# Cluster Computing Tools, Applications, and Australian Initiatives for Low Cost Supercomputing

Hai Jin[†], Rajkumar Buyya*, Mark Baker[§]

Department of Electrical Engineering[†]
University of Southern California,
Los Angeles, California, USA
hjin@ceng.usc.edu

School of Computer Science and Software Eng. *
Monash University,
Melbourne, Australia
rajkumar@csse.monash.edu.au

School of Computer Science[§]
University of Portsmouth,
Portsmouth, Hants, UK
mark.baker@computer.org

**Abstract:** Cluster computing has become the paradigm of choice for executing large-scale science, engineering, and commercial applications. This is due to their low cost, high performance, availability of off-the-shelf hardware components and freely accessible software tools that that can be used for developing cluster applications. In this article, we discuss cluster-computing technologies, system architecture, and the software tools for developing applications. We present recent Australian national and state level initiatives (particularly Victorian) and their focus areas. We conclude with a discussion on future directions in cluster computing and briefly mention the initiatives being led by the IEEE Task Force on Cluster Computing (TFCC).

## 1. Introduction

During the past decade, many different computer systems supporting high performance computing have emerged. Their taxonomy is based on how their processors, memory, and interconnect are laid out. The most common systems include vector computers (VC); massively parallel processors (MPP), symmetric multiprocessors (SMP), cache-coherent non-uniform memory access (CC-NUMA), distributed systems and clusters. A cluster is defined as a collection of interconnected stand-alone workstations or PCs cooperatively working together as a single, integrated computing resource.

Parallel supercomputers have been in the mainstream of high-performance computing for the past decade. However, their popularity is waning. The reasons for this decline are many, but include factors like being expensive to purchase and run, potentially difficult to program, slow to evolve in the face of emerging hardware technologies, and difficult to upgrade without, generally, replacing the whole system. The decline of dedicated parallel supercomputer has been compounded by the emergence of commodity-off-the-shelf clusters of PCs and workstations. Cluster computing is not new, but certain recent technical capabilities, particularly in the area of networking, have brought this class of machine to the vanguard as a platform of choice to run all types of parallel and distributed applications.

The use of clusters as a platform for running high-performance and high-availability applications is increasing mainly due to their cost-effective nature. This is evident by the fact that the performance of cluster components has almost reached the performance of those used in supercomputers in addition the commodity components are improving in terms of performance and functionality all the time. The performance of workstation is doubling every 18 to 24 months. The performance of network

hardware and software is improving with ever increasing bandwidths and later latencies being achieved between cluster nodes. The introduction of LINUX and the Windows NT operating system has led to increasing levels of interest in utilizing PC-based systems as a cost-effective computational resource for parallel computing. The development tools and programming environments for PCs/Workstations are becoming mature and more sophisticated compared and comparable to the contrasting solutions for traditional parallel supercomputers.

While traditional supercomputers are potentially easier to program, the of use parallel computing on cluster systems is increasing, a trend that is likely to continue. Recently many organizations have replaced their traditional or proprietary supercomputers with clusters as a cost saving measure. Cluster-based systems with supercomputing ability are now being assembled with commodity-off-the-shelf (COTS) hardware components and freely available software such as Linux operating systems and parallel programming environments and runtime systems like MPI.

Among the most recent Top500 supercomputer list [12] released at June 2000, HPC 4500 Cluster [13] from Sun Microsystems and CPlant Cluster [14] in Sandia National Laboratories were ranked at No. 50 and 62, respectively. CPlant is a 580-node Compaq XP1000 workstations with 500 MHz 21264 microprocessor each node. The peak LINPACK [15] performance is 580 Gflops/s, similar to that of 2048-node Hitachi/Tsukuba CP-PACS supercomputer [16] in 1996, which ranked the first in Top500 list. With the growing importance of clusters and the increasing number of clusters in Top500 list, a new dedicated top cluster list called Top Clusters [17] has been initiated jointly by the IEEE Task Force on Cluster Computing (TFCC) and the TOP500 teams. The TopClusters list ranks all types of clusters irrespective of their operating systems and configurations by measuring the performance of various parameters including numeric, I/O, database, web, TPC, simulation, application level performance.

