

## ARCHITECTING AND IMPLEMENTING GRID COMPUTING

ZAINAB NAYYAR & NAZISH RAFIQUE

Student, Department of Software Engineering, Army Public College of Management Sciences, Rawalpindi, Pakistan

### ABSTRACT

Few years back supercomputers were used for large computational work but they were used only for specific purposes, but after the completion of particular task one could not use them for any other general purpose and not more than one task could run on supercomputers at a time. After the invention of desktops it becomes easy to run more than one task but large computational work cannot be handled by them due to the limitation of resources. Due to these limitations, grid technology has emerged to a large scale to cater the jobs of supercomputers but without the resource limitation. By seeking this purpose a computational grid is deployed the job submission and resource utilization has been viewed. The basic purpose of this paper was to implement a grid environment and overcome the limitations of supercomputers and desktop computers. For Cost and time factors are considered to deploy a cost effective grid environment in less time period.

**KEYWORDS:** Grid Computing, Supercomputers, Clusters

### INTRODUCTION

Grid computing is a computer network where each computer is loosely coupled with each other and shares its resources to every other computer in the network. A system included in the grid computing environment share the load of a work across multiple computers to reach a common goal more efficiently and quickly. Computer resources consist of, Central Processing Unit (CPU), Memory, Storage [1].

### COMPARISON OF GRID COMPUTING WITH CLOUDS, CLUSTERS & SUPER COMPUTERS GRID COMPUTING

In grid computing systems, application is broken down into discrete modules, each module run on a separate server. Grid can be homogenous or heterogeneous i.e.: having different hardware configuration and operating systems in grid computing, computers are loosely coupled i.e.: hardware and software resources of computers interact with each other to perform a specific task but they do not depend on each other and can work independently. Grid computing is used in scientific community where several computers are connected to each other to solve a single technical problem that requires a large number of computer processing cycles or access to large amounts of data [2][3][4].

### CLUSTER COMPUTING

Cluster computing creates redundancy in applications because entire application is run on many servers. Clusters contain single type of processors and operating systems. Clusters contain static number of resources. In cluster computing, computers are tightly coupled [2].

### CLOUD COMPUTING

Cloud computing uses the computing resources which are available in a remote location and accessible over a network. It provides on demand resource provision. Cloud computing is provided as a 3rd party service that includes

software as a service, platform as a service or infrastructure as a service [2][3].

## SUPERCOMPUTERS

Supercomputers were used where large calculations were required. i.e.: weather maps, finding oil, earthquake prediction and sciences where lot of calculations must be done. The supercomputers had limited number of resources only a single project can be performed on them at a time. They were designed for specific purpose [4].

## Uses & Benefits of Grid Computing

Grid computing aims to involve everyone in the advantages of resource sharing and the benefits of increased efficiency. Grids give you shared access to extra computing power. A grid can also give you direct access to remote software, computers and data. A grid can even give you access and control of remote sensors, telescopes and other devices that do not belong to you. Grid development is used where secure access to shared resources is top priority to ensure secure access, grid developers and users need to manage three important things:

- Access policy - What is shared? Who is allowed to share? When can sharing occur?
- Authentication - How do you identify a user or resource?
- Authorization - How do you determine whether a certain operation is consistent with the rules?

It provides efficient and balanced use of computing resources. Grid provides access to computer resources from wherever you are. Interoperability between different grids is a big goal, and is driven forward by the adoption of open standards for grid development, making it possible for everyone can contribute constructively to grid development. Standardization also encourages industry to invest in developing commercial grid services and infrastructure [5].

## METHODS

Our system design composed of a frontend with which 3 compute nodes are connected via hub. When the user submit the job the frontend will distribute that job into small processes among all the compute nodes the compute nodes will then complete their task and submit it back to the frontend. Figure 1 shows the block diagram of the system, in which the complete infrastructure of the system is described.

