BEOWULF CLUSTER FOR RESEARCH & EDUCATION

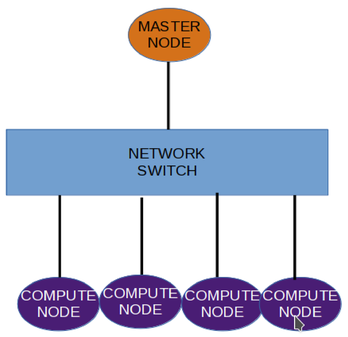
Stefano Coronado, Brandon Rozek & Ethan Carlos Ramirez
Advisor: Dr. Maia Magrakvelidze
University of Mary Washington Department of Physics

Abstract

As theoretical simulations become more complex, the amount of computational power needed to solve the models increase proportionally. Thus in order to keep up with the trend, we set out to design and construct an array of computers, called a Beowulf cluster, to serve the need by using a scheduling service to break apart large computational problems and distribute a piece of it to each computer in the cluster. As each computer solves its piece of the problem, it sends it back to the master where it recombined and presented to the user. This principle of parallelizing serves to cut down the total computational time, enabling the user to build more complicated and physically accurate models. Our cluster is designed with modularity and usability in mind so that additional computers can be added to the cluster with minimal user configuration and researchers from any discipline can run the system without extensive prerequisite knowledge. Proof of concept has been demonstrated and future work will involve refining and optimizing the system; **L**arge **U**niversal Networked Array of Computers (LUNA-C).

Introduction

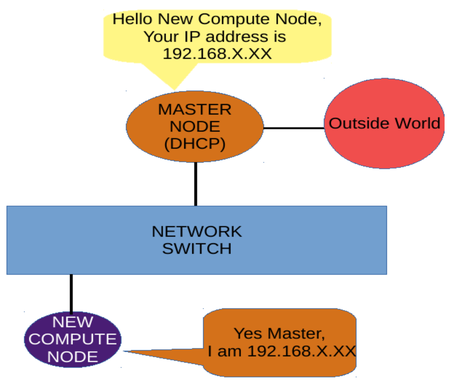
A Beowulf Cluster is an array of computers, known as nodes, that communicate with each other and a master node.[3] The master node will send instructions to the compute nodes over a Local Area Network (LAN) to complete a given task. The hardware used were surplused Dell OptiPlex 990 models generously donated by the University of Mary Washington ITSS.



The software utilized by the cluster is called Rocks Clusters, based on the CentOS 7.4 GNU/Linux distribution. Rocks Clusters is funded by the National Science Foundation [4] and was built specifically for high performance computing. The benefits of the distribution of the GNU/Linux operating system, CentOS is that it is used by many enterprises as an alternative to the proprietary Microsoft Windows operating system. CentOS itself is funded by RedHat, one of the world's leading Linux vendors. CentOS is commonly used by enterprise businesses that are unable to pay an expensive license fee to a software giant.

Networking

The compute nodes connect to the master node over a network switch. This is one network that maintained by the master node. The master node is configured to act as a Dynamic Host Configuration Protocol (DHCP) server, meaning if a new node were to be added, the master node would determine the IP address of the new node instead of the network switch. This allows the cluster to be scalable, meaning that new nodes will be able to add on to the total computing power.



The master node by design uses two networking interfaces, the interface described previously is used to communicate privately with the slave nodes. The other interface can be used to communicate with the outside world, allowing for people to connect to the cluster over the network and for an administrator to apply security updates and patches to the operating system without physically connecting to the machine via the network switch.

Hardware

The cluster is made up of surplused Dell OptiPlex 990 Desktop Towers. In addition to the version of the GNU/Linux operating system currently in commission, the master node is made up of the standard Dell-branded motherboard shipped from the factory with a mostly standard configuration.

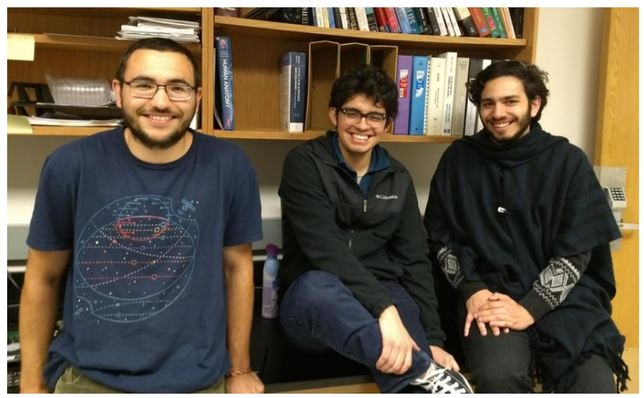
On each node, including the master, 8 GB of memory will be installed in addition to the Kingston HyperX 8GB DDR3 RAM module. The included processors are all Intel Core i5-2400 Sandybridge chipsets clocked at the standard 3.10 GHz across four CPU cores. We have replaced the standard hard drive for 16 GB Transcend solid-state disks (SSD) connected to the motherboard via the standard SATA connection. New nodes should prioritize booting from the network using the built-in Network Controller. Booting a new compute node from the network should display to a connected screen with the standard RedHat-based installer while the new node downloads the necessary software packages from the master node.

