Clusters and Node Architecture

by

Jason Ng and Greg Rogers

School of Computer Science and Software Engineering
Monash University
Clayton Campus, Melbourne, Australia

Email: {jng, gjr}@csse.monash.edu.au

Abstract

Clustering is a term that describes the idea of linking together multiple low-cost, commercially available machines in order to achieve processing power of a magnitude impossible to otherwise obtain without the use of purpose-built supercomputers. A cluster of less individually capable machines can rival the performance of a supercomputer, while being more upgradable and cheaper.

This paper briefly examines the advantages and disadvantages of several different computer architectures as building blocks for cluster systems. Additionally, the configurations of a number of conspicuously successful clustered systems are examined in more detail.

Introduction

Research in cluster computing grew out of a sense of frustration with certain practical limitations on the utility of supercomputers. Some problems require sheer computing power – no tricky algorithms can substitute. Obvious examples are meteorological and weather prediction problems, large particle-number physical simulations such as internal reactor simulations and geophysical simulations, and fluid dynamics problems.

The processing power required to run these programs was clearly way beyond the capability of any desktop machine, so supercomputers seemed the only solution. But the use of supercomputers has drawbacks. Firstly, supercomputers are extremely expensive. The inherent high price of sophisticated computer equipment was only increased by the small production runs of a typical model. The market was unlikely to grow in the short to medium term, with only government agencies, certain engineering corporations, and universities actually needing the product, so the price was not going to come down. And, in a nice Catch-22, the market was unlikely to grow while the prices remained so high. A second problem was the instability in the industry. When a product is so complex to design and manufacture, and the absolute number of sales is so low, a single supercomputer model takes is very significant to the fortunes of its manufacturer. Not only is the time between one model and the next inconveniently long, but one poor selling model can seriously hurt a company. This means that, not only is after-sale support for a system going to be unreliable, but a client whose provider goes broke is going to have to change system architectures when they next upgrade. This means that staff are going to have to be trained on the new system, code ported over, and so on.

The clustering approach focuses on utilizing the idle resources of a network to aid processes that are running. A typical networked computer spends much of its time with the CPU idle while I/O is performed, or other tasks undertaken. If the idle CPU cycles can be put toward assisting a numerically intensive computation taking place on another network node, overall system performance could be speeded up considerably. Modern clustered systems frequently take this one step further, by incorporating two layers of machines. Firstly are the workstation nodes, terminals that can be used to log on and execute programs from. Then there are the dedicated nodes whose only role is to be remote assistants to the workstation machines in running parallel applications.

This paper investigates the various computer architecture types that can be used as nodes in cluster layout and construction. It contains a brief investigation into their relative advantages and disadvantages, and examines the available operating systems for each architecture, and the range of software available.

Works on this area

Name	Description	Remarks	URL
Grendel	18x150MHz Pentium	Uses Linux Red Hat and Beowulf, used for parallel I/O research	Http://ece.clemson.edu/parl/grendel.htm
Collage	18x PC	Linux Red Hat, not Beowulf, used for experimental rendering & visualization	http://www-fp.mcs.anl.gov/collage/
Appleseed	16 Mac G4	Uses MacOS, Mac version of Beowulf, performs numerically intensive experiments and simulations	http://exodus.physics.ucla.edu/appleseed/appleseed.html
Monash Parallel Parametric Modeling Engine	62 Dual Pentium II & III PC	Uses Turbo Linux, rely on software 'Clustor' for data intensive parametric modeling experiments and simulations	http://hathor.cs.monash.edu.au
NCSC Cluster	Hybrid of Cray, SGI Supercomputers, Sun, IBM and DEC server workstations	Runs on Unix Operating system, open for academic and government to carry out research and teaching. For both commercial and academic purposes.	http://www.ncsc.org
Cocoa	25x dual 400MHz Pentium	Red Hat Linux, used for aerospace problems	http://cocoa.aero.psu.edu/

PC Clusters

Probably the most commonly used node architecture in cluster construction is the Intel or Intel-clone PC

processor. There are various reasons for this. Firstly is the price factor. PCs are by far the market leader in the standalone and LAN terminal market. Due to the sheer production volume that this market leadership brings, PC machines are significantly cheaper on a one-to-one basis than Unix workstations, although workstations still have a slight edge in sheer speed. Since one of the main goals of clustering systems in the first place is to reduce the cost of high-performance computing systems, the lower unit price makes PCs very attractive. A further beneficial side-effect of the high-demand PC market is that the performance of PCs improves faster than that of Unix workstations, as vendors constantly try to offer a faster machine than that their competition is selling. A less obvious financial argument for the PC architecture is its modularity. A cluster administrator wants to keep the cluster running as fast as possible, which means that the nodes will be frequently upgraded. If a Unix workstation such as a Dec Alpha was used, upgrading would mean replacing the entire node. Upgrading a PC, on the other hand, merely involves replacing the motherboard, or slotting in more memory. Components such as the hard disks and power supplies are completely compatible with the new processor, and upgrading is a fairly fast and simple process.