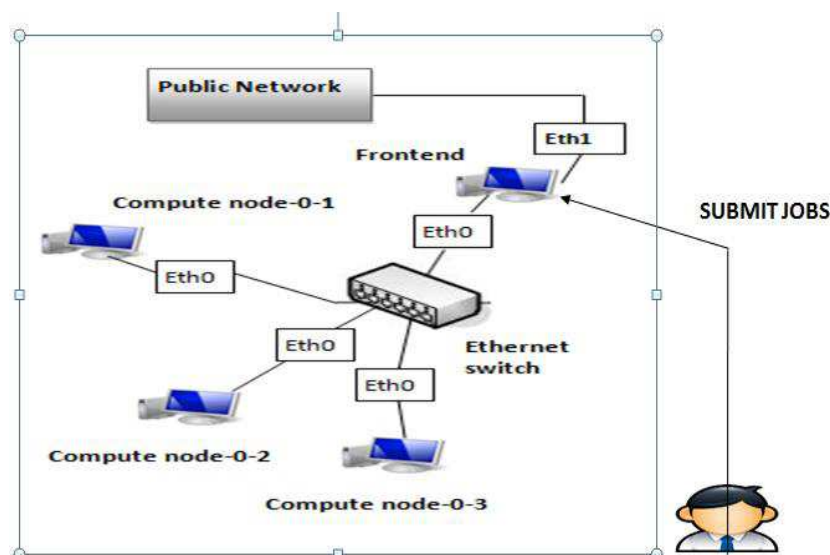


Figure 1: Block Diagram of Grid

When the user will submit the job for processing to the front end (server), it will divide the tasks into subtasks and distribute them among all the compute nodes. After the task completion every node will send back the output to the front end and the front end will compile the subtasks into a single task and return it to the user. Figure 2 shows the sequences operations performed by the system.

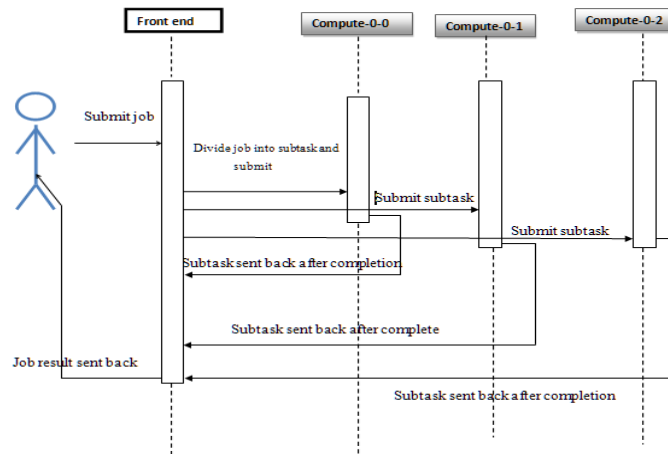


Figure 2: Sequence Diagram of Grid System

For implementing the system cent OS and rocks cluster 6.1 is used the user will directly interact with the rocks cluster at application layer. After installing the OS the compute nodes will be installed, so that they can share their resources for job submission. Now for job submission message passing interface and sun grid engine will be used. After submitting the job the process compilation phase will occur which will decompose the processes into sub processes so that they can be distributed to the processors for processing, every sub process will have a unique process id P-id through which it can be recognized. Then the sub processes will be distributed among the compute nodes for parallel processing. After task execution the processes will again sent back and compiled and then displayed to the user. The utilization of the resources can be viewed through ganglia monitoring tool. For further guidance visit ([www.rockscluster.org](http://www.rockscluster.org)). Figure 3 demonstrates the architecture diagram of the system.

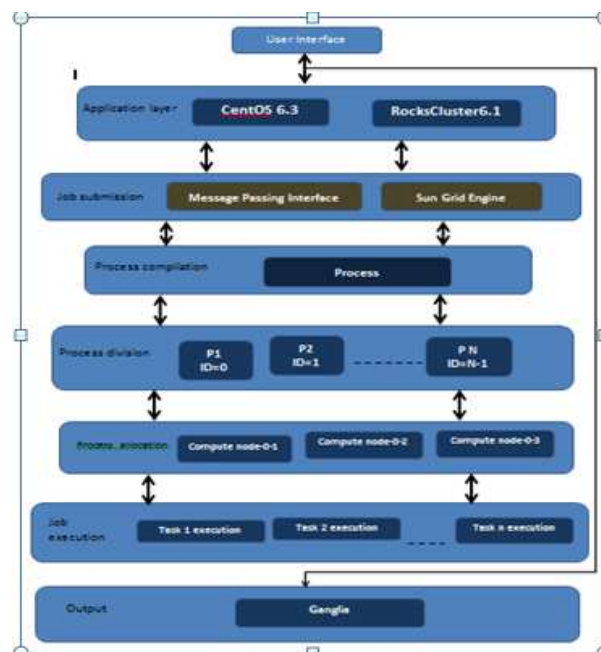


Figure 3: Architecture Diagram of Grid Computing

The smallest job first algorithm will be used in this process; the smallest jobs will be sorted from the list. The system will find the free processors and resources for task allocation. If the resources are not available than the system will keep the job in the waiting position unless or until the availability of resources. Figure 4 shows the basic flow of operations occurring in the system.

