

# USE OF LOW-COST BEOWULF CLUSTERS IN COMPUTER SCIENCE EDUCATION

Timothy J. McGuire<sup>\*</sup>

## ABSTRACT

A Beowulf cluster is a multiprocessor built from COTS (commodity off-the-shelf) components connected via a dedicated network, running open-source software. Beowulf clusters can provide performance rivaling that of a supercomputer for a miniscule fraction of the cost. This paper shares the author's experience in building two such clusters.

## INTRODUCTION

The Beowulf project was initiated in 1994 under the sponsorship of the NASA HPCC (High-Performance Computing and Communications) program to explore how computing could be made "cheaper, better, faster". They turned to gathering a collection of standard personal computers (a "Pile of PCs", or PoPCs) and inexpensive software to meet their goals. This approach is very similar to a *COW* (cluster of workstations) and shares the roots of a *NOW* (network of workstations,) but emphasizes:

- COTS (commodity off the shelf) components
- dedicated processors (rather than scavenging cycles from idle workstations)
- a private system area network (rather than an exposed local area network)

The Beowulf adds to the PoPC model by emphasizing:

- no custom components
- easy replication from multiple vendors
- scalable I/O
- a freely available software base
- using freely available distributed computing tools with minimal changes
- a collaborative design

Some of the advantages of the Beowulf approach are that no single vendor owns the rights to the product, so it is not vulnerable to single vendor decisions; it permits technology tracking, that is, using the best, most recent components at the best price; and, it allows "just in place" configuration so that it permits flexible and user driven decisions. It exploits readily available (usually free) software packages which are as sophisticated, robust, and efficient as commercial-grade software. These systems have been derived from community-wide collaboration in operating systems, languages, compilers, and parallel computing libraries.

Two of the operating systems which have been used are Linux (Slackware, RedHat, and Debian distributions have all been used) and FreeBSD. Both of these have commercial distributors and support full windowing capabilities, a variety of shells, a variety of quality compilers, and message passing libraries, such as PVM and MPI.

## **BEOWULF ARCHITECTURE**

Beowulf clusters have been assembled around every new generation of commodity CPUs since the first 100 MHz 486DX4 in 1994. The idea here is to use fast, but cheap CPUs, giving the highest possible price-performance ratio. They are typically interconnected with fast Ethernet cards and switches, so that network latency is minimized. Beowulf clusters have also been built from other chips, such as the Alpha RISC chip where performance is preferred over cost.

Small systems (typically with 24 or fewer nodes) have a simple topology – a single switch controlling the private network. If the cost of a switch is prohibitive, a hub may be used instead of a switch, but this will degrade the overall performance of the cluster. Most often, one node of the cluster has a second network card connecting it to the local area network. This would be the typical configuration used in an academic environment.

## **BUBBAWULF**

At Sam Houston State University, a Beowulf cluster, dubbed “Bubbawulf” has been constructed. Bubbawulf consists of 8 nodes. These are Pentium 350s, each with 64MB RAM. The main node has a 4GB disk with a keyboard and monitor. The other 7 nodes are diskless and headless. They are interconnected through a Cisco 2900 24-port switch, giving a 1Mb/s full-duplex switched Ethernet network. The slave nodes mount their file systems off the main node via NFS.

Bubbawulf was constructed early in 1999 at a total cost of about \$15,000. The cost to build an identical node today would be much less, falling in the range of a “no-cost” cluster discussed below.

## **CHEAPER CLUSTERS**

When the author was at West Texas A&M University in 1998, the “Buffalo CHIP (Cluster of Hierarchical Integrated Processors) was constructed at a total cost of about \$2,500. The most expensive part was a 16-port Fast Ethernet switch which cost \$1000. Node 0 was scavenged (an Intel Pentium 90 processor with 16MB RAM and a 1.0 GB hard drive.) Nodes 1-3 were purchased for less than \$500 each via mail order. They had 32MB RAM each with 3.2 GB drives. RedHat Linux 6.0 was installed, as was LAM-MPI, a standard message-passing library.

An even cheaper cluster, the Stone Soupercomputer, was built at Oak Ridge National Laboratories by a group of physicists with no budget from cast-off PCs and outdated network hardware (coax Ethernet.)

## **HOW TO BUILD A “FREE” BEOWULF**

Gather a collection of machines that are considered “too slow” to run Microsoft software. Typically, these might be older Pentiums in the 233 – 500 MHz range. The machines need not be identical. 32MB RAM would likely be considered a minimal amount for each node, and they

will likely have hard drives in the range of 4GB, which is more than adequate. It is possible to set up diskless stations, and they have some administrative advantages, but the initial setup is more complicated. If they have CD-ROM drives, that will simplify the hardware installation. At least one monitor and keyboard will be necessary.

Next, gather network hardware. Fast Ethernet switches are preferable, but not always available. Ethernet hubs are inexpensive, and possibly surplus if they are older 10Mb/s technology. 10Base2 Ethernet (coaxial cable) is slow, but does not require a hub or a switch.

After the hardware setup is complete, installation of the operating system comes next. Usually, Linux will be the operating system of choice. Set up each node as a stand-alone system, then let each node know about the others by assigning IP addresses (192.168.0.x is a good choice) in the /etc/hosts file. Finally, install communication software, such as MPI or PVM.

Message passing has been discussed in a companion paper [1].

## CONCLUSIONS

Cluster computing offers a very attractive, cost-effective method of achieving high-performance, and holds a promising future. A cluster can be built relative inexpensively, even at an institution with no equipment budget.

## ACKNOWLEDGEMENTS

The author would like to thank his students at Sam Houston State University for their support on this effort, and the anonymous reviewers for their comments and suggestions.

## REFERENCES

- [1] T. McGuire, "Message Passing as an Introduction to Distributed Processing in the Undergraduate Curriculum," *ACET Journal of Computer Education and Research*, vol. 1, no. 1, 2003.

---

\* Sam Houston State University, Department of Computer Science, Huntsville, TX 77341-2090, (936) 294-1571, email: [mcguire@shsu.edu](mailto:mcguire@shsu.edu)