## 2. Cluster Architecture

A cluster is a type of parallel or distributed processing system that consists of a collection of interconnected stand-alone computers working together as a single, integrated computing resource [3]. A node the cluster can be a single or multiprocessor system, such as PC, workstation, or SMP. Each node will have its own memory, I/O devices and operating system. A cluster can be in a single cabinet or physically separated and connected via a LAN. Typically a cluster will appear as a single system to users and applications. Figure 1 shows a typical cluster architecture.

In such cluster architecture, the network interface hardware is responsible for transmitting and receiving packets of data between nodes. The communication software should offer a fast and reliable means of data communication between nodes and potentially outside the cluster. For example, clusters with a special network like Myrinet [18] use communication protocol such as Active Messages [19] for fast communication among its nodes. This hardware interface bypasses the operating system and provides direct user-level access to the network interface, thus remove the critical communication overheads.

Cluster middleware is responsible for offering the illusion of a unified system image (single system image) and availability out of a collection of independent but interconnected computers.

Programming environments can offer portable, efficient, and easy-to-use tools for developing applications. Such environments include tools and utilities such as compilers, message passing libraries, debuggers, and profilers. A cluster can execute

both sequential and parallel applications. A comprehensive discussion of cluster-based architectures, programming, and applications can be found in [3] and [4].

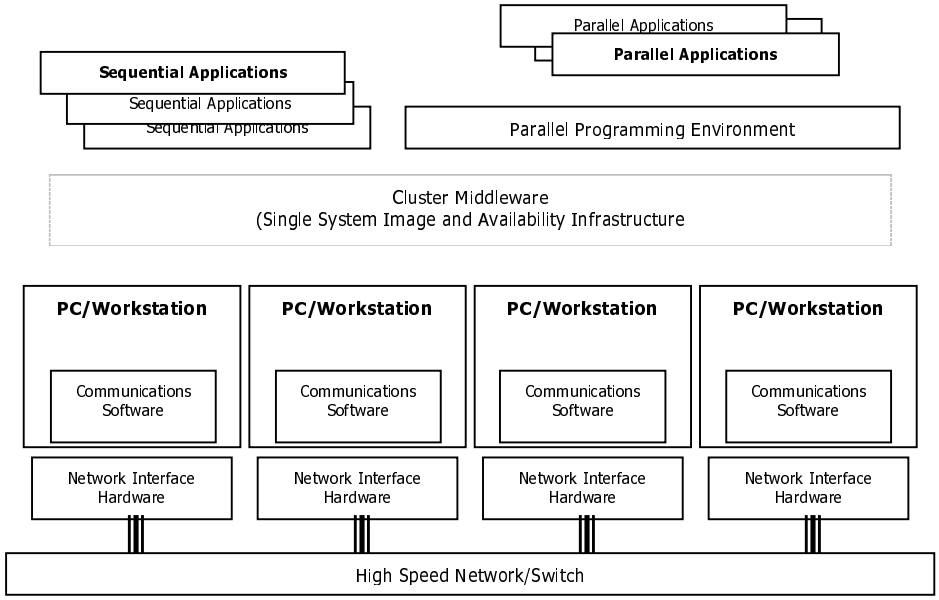


**Figure 1: A Typical Cluster Computer Architecture [3].**

## 3. Linux based Software and Tools for Clustering

Linux [20] is a freely available UNIX-like open operating system that is supported by its users and developers. Linus Torvalds, a Finnish student in 1992, initially developed Linux. Now, Linux has become a robust and reliable POSIX compliant operating system. Several companies have built businesses from packaging Linux software into organized distributions, RedHat [21] is an example of such a company.

Linux provides the features typically found in standard UNIX such as multi-user access, pre-emptive multi-tasking, demand-paged virtual memory and SMP support [22]. In addition to the Linux kernel, a large amount of application and system software and tools are is also freely available. This makes Linux the preferred operating system for clusters.

