Cluster and Grid Computing

Cheng-Zhong Xu

What’s a Cluster?

- Collection of independent computer systems working together as if a single system.
- Coupled through a scalable, high bandwidth, low latency interconnect.
PC Clusters: Contributions of Beowulf

- An experiment in parallel computing systems
- Established vision of low cost, high end computing
- Demonstrated effectiveness of PC clusters for some (not all) classes of applications
- Provided networking software
- Conveyed findings to broad community (great PR)
- Tutorials and book
- Design standard to rally community!
- Standards beget:
  - books, trained people,
  - Open source SW

Adapted from Gordon Bell, presentation at Salishan 2000

Towards Inexpensive Supercomputing

It is:

Cluster Computing..

The Commodity Supercomputing!
Scalable Parallel Computers

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>MPP</th>
<th>SMP, CC-NUMA</th>
<th>Cluster</th>
<th>Distributed System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>O(100) - O(1000)</td>
<td>O(10) - O(100)</td>
<td>O(100) or less</td>
<td>O(10) - O(1000)</td>
</tr>
<tr>
<td>Node Complexity</td>
<td>Fine or medium</td>
<td>Medium or coarse</td>
<td>Medium</td>
<td>Wide range</td>
</tr>
<tr>
<td>Interface Communication</td>
<td>Message passing</td>
<td>Centralized and</td>
<td>Message</td>
<td>Shared files, RPC,</td>
</tr>
<tr>
<td></td>
<td>of shared variables</td>
<td>distributed shared</td>
<td>Passing</td>
<td>message passing,</td>
</tr>
<tr>
<td></td>
<td>for DSM</td>
<td>memory</td>
<td></td>
<td>IPC protocol</td>
</tr>
<tr>
<td>Job Scheduling</td>
<td>Single run queue at host</td>
<td>Single run queue mostly</td>
<td>Multiple queues but coordinated</td>
<td>Independent multiple Queues</td>
</tr>
<tr>
<td>SSI Support</td>
<td>Partially</td>
<td>Always in SMP and some NUMA</td>
<td>Desired</td>
<td>No</td>
</tr>
<tr>
<td>Node OS, Cores and Type</td>
<td>N microkernels</td>
<td>One monolithic for SMP and multiple for NUMA</td>
<td>N OS platforms (Homogeneous or heterogeneous)</td>
<td>N OS platforms (Heterogeneous)</td>
</tr>
<tr>
<td>Address Space</td>
<td>Multiple</td>
<td>Single</td>
<td>Multiple or single</td>
<td>Multiple</td>
</tr>
<tr>
<td>Interconnect Security</td>
<td>Unnecessary</td>
<td>Unnecessary</td>
<td>Required or optional</td>
<td>Required</td>
</tr>
<tr>
<td>Ownership</td>
<td>One organization</td>
<td>One organization</td>
<td>One or More organizations</td>
<td>Many organizations</td>
</tr>
<tr>
<td>Network Protocol</td>
<td>Nonstandard</td>
<td>Nonstandard</td>
<td>Standard &amp; Lightweight</td>
<td>Standard</td>
</tr>
</tbody>
</table>

Design Space of Competing Computer Architecture

Size Scalability

Distributed System

Future Cluster

MPP

CC-NUMA

SMP

Present Cluster

Single System Image
Clusters of SMPs

- SMPs are the fastest commodity machine, so use them as a building block for a larger machine with a network
- Common names:
  - CLUMP = Cluster of SMPs
  - Hierarchical machines, constellations
Shared Pool of Computing Resources:
Processors, Memory, Disks

Interconnect

Guarantee at least one workstation to many individuals (when active)
Deliver large % of collective resources to few individuals at any one time

Opportunity of Large-scale Computing on NOW
Windows of Opportunities

- **MPP/DSM:**
  - Compute across multiple systems: parallel.

- **Network RAM:**
  - Idle memory in other nodes. Page across other nodes idle memory

- **Software RAID:**
  - file system supporting parallel I/O and reliability, mass-storage.

- **Multi-path Communication:**
  - Communicate across multiple networks: Ethernet, ATM, Myrinet

Parallel Processing

- Scalable Parallel Applications require
  - good floating-point performance
  - low overhead communication scalable network bandwidth
  - parallel file system
Network RAM

- Performance gap between processor and disk has widened.