The base configuration for PC machines intended for clustering would involve Pentium III processor, giving a 550MHz clock speed. 256 MB Ram and a 10GB hard disk would be minimums if the cluster was to be used for numerically intensive applications. A wide variety of networking systems and link speeds are supported by PCs, another result of their widespread use, so it is possible to customise the cluster to the desired performance specifications and/or cost limitations. However, the choice of operating systems will also partially dictate cluster setup.

There are many possible choices of operating system for a PC cluster. The market leader in standalone operating systems, Microsoft, offers Windows NT. Microsoft products, particularly operating systems, have an unfortunate, and not entirely undeserved, reputation for chronic unreliability, and Windows NT is no exception. Server availability for an NT system is considered good if it achieves 99.5% to 99.9% - translating to between 10 and 40 hours of unplanned downtime per year. Server availability for a Unix system, by comparison, is considered poor if it drops below 99.9%, and Unix systems can frequently achieve ‘five nines’ – 99.999% - availability, translating to only around 5 minutes of unplanned downtime per year. A further disadvantage of NT as a cluster operating system is its limitations on the number of processors. Microsoft is yet to demonstrate an NT cluster incorporating more than 128 processors over 8 servers. A single Unix server can deal with from 64 up to 256 processors, while a cluster can have processors numbering over ten thousand.[8] NT can therefore support far fewer users on a cluster than Unix, and is significantly less dependable, since extra processors can be used to ‘cover for’ those that have hit errors. To make matters worse, NT is notably less likely to recover gracefully from a memory referencing or similar error than a Unix/Linux system. The advantages of an NT cluster are speed (in certain circumstances) plus the support advantages of dealing with a market leader. Regarding speed, the performance of an NT cluster has been measured to match that of a Cray T3E, but only when 8 byte messages are used[4]. Software availability is another advantage. There is a huge variety of tools and application software for the NT OS, although application software destined to be run on high-performance computers is generally custom designed by the users rather than bought from a software vendor. The other advantage of NT is the system interface and philosophy that is generally familiar to those with experience of MS operating systems (at least at the user level).

Some variant of the Linux operating system is the most common choice for running a cluster of PC machines. The many variants of Linux, due to the open source concept, make it possible to choose a version that best suits a particular cluster setup. Linux is usually the operating system of choice for PC clusters because of its reliability and scalability. The problems that Windows NT has with reliability are documented above, and for many cluster applications the severity of these is simply unacceptable. Server applications, for instance, especially DB servers, are far too unreliable if they are unpredictably down for one hour out of every 200.[8] A DB server going down is not only a hindrance to those wishing to work on the system at the time, but frequently requires a restore from backup or similar tedious procedure to

restart. Linux also offers improved ability to expand the cluster as more nodes are required, although the Microsoft 'Millennium' project may improve the situation for users who prefer NT. Linux is well established in the scientific computing community, and many programming and system management tools are available. Linux as a whole is less well supplied with user applications such as spreadsheets and WYSIWYG word processors, but running this sort of application is not the prime function of a cluster anyway.

Although NT and Linux comprise the majority of operating systems installed on PC clusters, there are a number of less widely used options. For example, FreeBSD, a freeware Unix-based OS, is sometimes used because of the increased support that it supplies for PCI devices.[6]

Software tools are widely available for both NT and Linux clusters. Both have several usable implementations of the two most commonly used message passing interfaces, MPI and PVM. Other PC operating systems will generally have at least one of these available as a library, plus an associated compiler.

In general, PC clusters have a great deal of potential. The increased effort that the hardware side of the industry is devoting towards the PC architecture means that the performance gap between fast workstations and PCs is narrowing rapidly, and the potential financial advantages of using a PC cluster rather than a single supercomputer or a cluster of workstations are significant both at purchase time and in the long run. There is a wide choice of OS and application software available, and all standard programming languages are supported.