The most popular compilers used in Linux cluster are GNU C and Fortran compilers [23]. For most applications, `gcc` and `g77` are adequate. However, if an application uses the Fortran programming paradigm and tools, then GNU Fortran is not as robust, nor is there sufficient support for the parallel programming libraries as there is in the commercial Fortran compilers.

Absoft [24] produces Fortran compilers (F77/F90) for Intel-based Linux systems, Apple Macintosh and Microsoft Windows. Absoft also market various mathematical libraries for use with their compilers. The Numerical Algorithms Group (NAG) [25] produces and markets various Fortran solutions. NAG has been providing compilers and libraries for high-performance computers since 1975. Their compilers run on various platforms, including Linux. The Portland Group [26] is a vendor of Fortran (F77/F90) compilers. Portland also produces C and C++ compilers. Portland tailors their systems for use in clustered environments.

Message passing libraries provide a high-level means of passing data between process

executing on distributed memory systems. These libraries are currently at the heart of what makes it possible to achieve high performance out of collections of individual cluster nodes. Message passing libraries typically provide routines to initialise and configure the messaging environments as well as sending the receiving packets of data. The two most popular high-level message-passing systems for scientific and engineering applications are MPI (Message Passing Interface) [28] defined by the MPI Forum [29] and the PVM (Parallel Virtual Machine) [27] from Oak Ridge National Laboratory and the University of Tennessee at Knoxville, USA.

MPI I is the de facto standard for parallel programming, both on clusters, and on traditional parallel supercomputers, such as the Cray T3E and IBM SP2. MPI consists of a rich set of library functions to do both point-to-point and collective communication among parallel tasks. The first version of MPI did not specify the mean of spawning a MPI task on a run-time environment. Generally, however, conventions have been adopted by most MPI implementations. For example, the command `mpirun` is now almost the universal way to launch MPI programs.

There are several implementations of MPI are available from different sources. Most high-performance hardware vendors support MPI. This provides users with a portable programming model and means their programs can be executed on almost all of the existing platforms without the need to rewrite the program from scratch. There are two popular and free implementations of MPI, MPICH and LAM. Each of these is a complete version of MPI I.

Argonne National Laboratory and Mississippi State University developed the MPI reference implementation, MPICH [30]. It has been ported to most versions of UNIX. LAM [31] (Local Area Multicomputer) is an MPI programming environment and development system developed by Notre Dame University. LAM includes a visualization tool that allows a user to examine the state of the machine allocated to their job as well as provides a means of studying message flows between nodes.

PVM is a software system that allows users to set up a controlling workstation that spawns child processes onto other machines. What makes PVM unique as a parallel programming environment is that it allows for the creation of an encapsulated virtual environment for running parallel programs. The virtual machine provided by PVM allows parallel programs can be run on heterogeneous collections of computers.

With PVM, each user may construct their parallel environment controlled from a single host on which child processes can be launched onto other machines. PVM is implemented via a daemon that is installed on each machine. Any user who has enough resources to compile and install the PVM package on a number of machines can therefore, run it. The PVM library has functions to support and aid the integration of parallel processing tools into the PVM environment. There are tools produced by researcher and vendors for PVM environments.

The MPI infrastructure supports parallel tools and utilities by providing standard semantics for communications, contexts, and topologies. Among the tools available, XPVM [32] is the most popular visualization tool for PVM. XPVM supports both on-line monitoring of events in a virtual machine, and post mortem analysis. XPVM is a valuable tool for both performance and functional analysis of a PVM program. XMPI [33] is a widely used public domain tool adopted by several vendors. XMPI is a X/Motif based GUI for running, debugging and visualizing MPI programs. Extensive MPI information is extracted from running an application on demand, or from a cumulative log of communication.

Among parallel debuggers, Totalview [34] and P2D2 [35] are the two notable tools that

support debugging PVM and MPI programs. Many resource management tools also support PVM programs, such as LSF [36] from Platform Computing, NQE [37] from SGI, Loadleveler [38] from IBM, and Condor [39] from University of Wisconsin.