- Thrashing to disk degrades performance significantly

- Paging across networks can be effective with high performance networks and OS that recognizes idle machines

- Typically thrashing to network RAM can be 5 to 10 times faster than thrashing to disk

Software RAID: Redundant Array of Workstation Disks

- I/O Bottleneck:
  - Microprocessor performance is improving more than 50% per year.
  - Disk access improvement is < 10%
  - Application often perform I/O

- RAID cost per byte is high compared to single disks

- RAIDs are connected to host computers which are often a performance and availability bottleneck

- RAID in software, writing data across an array of workstation disks provides performance and some degree of redundancy provides availability.
Cluster Computer and its Components
Clustering Today

- Clustering gained momentum when 3 technologies converged:
  - 1. Very HP Microprocessors
    - workstation performance = yesterday supercomputers
  - 2. High speed communication
    - Comm. between cluster nodes >= between processors in an SMP.
Cluster Components...1a
Nodes

- Multiple High Performance Components:
  - PCs
  - Workstations
  - SMPs (CLUMPS)
  - Distributed HPC Systems leading to Metacomputing

- They can be based on different architectures and running difference OS

Cluster Components...1b
Processors

- There are many (CISC/RISC/VLIW/Vector..)
  - Intel: Pentiums, Xeon, Merceed…
  - Sun: SPARC, ULTRASPARC
  - HP PA
  - IBM RS6000/PowerPC
  - SGI MIPS
  - Digital Alphas

- Integrate Memory, processing and networking into a single chip
  - IRAM (CPU & Mem): (http://iram.cs.berkeley.edu)
  - Alpha 21366 (CPU, Memory Controller, NI)
Cluster Components...2
OS

- State of the art OS:
  - Linux (Beowulf)
  - Microsoft NT (Illinois HPVM)
  - SUN Solaris (Berkeley NOW)
  - IBM AIX (IBM SP2)
  - HP UX (Illinois - PANDA)
  - Mach (Microkernel based OS) (CMU)
  - Cluster Operating Systems (Solaris MC, SCO Unixware, MOSIX (academic project)
  - OS gluing layers: (Berkeley Glunix)

Cluster Components...3
High Performance Networks

- Ethernet (10Mbps),
- Fast Ethernet (100Mbps),
- Gigabit Ethernet (1Gbps)
- SCI (Dolphin - MPI- 12micro-sec latency)
- ATM
- Myrinet (1.2Gbps)
- Digital Memory Channel
- FDDI
Cluster Components...4

Network Interfaces

- Network Interface Card
  - Myrinet has NIC
  - User-level access support
  - Alpha 21364 processor integrates processing, memory controller, network interface into a single chip..

Cluster Components...5

Communication Software

- Traditional OS supported facilities (heavy weight due to protocol processing).
  - Sockets (TCP/IP), Pipes, etc.
- Light weight protocols (User Level)
  - Active Messages (Berkeley)
  - Fast Messages (Illinois)
  - U-net (Cornell)
  - XTP (Virginia)
- System systems can be built on top of the above protocols
Cluster Components...6a
Cluster Middleware

- Resides Between OS and Applications and offers in infrastructure for supporting:
  - Single System Image (SSI)
  - System Availability (SA)

- SSI makes collection appear as single machine (globalised view of system resources). Telnet cluster.myinstitute.edu

- SA - Check pointing and process migration..

Cluster Components...6b
Middleware Components

- Hardware
  - DEC Memory Channel, DSM (Alewife, DASH) SMP Techniques

- OS / Gluing Layers
  - Solaris MC, Unixware, Glunix

- Applications and Subsystems
  - System management and electronic forms
  - Runtime systems (software DSM, PFS etc.)
  - Resource management and scheduling (RMS):
    - CODINE, LSF, PBS, NQS, etc.
Cluster Components... 7a
Programming environments

- Threads (PCs, SMPs, NOW..)
  - POSIX Threads
  - Java Threads
- MPI
  - Linux, NT, on many Supercomputers
- PVM
- Software DSMs (Shmem)

Cluster Components... 7b
Development Tools?