Macintosh Clusters

Clusters using the Macintosh architecture generally use either the G3 or G4 PowerPC processors. These machines approach or exceed the fastest PC machines in performance, so they are a logical choice to attempt to combine into a high-performance cluster. The current minimum specifications for a Macintosh intended for clustering would be a clock speed of around 400MHz, 256+MB of RAM, 1MB cache and 10GB hard disk. Since G4s come with a 100BaseT Ethernet adaptor preinstalled, this is the usual choice for network medium. Two machines can be clustered with no extra hardware. In order to cluster more than two Macintoshes, an Ethernet switch or hub is required. A hub can be used, but since a switch gives better performance when a cluster contains more than four Macs, hubs are not commonly employed. It is rare to want to construct a cluster with four or less machines if the cluster is going to be used for numerical applications, although a small cluster may be useful in other circumstances such as in a university environment where a platform is needed to give students experience in parallel programming.

There are two main operating systems that can be used on Macintosh clusters. One option is some variant of Linux, which is discussed in more detail in the section on PC clusters. Using this OS has several advantages. Firstly, it is available free, although in the context of constructing a cluster, relative savings are likely to be minimal. Secondly, Linux is an extremely common operating system in the High-Performance Computing community, and this gives new and experienced users a vast body of knowledge to draw on when constructing or troubleshooting a system. Finally, and on a related matter, the popularity of Linux, and its open source philosophy, has led to a very wide and generally robust variety of software for the platform, including various system tools and the GNU compilers.[2]

The other possible operating system for a Mac cluster is MacOS. MacOS is less widely used in the field of cluster technologies than Linux, and so has a less mature set of tools available, but it has several advantages that make it a genuine option. The initial setup of a Mac cluster is far easier than that of a Linux/Unix cluster, and substantial experience in dealing with Unix is necessary to maintain the system. The use of Macintosh systems is standard in many workplaces and university departments, and as little

change as possible is desirable during the transition to or addition of a cluster system. Requiring that all system users learn the intricacies of another operating system is intrusive and inefficient, especially when the transition is from the simple, GUI-based MacOS to command-line based Unix/Linux. After all, the decision to use MacOS in the workplace initially was probably based on the performance of desired applications, such as Mathematica, being superior under MacOS than under other operating systems.

One further advantage of the use of MacOS as a cluster operating system is that the operating system contains a native message-passing interface called Program-To-Program Communications, that is usable by applications.[1] The operation of PPTC is very similar to the lower-level communication methods of MPI, and it can be used very well as the basis of a full Mac MPI library. This MPI implementation is adequate for most applications, but is not the fastest available, since PPTC was written when networks were much slower. The AppleSeed Macintosh clustering project has created new MPI implementations for both C and Fortran based on Apples Open Transport protocol-independent communications library. These libraries have performance approximately seven times faster than the PPTC MPI for large message sizes.[1]

Clustering Macs is not yet a mature technique. The individually powerful PowerPC processors and the comparatively easy cluster construction process makes a Mac cluster a cheap and powerful option for intensive numerical and/or parallel applications such as simulations of particle systems. However, the tools available for the NT or Linux environments simply don't exist under MacOS, although MacOS X may improve the situation. Even the AppleSeed project admit that they have solved the problem of missing job management and scheduling tools by ignoring them and making sure that circumstances in which they would be necessary never happen. This is not a scalable solution. In addition, some of the MacOS's error recovery routines, while robust, are not suited for use on remote systems.

Workstation Clusters

Avalon Cluster

Avalon is a 140 processor Alpha Beowulf cluster constructed entirely from commodity personal computer technology. It is a co-operative venture of the Los Alamos National Laboratory Center for Nonlinear Studies and Theoretical Division. Each node on the cluster is a DEC Alpha workstation, with a 533MHz 21164A Alpha microprocessor with 256 MB of memory and 3 gigabyte disk space. The cluster runs with the Linux operating system.

In the current configuration of 140 nodes, Avalon ran the parallel Linpack benchmark at 47.7 Gflops, a 320 million particle molecular dynamics simulation (SPaSM) at 29.6Gflops, and a gravitational treecode at 17.6 Gflops. The treecode and Linpack run at about the same speed on a configuration of 70 Avalon nodes as on a 64 processor 195 MHz SGI Origin 2000. Comparing the cost of the Avalon cluster with the SGI Origin 2000 super computer, in May 1998 list price for a 64-processor Origin 2000 with 250 MHz processors and 8 Gbytes of memory is around 1.8 million dollars but the total cost of the Avalon cluster is just around USD$313000., with 140 533Mhz processors, 36Gbyte of memory and 420Gbtyes of disk space. We can buy 5 Avalon clusters for the price of a SGI Origin 2000 !!!