One of the most important features of cluster will be that users can run their jobs non-interactively on cluster. Therefore, it is necessary to have load-balancing and queueing system for cluster. There are a large number of public domain and commercial packages to sever this purpose, such as DQS [40] from Florida State University, CONDOR from University of Wisconsin, Legion [41] from University of Virginia, Codine [42] from Gridware, Clustor [43] from Active Tools, LSF from Platform Computing. A comprehensive review of cluster management software can be found at the University of Portsmouth [44].

The number of Linux-based software and tools for cluster is growing every day. There are too many to consider attempting to list them all. But there are several Web sites that provide further information about this area.

- The Cluster Computing Info Centre [45] has links to cluster-related computing information, including books, documents, slides, free and commercial software.
- The Parallel Tools Consortium [46] initiative supports a collaboration of researchers, developers, and users working towards making parallel tools responsive to user needs. This includes timing libraries, parallel execution and visualization tools.
- Beowulf [47] is a typical COTS-based Linux cluster designed by Center of Excellence in Space Data and Information Sciences (CESDIS), a division of the University Space Research Association (USRA) located at the Goddard Space Flight Center. The Beowulf Web site provides all kinds of information on Beowulf-like Linux clusters, including academic research and commercial Beowulf sites. The site also maintains a mailing list for various discussions on Beowulf clusters.
- The Beowulf Underground [48] is a site that tracks developments in Linux clustering, including announcement from commercial vendors and links to various software releases.
- The Extreme Linux [49] Web site is an offshoot of the Linux Expo that tracks the use of Linux in parallel computing. A number of interesting papers are kept there, along with links to leading work on Linux clusters.

## 4. Cluster Applications

Clusters have been employed as an execution platform for a range of application classes, ranging from supercomputing and mission-critical ones, through to e-commerce, and database-based ones.

Clusters are being used as execution environments for Grand Challenge Applications (GCAs) [57] such as weather modeling, automobile crash simulations, life sciences, computational fluid dynamics, nuclear simulations, image processing, electromagnetics, data mining, aerodynamics and astrophysics. These applications are generally considered intractable without the use of state-of-the-art parallel supercomputers. The scale of their resource requirements, such as processing time, memory, and communication needs distinguishes GCAs from other applications. For example, the execution of scientific applications used in predicting life-threatening situations such as earthquakes or hurricanes requires enormous computational power and storage resources. In the past, these applications would be run on vector or parallel supercomputers costing millions of dollars in order to calculate predictions well in advance of the actual events. Such applications can be migrated to run on commodity off-the-shelf-based clusters and deliver comparable performance at a much lower cost. In fact, in many situation expensive parallel supercomputers have been replaced by

low-cost commodity Linux clusters in order to reduce maintenance costs and increase overall computational resources.

Clusters are increasingly being used for running commercial applications. In a business environment, for example in a bank, many of its activities are automated. However, a problem will arise if the server that is handling customer transactions fails. The bank's activities could come to halt and customers would not be able to deposit or withdraw money from their account. Such situations can cause a great deal of inconvenience and result in loss of business and confidence in a bank. This is where clusters can be useful. A bank could continue to operate even after the failure of a server by automatically isolating failed components and migrating activities to alternative resources as a means of offering an uninterrupted service.

With the increasing popularity of the Web, computer system availability is becoming critical, especially for e-commerce applications. Clusters are used to host many new Internet service sites. For example, free email sites like Hotmail [58], and search sites like Hotbot [59] (that uses Inktomi [60] technologies) use clusters. Cluster-based systems can be used to execute many Internet applications:

- Web servers;
- Search engines;
- Email;
- Security;
- Proxy; and
- Database servers.

