.s.yu@ieee.org