Avalon is being utilized as a general purpose supercomputer for applications, which include astrophysics, Nonlinear Dynamics and Partial Differential Equations, Stochastic Partial Differential Equations, Molecular Dynamics, Phase Transitions in the Early Universe , Monte Carlo Simulation of Spin Glasses and Instanton Liquid Simulations.

The Avalon cluster mainly uses Message Passing Interface (MPI) as the communication tool between the cluster nodes.

Success Stories

In this section we will have a look at a few successfully set up and running clusters, in different architectures.

The Monash Parallel Parametric Modeling Engine (PPME)

The Monash Parallel Parametric Modeling Engine (PPME)[9] is created by the School of Computer Science and Software Engineering in Monash University. This Parallel Parametric Modeling Engine is a 62-node cluster spread across the Clayton and Caulfield campus of Monash University. Each node is a PC workstation with dual Pentium processors and those 62 machines are partitioned into 2 sub-cluster. At Clayton campus, the cluster host is named 'hathor' which consist of the host and 14 machines, while 'mahar' the sub-cluster at Caulfield campus is formed by 16 machines. These machine are configured to run with Linux and Windows NT. With a total of 62 microprocessors and 5.8 gigabytes of memory, 180 gigabytes of disk space, this cluster can perform 62 times faster than any single stand alone machine.

This parallel parametric modeling engine primarily used perform various numerical simulations such as finite element analysis, computational fluid dynamics, electromagnetic and electronic simulation, pollution transport, granular flow and digital logic simulation. These experiments are carried out with the help of a software tool called 'Clustor', a commercial version of Professor David Abramson's Nimrod project[10] in Monash University. Clustor act as an agent to prepare parameters of simulation models, generate data, dispatch jobs and summon up results for the simulation program.
The Monash Parallel Parametric Modeling Engine (PPME) is a successful implementation of the 'Beowulf Project', initiated by NASA, originally created to be a substitution of the expensive Massively Parallel Processor (MPP) machines by researchers working there. These researchers have spent years fighting with MPP vendors, and system administrators over detailed performance information and struggling with underdeveloped tools and new programming models. Sharing a MPP or a super computer cannot meet the requirements of the high volume data computation since they can only access a small portion of the resources, but using a Beowulf cluster means with a fraction of the cost of super computers and MPP machines they can achieve the same or even better performance throughput. The Beowulf project also helped to introduce parallel computing environment to communities with limited resources, especially universities with limited funding, where most of the research are carried out.

Project Appleseed

On the other side of cluster development, University of California, Los Angeles (UCLA), produced a cluster made up of 16 Macintosh PowerPCs with the latest G4 processor, named 'Appleseed'[1]. According to Apple, the processor has the ability to execute at least one billion floating-point operations per second. It's a staggering measure of speed known as a "gigaflop." The Velocity Engine used in the chip uses the 128-bit vector processing technology used in scientific supercomputers.

According to the latest announcement from UCLA, the cluster now has 22 Apple Macintosh G3 and G4 computers running the MacOS, and making use of the Message Passing Interface (MPI) and Fortran to help them accomplish those intensive computations, 50-150 MFlops/node are possible where packet size is large.

A few remarkable research done by using the Appleseed cluster include simulation of the Ion Temperature Gradient Instability in a Tokamak, a fusion energy device. The turbulence in the nonlinear phase determines the confinement of plasma and heat in these devices. A typical calculation takes 90 hours on

four G3/266 Macintoshes. The simulation of Sixteen Dynamically Interacting Quantum Mechanical Particles in a Box. These particles exhibit behavior common in the microscopic world. One time step, containing over 40 million classical pushes and other calculations, takes five minutes on two G3/266 Macintoshes!!

According to UCLA - "A cluster of 4 G3s now has the same computational power (and twice the memory) as one of the best supercomputers of 8 years ago, a 4 processor CRAY Y-MP !!"[11]. Now we can see the power of cluster computing!