In the commercial arena these servers can be consolidated to create what is known as an enterprise server. The servers can be optimized, tuned, and managed for increased efficiency and responsiveness depending on the workload through various load-balancing techniques. A large number of low-end machines (PCs) can be clustered along with storage and applications for scalability, high availability, and performance. The leading companies building these systems are Compaq [61], Hewlett-Packard [62], IBM [63], Microsoft [64] and Sun [65].

The Linux Virtual Server [66] is a cluster of servers, connected by a fast network. It provides a viable platform for building scalable, cost-effective and a more reliable Internet service than a tightly coupled multi-processor system since failed components can be easily isolated and the system can continue to operate without any disruption. The Linux Virtual Server directs clients' network connection requests to the different servers according to a scheduling algorithm and makes the parallel services of the cluster appear as a single virtual service with a single IP address. Prototypes of the Linux Virtual Server have already been used to build many sites that cope with heavy loads, such as Linux portal [67] and UK National JANET Web Cache Service [68].

Client applications interact with the cluster as if it were a single server. The clients are not affected by the interaction with the cluster and do not need modification. The applications performance and scalability is achieved by adding one or more nodes to the cluster, by automatically detecting node or daemon failures and by reconfiguring the system appropriately to achieve high availability.

Clusters have proved themselves to be effective for a variety of data mining applications. The data mining process involves both compute and data intensive operations. Clusters provide two fundamental roles:

- Data-clusters that provide the storage and data management services for the data sets being mined.
- Compute-clusters that provide the computational resources required by the data filtering, preparation and mining tasks.

The Terabyte Challenge [69] is an open, distributed testbed for experimental work related to managing, mining and modelling large, massive and distributed data sets. The Terabyte Challenge is sponsored by the National Scalable Cluster Project (NSCP) [70], the National Center for Data Mining (NCDM) [71], and the Data Mining Group (DMG) [72]. The Terabyte Challenge's testbed is organized into workgroup clusters connected with a mixture of traditional and high performance networks. They define a meta-cluster as a 100-node work-group of clusters connected via TCP/IP, and a super-cluster as a cluster connected via a high performance network such as Myrinet. The main applications of the Terabyte Challenge include:

- A high energy physics data mining system called EventStore;
- A distributed health care data mining system called MedStore;
- A web documents data mining system called Alexa Crawler [73];
- Other applications such as distributed BLAST search, textural data mining and economic data mining.

An underlying technology of the NSCP is a distributed data mining system called Papyrus. Papyrus has a layered infrastructure for high performance, wide area data mining and predictive modelling. Papyrus is built over a data-warehousing layer, which can move data over both commodity and proprietary networks. Papyrus is specifically designed to support various cluster configurations, it is the first distributed data mining system to be designed with the flexibility of moving data, moving predictive, or moving the results of local computations.

## 5. Cluster Computing in Melbourne

At Monash University, a cluster system called the Monash Parallel Parametric Modeling Engine (PPME) [50] has been assembled. The aim of PPME is being used to develop the next generation of software tools and also provide resources for large parametric models. Parametric computing involves the concurrent execution of one program on multiple machines over a range of parametric settings. PPME comprises 32 dual processor PC nodes (Intel Pentium IIs) running the Linux operating system. The machine is divided between the two Monash campuses, Clayton and Caulfield. The two clusters are connected using an ATM networks, creating a small cluster of clusters. A number of research and student projects are underway. Example projects include, factory, OPNET, ad hoc network and fuzzy logic simulations, static channel assignment, the N queens problem, disk fracture prediction, as well as biological and financial modelling. These projects have used a tool called Clustor (a commercial version of DSTC/Monash Nimrod [51] project). PPME has also been coupled into the Globus [52] GUSTO Grid Computing test-bed and used for the development of a wide-area resource management and scheduling system called Nimrod/G.

Nimrod is a tool for parametric computing on clusters and it provides a *declarative* parametric modelling language for expressing an experiment. Application experts can create a *plan* for parametric session and then use the Nimrod runtime system to submit, run, and collect the results from multiple computers. Nimrod has been used to run applications ranging from bio-informatics and operations research, to the simulation of business processes. A reengineered version of Nimrod, called Clustor, has been commercialised by Active Tools [43]. However, research on Nimrod has continued. Currently research is addressing its use in the global computational Grid environments and the new tool/system is called Nimrod/G. It uses Globus services for dynamic

resource discovery and dispatching jobs over wide-area distributed systems called computational grids.