- Compilers
  - C/C++/Java /
  - Parallel programming with C++ (MIT Press book)
- RAD (rapid application development tools).. GUI based tools for PP modeling
- Debuggers
- Performance Analysis Tools
- Visualization Tools
Cluster Components...8
Applications

- Sequential
- Parallel / Distributed (Cluster-aware app.)
  - Grand Challenging applications
    - Weather Forecasting
    - Quantum Chemistry
    - Molecular Biology Modeling
    - Engineering Analysis (CAD/CAM)
    - PDBs, web servers, data-mining

Key Operational Benefits of Clustering

- **System availability (HA).** Offer inherent high system availability due to the redundancy of hardware, operating systems, and applications.
- **Hardware Fault Tolerance.** Redundancy for most system components (e.g., disk-RAID), including both hardware and software.
- **OS and application reliability.** Run multiple copies of the OS and applications, and through this redundancy
- **Scalability.** Adding servers to the cluster or by adding more clusters to the network as the need arises or CPU to SMP.
- **High Performance.** (Running cluster enabled programs)
Classification of Cluster Computer

Clusters Classification..1

- Based on Focus (in Market)
  - High Performance (HP) Clusters
    • Grand Challenging Applications
  - High Availability (HA) Clusters
    • Mission Critical applications
HA Cluster: Server Cluster with "Heartbeat" Connection

Clusters Classification.2

- Based on Workstation/PC Ownership
  - Dedicated Clusters
  - Non-dedicated clusters
    - Adaptive parallel computing
    - Also called Communal multiprocessing
Clusters Classification..3

- Based on Node Architecture..
  - Clusters of PCs (CoPs)
  - Clusters of Workstations (COWs)
  - Clusters of SMPs (CLUMPs)

Clusters Classification..4

- Based on Node OS Type..
  - Linux Clusters (Beowulf)
  - Solaris Clusters (Berkeley NOW)
  - NT Clusters (HPVM)
  - AIX Clusters (IBM SP2)
  - SCO/Compaq Clusters (Unixware)
  - …….Digital VMS Clusters, HP clusters,
Clusters Classification

- Based on node components architecture & configuration (Processor Arch, Node Type: PC/Workstation.. & OS: Linux/NT..):
  - Homogeneous Clusters
    - All nodes will have similar configuration
  - Heterogeneous Clusters
    - Nodes based on different processors and running different OSes.

Dimensions of Scalability & Levels of Clustering:

1. Network Technology
   - Uniprocessor
   - SMP
   - Cluster
   - MPP
   - Metacomputing (GRID)

2. CPU / I/O / Memory / OS

3. Platform
   - Campus
   - Enterprise
   - Workgroup
   - Department
   - Public
Clusters Classification..6b
Levels of Clustering

- Group Clusters (#nodes: 2-99)
  - (a set of dedicated/non-dedicated computers - mainly connected by SAN like Myrinet)
- Departmental Clusters (#nodes: 99-999)
- Organizational Clusters (#nodes: many 100s)
- (using ATMs Net)
- Internet-wide Clusters=Global Clusters: (#nodes: 1000s to many millions)
  - Metacomputing
  - Web-based Computing
  - Agent Based Computing
    - Java plays a major in web and agent based computing

Major issues in cluster design

- Size Scalability (physical & application)
- Enhanced Availability (failure management)
- Single System Image (look-and-feel of one system)
- Fast Communication (networks & protocols)
- Load Balancing (CPU, Net, Memory, Disk)
- Security and Encryption (clusters of clusters)
- Distributed Environment (Social issues)
- Manageability (admin. And control)
- Programmability (simple API if required)
- Applicability (cluster-aware and non-aware app.)
Cluster Middleware and Single System Image

A typical Cluster Computing Environment

Application

PVM / MPI / RSH

Hardware/OS
**CC should support**

- Multi-user, time-sharing environments
- Nodes with different CPU speeds and memory sizes (heterogeneous configuration)
- Many processes, with unpredictable requirements
- Unlike SMP: insufficient “bonds” between nodes
  - Each computer operates independently
  - Inefficient utilization of resources

The missing link is provided by cluster middleware/underware

Application
**SSI Clusters--SMP services on a CC**

“Pool Together” the “Cluster-Wide” resources

- Adaptive resource usage for better performance
- Ease of use - almost like SMP
- Scalable configurations - by decentralized control

Result: *HPC/HAC at PC/Workstation prices*

---

**What is Cluster Middleware?**

- An interface between between use applications and cluster hardware and OS platform.
- Middleware packages support each other at the management, programming, and implementation levels.
- Middleware Layers:
  - SSI Layer
  - Availability Layer: It enables the cluster services of
    - Checkpointing, Automatic Failover, recovery from failure,
    - fault-tolerant operating among all cluster nodes.
**Middleware Design Goals**

- **Complete Transparency (Manageability)**
  - Lets the see a single cluster system..
    - Single entry point, ftp, telnet, software loading...