The North Carolina Supercomputing Center

The North Carolina Supercomputing Center (NCSC)[12] has one of the most impressive collection of supercomputer cluster, it is a hybrid of super computers and workstations.

The North Carolina Supercomputing Center was opened in 1989 for use by higher education, commercial, and government organizations. Over the decade, the scope of the Center has grown to include collaborations with state and federal organizations, and academic and commercial users from outside North Carolina.

NCSC has installed a heterogeneous cluster of workstations consisting of four DEC 3000 model 600S, two DEC 3000 model 800S, one DEC 4000 model 620 compute nodes and two IBM RS6000 model 590 compute nodes, an SGI ONYX visualization server, and a Sun Ultra 450 file server.

The DEC compute nodes are 175 MHz alpha workstations with at least 320 MB of memory and 4 GB of local disk space. The IBM compute nodes have 128 MB of memory and 3 GB of local disk space. The dual processor SGI ONYX, the only workstation in the cluster with a graphics head, provides visualization capability and is available for remote use as well as reserved console use at NCSC. The Sun Ultra 450 file server provides 60 GB of shared disk space to the cluster compute nodes also can be used as a compute node. It has 4 processors and 1 GB of memory.

The Cray Fellows program from NCSC provides computational resources and stipends to participating graduate students and faculty, which also include a CRAY T916 and CRAY T3E super computer. It provides a collection of software to assist in research and teaching in universities, which suit engineering, chemistry, mathematical and statistical analysis. Visualization and graphics Packages are also available. NCSC has also developed relationships with some third party software companies that allow North Carolina academic users to obtain the use of packages on their own campus for no charge by using a license from NCSC.

With the powerful clusters provided by NCSC, universities no longer have to worry about lack of computational resources to help the academics and students carry out research and assistance with their studies.

NCSC also has some environmental programs to carry out research like Air Quality Modeling, Numerical Weather Prediction, Decision Support Systems, Meteorological Modeling and so on, which can help improve our life and also future development of advanced techniques for research and development.

Summary and Conclusions

Clustered systems hold great promise in supplying supercomputer-magnitude processing power for far less financial cost than that required for a traditional supercomputer. The increased modularity offered,

especially by PC-based systems, can both further reduce costs, as well as extending the useful life of a cluster well beyond that expected for a monolithic system. Additionally the use of commercial components in clusters promotes OS and code standardization, and reducing the learning (and code translation) time when an application or a user transfers from one system to another.

Currently the most powerful individual node types used in cluster construction are various higher-end workstations. These will usually run Unix. However, PCs used as nodes are usually cheaper and more easily upgraded, so many clusters are based on this architecture. The majority of these are running Linux or some variant, as the major alternative, Windows NT, is too limited and unstable for truly large-scale clusters. Macintosh clusters are still in their infancy, and as yet lack required cluster administration software.

References

1. AppleSeed – A parallel Macintosh Cluster for numerically intensive computing: http://exodus.physics.ucla.edu/appleseed/appleseed.html
2. The Beowulf Project: www.beowulf.org
3. Clustering Intel architecture machines: http://www.ats.ucla.edu/at/clustering/
4. Comparing the Communication Performance and Scalability of a Linux and an NT Cluster of PCs, a SGI Origin 2000, an IBM SP and a Cray T3E-600. G. R. Luecke, B. Raffin and J. J. Coyle, Journal of Performance Evaluation and Modeling of Computer Systems
5. Cplant – Computational Plant: http://www.cs.sandia.gov/cplant/
6. DAISy – Distributed Array of Inexpensive Systems: http://rocs-PC.ca.sandia.gov/DAISy/DAISy.html
7. The Berkeley NOW Project: http://now.cs.berkeley.edu/
8. Clustering for NT: Smooth Scaling?: http://www.windowstechedge.com/wte/wte-1999-07/wte-07-clustering.html
9. The Avalon Cluster: http://cnls.lanl.gov/avalon/
10. Monash Parallel Parametric Modeling Engine, http://hathor.cs.monash.edu.au
11. The Nimrod Project, Professor David Abramson, School of Computer Science and software Engineering, Monash University, http://www.csse.monash.edu.au/~davida
12. Viktor K. Decyk, Dean E. Dauger, and Pieter R. Kokelaar, How to Build an AppleSeed: A Parallel Macintosh Cluster for Numerically Intensive Computing, Department of Physics and Astronomy, University of California, Los Angeles
13. The North Carolina Supercomputing Center: http://www.ncsc.org