Nimrod/G allows scientists and engineers to model whole parametric experiments and transparently stage the data and program at remote sites, and run the program on each different machine and finally gather results from remote sites. The user can also set the deadline by which the results are needed and the scheduler tries to find the cheapest computational resources available in the global computing grid and use them so that the user deadline is met and cost of computation is kept to a minimum.

The current focus of Nimrod/G research is on the use of economic theories in Grid resource management and scheduling as part of a new framework called GRACE [53] (Grid Architecture for Computational Economy). The components that make up GRACE include a global scheduler, a bid-manager, a directory server, and a bid-server that works closely with the Grid middleware. The GRACE infrastructure also offers generic interfaces (APIs) that Grid tools and applications programmers may use to develop software supporting the computational economy.

PARMON [54] is a tool that allows the monitoring of system resource and their activities at three different levels: system, node and component. PARMON also supports the monitoring of multiple instances of the same component, for example, multiple processors in SMP node. The major components of PARMON are:

- The *parmon-server* – It provides system activities and utilization information to clients.
- The *parmon-client* – It interacts with parmon-server and users for data gathering and presents the same in GUI format.

The client is a Java application and the server is a multi-threaded server written in C using POSIX/Solaris threads. PARMON is used to monitor C-DAC PARAM 10000 supercomputer [55], a cluster of 48 Ultra-4 workstations powered by the Solaris operating system.

The scientists at the Swinburne Centre for Astrophysics and Supercomputing (and also in Monash) use clusters for conducting research in computationally demanding area of pulsar astrophysics. The Swinburne Supercluster [56] consists of a 64-node Compaq Alpha system. The theoretical peak performance the Supercluster is in excess of 60 Gflops/s. The Supercluster is used in several major projects involving large-scale processing. The primary purpose of the Supercluster is to provide the computational power necessary to process the large amounts of astrophysical data that is collected on a regular basis. The cluster is also equipped with commercial licenses of several popular graphical rendering packages, commonly used in television and movie animations. A monitoring system with a Java applet-based interface has been implemented on the Supercluster. The applet can display individual system statistics (CPU state, uptime, network information, users, jobs, load average, memory, disk space) for all nodes or a sub-group of nodes.

The Australian government is investing heavily to setup Advance Computing centres at national and state sites. At the national level APAC [77] (Australian Partnership for Advanced Computing) has been established with a vision to underpin significant achievements in Australian research, education and technology dissemination by sustaining an advanced computing capability ranked in the top 10 countries. A similar initiative called VPAC [78] (Victorian Partnership for Advanced Computing) has been established at the state level. This is a consortium of six Victorian universities: Ballarat, La Trobe, Melbourne, Monash, RMIT, Swinburne. The Victoria state government has funded this initiative. VPAC aims to become a leader in the use of clusters for

"production" HPC applications. It also aims to gain substantial expertise in engineering and astrophysics applications. Along with the APAC partners, VPAC will also be focusing on developing computational tools and techniques, interactive visualization, computational biology and bio-informatics, computational chemistry and biology, environmental modelling, as well as data mining tools and applications.

## 6. Conclusions

In this paper, we have discussed the incentive for using clusters as well as the technologies available for building clusters. The main importance is placed on using commodity-based hardware and software components to achieve high performance and scalability and at the same time keeping the price/performance low. We have discussed a number of Linux-based tools and utilities for building clusters. As an example, we detailed some of the research efforts on cluster computing being undertaken by the researchers at Melbourne University, Australia.

Clusters are being used to solve many scientific, engineering, and commercial problems. We have discussed a example of Web server and data mining applications and how these can benefit from the use of clusters. Currently many large international Web portals and e-commerce sites use clusters to process customer requests quickly and also maintain a high availability of 24X7 throughout the year. The capability of clusters to deliver high performance and availability within a single environment is empowering many new, existing and emerging applications and making clusters the platform of choice.