Software

Message Passing Interface (MPI) is a parallel computing framework that is used by the master node to send tasks and commands to the slave nodes. [6] MPI is implemented by Rocks in their own compilers shipped with their GNU/Linux configuration. A user can then implement MPI in C, C++, FORTRAN, Python and other major programing languages by using the appropriate import statement in the program that will be run through the cluster.

A scheduler can be used in conjunction with MPI. The scheduler is responsible for dividing resources between different tasks in a queue. Prioritization can be given to different users depending on their privilege level, meaning that faculty running tasks through the cluster can be given priority over a student task, should this be necessary.

A Network File System (NFS) share is synchronized between the master node and compute nodes allowing for files to be shared across all of the deployed nodes. The NFS share is useful for storing intermediate calculations or reading in input files.

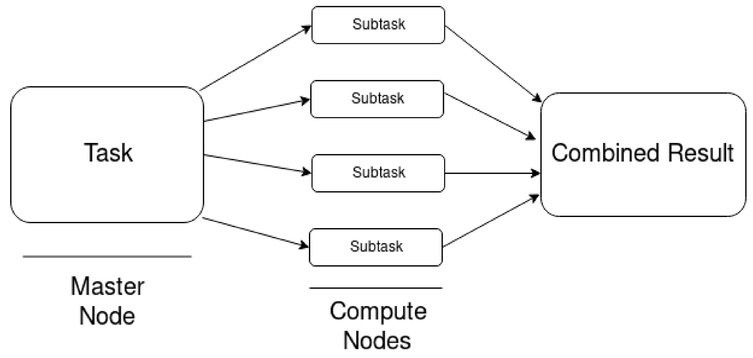


Tests Run

MPI comes bundled with an application called MPI Ring Test. This test ensures that the master node is able to communicate with the compute nodes and vice versa. Once the base functionality was tested, a pi approximator program was then executed.

For each compute node the pi approximator program does the following:
1. Randomly generate a x and y value between 0 and 1.
2. Check to see if x^2 + y^2 <= 1
3. If it is, increase the hit counter.
4. Increase the overall counter.

The master node aggregates all of the hit counters and overall counters and creates a proportion by dividing the overall counter by the hit counter. The approximate value of pi can then be found by multiplying this proportion by 4.



Applications

When fully implemented and connected to the University Network, students and researchers can take advantage of the computational power that would not be available on a consumer desktop or laptop. By logging into the master node using Secure Shell (SSH), users will be presented with a Unix shell to run their programs and tasks. The amount of resources has applications in machine learning, quantum chemistry, and brute-force algorithms.

Conclusion

Using the LUNA-C Cluster, students and researchers will be able to affordably run highly demanding computational tasks. Its modular design will allow for administrators to add more nodes to increase the overall performance. The software is backed by a major corporation and funded by the National Science Foundation, making it a recommended and long-lasting configuration to be used by future researchers in this area.

References & Commentary

[1] The Linux "Tux" Mascot is created by Larry Ewing. lewing@isc.tamu.edu
[2] The Rocks Cluster logo is not to indicate any affiliation with the project. Used for Non-commercial purposes
[3] "Frequently Asked Questions." *Beowulf.org*, www.beowulf.org/overview/faq.html.
[4] *NSF.GOV*, National Science Foundation, www.nsf.gov/awardsearch/showAward?AWD_ID=1032778.
[5] *NSF.GOV*, National Science Foundation, https://www.nsf.gov/awardsearch/showAward?AWD_ID=0721623.
[6] "The Message Passing Interface (MPI) Standard." *Message Passing Interface*, www.mcs.anl.gov/research/projects/mpi/.

Acknowledgements

We would like to thank Dr. Maia Magrakvelidze for taking up this project and giving us encouragement and support for a project outside her area of expertise. Thank you to UMW ITSS for donating 16 Dell OptiPlex 990 desktop computers that turned into our compute nodes. Finally, we would like to thank the University of Mary Washington for supporting our research in the form of grant money.