- **Scalable Performance**
  - Easy growth of cluster
    - no change of API & automatic load distribution.

- **Enhanced Availability**
  - Automatic Recovery from failures
    - Employ checkpointing & fault tolerant technologies
  - Handle consistency of data when replicated..

---

**What is Single System Image (SSI) ?**

- A single system image is the *illusion*, created by software or hardware, that presents a collection of resources as one, more powerful resource.

- SSI makes the cluster appear like a single machine to the user, to applications, and to the network.

- A cluster without a SSI is not a cluster
Benefits of Single System Image

- Usage of system resources transparently
- Transparent process migration and load balancing across nodes.
- Improved reliability and higher availability
- Improved system response time and performance
- Simplified system management
- Reduction in the risk of operator errors
- User need not be aware of the underlying system architecture to use these machines effectively

Desired SSI Services

- Single Entry Point
  - telnet cluster.wayne.edu
  - telnet node1.cluster.wayne.edu
- Single File Hierarchy: xFS, AFS, Solaris MC Proxy
- Single Control Point: Management from single GUI
- Single virtual networking
- Single memory space - Network RAM / DSM
- Single Job Management: Glunix, Codine, LSF
- Single User Interface: Like workstation/PC windowing environment (CDE in Solaris/NT), may it can use Web technology
**Availability Support Functions**

- **Single I/O Space (SIO):**
  - any node can access any peripheral or disk devices without the knowledge of physical location.

- **Single Process Space (SPS)**
  - Any process on any node create process with cluster wide process wide and they communicate through signal, pipes, etc, as if they are one a single node.

- **Checkpointing and Process Migration.**
  - Saves the process state and intermediate results in memory to disk to support rollback recovery when node fails. PM for Load balancing...

- **Reduction in the risk of operator errors**

- **User need not be aware of the underlying system architecture to use these machines effectively**

---

**Scalability Vs. Single System Image**

![Graph showing the relationship between scalability and single system image](image)
SSI Levels/How do we implement SSI?

- It is a computer science notion of levels of abstractions (house is at a higher level of abstraction than walls, ceilings, and floors).

- Application and Subsystem Level
- Operating System Kernel Level
- Hardware Level

Cluster Programming Environments: Example

- Shared Memory Based
  - DSM
  - Threads/OpenMP (enabled for clusters)
  - Java threads (HKU JESSICA, IBM cJVM)

- Message Passing Based
  - PVM (PVM)
  - MPI (MPI)

- Parametric Computations
  - Nimrod/Clustor

- Automatic Parallelising Compilers

- Parallel Libraries & Computational Kernels (NetSolve)
**Levels of Parallelism**

- **PVM/MPI**
  - Task i-1
  - Task i
  - Task i+1

- **Threads**
  - func1()
  - func2()
  - func3()

- **Compilers**
  - a(0) = ...
  - b(0) = ...
  - a(1) = ...
  - b(1) = ...
  - a(2) = ...
  - b(2) = ...

- **CPU**
  - +
  - x
  - Load

**Code-Granularity**
- Code Item
- Large grain (task level)
- Program

- Medium grain (control level)
- Function (thread)

- Fine grain (data level)
- Loop (Compiler)

- Very fine grain (multiple issue)
- With hardware

**Computer Food Chain (Now and Future)**


What Next??

Clusters of Clusters and Grid Computing

Computational/Data Grid

- Coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations
  - direct access to computers, sw, data, and other resources, rather than file exchange
  - Such sharing rules defines a set of individuals and/or institutions, which form a virtual organization
  - Examples of VOs: application service providers, storage service providers, cycle providers, etc
- Grid computing is to develop protocols, services, and tools for coordinated resource sharing and problem solving in VOs
  - Security solutions for management of credentials and policies
  - RM protocols and services for secure remote access
  - Information query protocols and services for configuration
  - Data management, etc
Scalable Computing

Administrative Barriers
- Individual
- Group
- Department
- Campus
- State
- National
- Globe
- Inter Planet
- Universe

Figure due to Rajkumar Buyya, University of Melbourne, Australia, www.gridbus.org