Due to the growing interest in cluster computing, the IEEE Task Force on Cluster Computing (TFCC) [74] was formed in early 1999. TFCC is acting as a focal point and guide to the current cluster computing community. The TFCC has been actively promoting the field of cluster computing research and education with the aid of a number of novel projects. The TFCC sponsors professional meetings, publishes newsletters and other documents, sets guidelines for educational programs, as well as helping to co-ordinate academic, funding agency, and industry activities. The TFCC organizes two major annual conferences [75][76] and holds a number of workshops that span the range of cluster activities.

A top clusters list (TopClusters), a joint-initiative of the TFCC and the Top500 teams, aims to maintain information related to the most powerful clusters around the world. This list will be updated twice in a year and the data collected by this initiative will used to help in tracking the cluster technology progress and at the same time maintaining a history of the deployment of cluster-based systems in the real world.

## References


[1] M. Baker, A. Apon, R. Buyya and H. Jin, “Cluster Computing and Applications”, *Encyclopedia of Computer Science and Technology, Vol.45*, (A. Kent and J. Williams, Editors), Marcel Dekker, Inc., August 2001.

[2] M. Baker, R. Buyya, K. Hawick, H. James, and H. Jin, *Cluster Computing R&D in Australia*, Asian Technology Information Program (ATIP) Report, ATIP00.015, April 2000.

[3] R. Buyya (editor). *High Performance Cluster Computing: Systems and Architectures*. Vol. 1, 1/e, Prentice Hall PTR, NJ, 1999.

[4] R. Buyya (editor). *High Performance Cluster Computing: Programming and Applications*. Vol. 2, 1/e, Prentice Hall PTR, NJ, 1999.

[5] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, and V. Sunderam,

*PVM: Parallel Virtual Machine: A Users' Guide and Tutorial for Networked Parallel Computing*. MIT Press, 1994.

[6] K. Hwang and Z. Xu, *Scalable Parallel Computing: Technology, architecture, Programming*. McGraw-Hill, 1998.

[7] P. Pacheco, *Parallel Programming With MPI*. Morgan Kaufmann Publishers, 1996.

[8] G. Pfister, *In Search of Clusters*. 2/e, Prentice Hall PTR, NJ, 1998.

[9] D. H. M. Spector, *Building Linux Clusters*. O'Reilly & Associates, 2000.

[10] T. Sterling, J. Salmon, D. Becker, and D. Savarrese. *How to Build a Beowulf*. The MIT Press, 1999.

[11] B. Wilkinson and M. Allen. Parallel Programming Techniques and Applications Using Networked Workstations and Parallel Computers. Prentice Hall, NJ, 1999.

[12] TOP500 supercomputer. http://www.top500.org/

[13] HPC 4500 Cluster, http://www.sun.com/servers/hpc/products/hpc4500.html

[14] CPlant Cluster, http://www.cs.sandia.gov/cplant/

[15] LINPACK Benchmark, http://www.netlib.org/linpack/

[16] Hitachi/Tsukuba CP-PACS supercomputer, http://www.rccp.tsukuba.ac.jp/cppacs/

[17] Top Clusters, http://www.topclusters.org/

[18] Myrinet, http://www.myri.com

[19] Active Messages, http://now.cs.berkeley.edu/AM/active_messages.html

[20] Linux Document Project, http://www.linuxdoc.org/

[21] RedHat Software, http://www.redhat.com/

[22] Linux Parallel Processing, http://yara.ecn.purdue.edu/~pplinux/

[23] GNU software, http://www.gnu.org/software/

[24] Absoft, http://www.absoft.com/

[25] Numerical Algorithms Group, http://www.nag.co.uk/

[26] Portland Group, http://www.pgroup.com/

[27] PVM, http://www.epm.ornl.gov/pvm/pvm_home.html

[28] Message Passing Interface, http://www-unix.mcs.anl.gov/mpi/index.html

[29] MPI Forum, http://www.mpi-forum.org/

[30] MPICH, http://www-unix.mcs.anl.gov/mpi/mpich/

[31] LAM/MPI, http://www.mpi.nd.edu/lam/

[32] XPVM, http://netlib2.cs.utk.edu/utk/icl/xpvm/xpvm.html

[33] XMPI, http://www.mpi.nd.edu/lam/software/xmpi/

[34] Totalview, http://www.etnus.com/products/totalview/index.html

[35] P2D, http://www.nas.nasa.gov/Groups/Tools/Projects/P2D2/

[36] LSF, http://www.platform.com/platform/platform.nsf/webpage/LSF?OpenDocument

[37] NQE, http://www.sgi.com/software/nqe/

[38] Loadleveler, http://www.rs6000.ibm.com/software/sp_products/loadlev.html

[39] Condor, http://www.cs.wisc.edu/condor/

[40] DQS, http://www.scri.fsu.edu/~pasko/dqs.html

[41] Legion, http://www.cs.virginia.edu/~legion/

[42] Codine, http://www.gridware.com/

[43] Active Tools Inc., Clustor. http://www.activetools.com/

[44] Cluster management software, http://www.dcs.port.ac.uk/~mab/Cluster/cms.html
[45] Cluster Computing Info Centre, http://www.buyya.com/cluster
[46] Parallel Tools Consortium, http://www.ptools.org/
[47] Beowulf Project, http://www.beowulf.org/
[48] Beowulf Underground, http://www.beowulf-underground.org/
[49] Extreme Linux, http://www.extremelinux.org/
[50] Parallel Parametric Modelling Engine, http://hathor.cs.monash.edu.au/
[51] Nimrod Project, http://www.csse.monash.edu.au/~davida/nimrod.html/
[52] Globus, http://www.globus.org/
[53] GRACE Project, http://www.csse.monash.edu.au/~rajkumar/ecogrid/
[54] PARMON, http://www.csse.monash.edu.au/~rajkumar/parmon/
[55] C-DAC PARAM 10000 supercomputer, http://www.cdac.org.in/html/param.htm
[56] Swinburne Centre for Astrophysics and Supercomputing, http://www.swin.edu.au/supercomputing/
[57] Grand Challenge Applications, http://www-fp.mcs.anl.gov/grand-challenges/
[58] Hotmail, http://www.hotmail.com/
[59] Hotbot, http://www.hotbot.com/
[60] Inktomi, http://www.inktomi.com/products/network/traffic/tech/clustered.html
[61] Compaq Server Consolidation, http://www.compaq.com/solutions/serverconsolidation/
[62] Hewlett-Packard System Consolidation, http://www.unixsolutions.hp.com/solutions/sc/
[63] IBM S390 Server Consolidation, http://www.s390.ibm.com/sc/
[64] Microsoft Application Center 2000, http://www.microsoft.com/applicationcenter/
[65] Sun Data Center Consolidation, http://www.sun.com/datacenter/consolidation/
[66] Linux Virtual Server, http://www.LinuxVirtualServer.org/
[67] Linux portal, http://www.linux.com
[68] UK National JANET Web Cache Service, http://wwwcache.ja.net/
[69] Terabyte Challenge, http://www.ncdm.uic.edu/TC2000.htm
[70] National Scalable Cluster Project, http://www.ncsp.uic.edu/
[71] National Center for Data Mining, http://www.ncdm.uic.edu
[72] Data Mining Group, http://www.dmg.org
[73] Alexa Crawler, http://www.alexa.com/support/technology.html
[74] IEEE Task Force on Cluster Computing, http://www.ieeetfcc.org/
[75] IEEE International Conference on Cluster Computing, http://www.clustercomp.org/
[76] IEEE International Symposium on Cluster Computing and Grid, http://www.ccgrid.org/
[77] APAC, http://www.apac.edu.au/
[78] VPAC, http://www.vicpac.org/