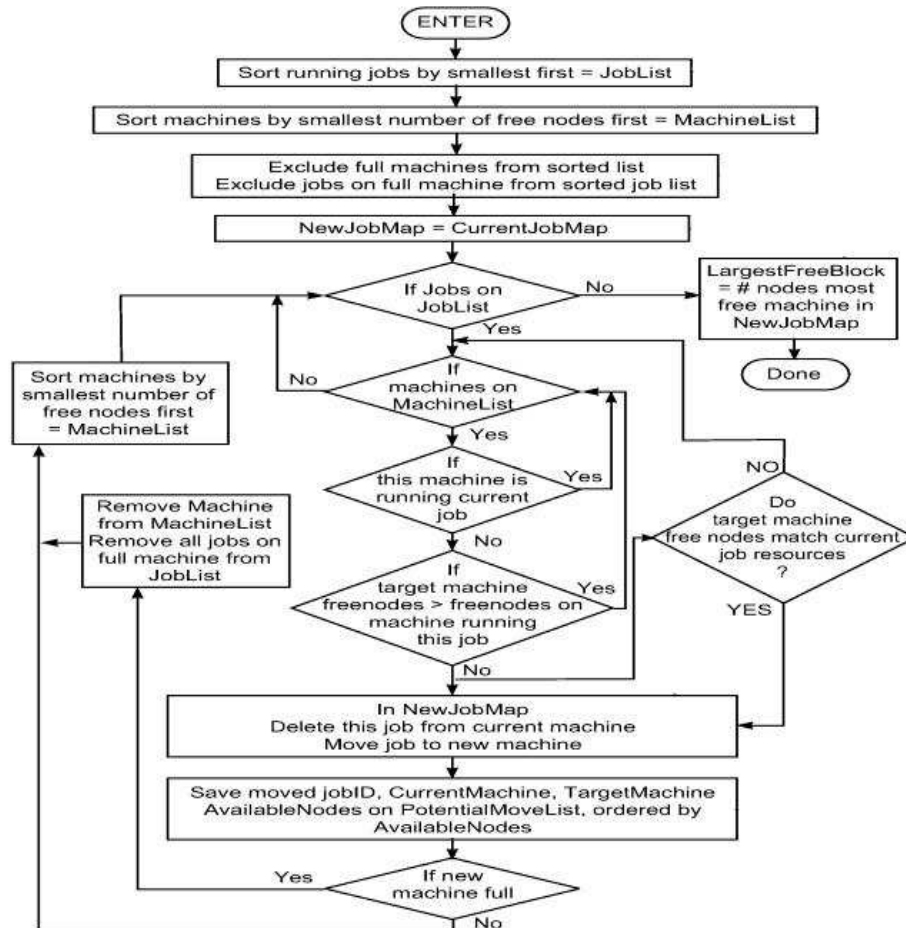


Figure 4: Flow Diagram of System

After job submission and compilation the job will enter into the running stage in it the resources are available it will execute otherwise it will send to the pending stage. Figure 5 shows the complete analysis of the states and transitions which occur in the system.

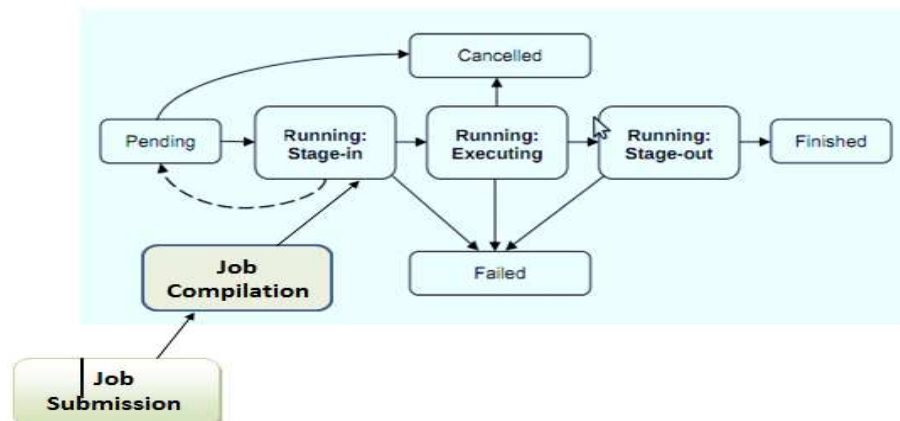


Figure 5: State Diagram of the System

## **APPLIED PROTOCOL**

The protocol applied on the system is tangosol cluster management protocol (TCMP): is a clustered IP based protocol for server discovery, cluster management, service provisioning and Data transmission. TCMP protocol is completely asynchronous, meaning that communication is never blocking, even when many threads on a server are communicating at the same time. TCMP uses a combination of UDP/IP multicast, UDP/IP unicast, TCP/IP.

## **JOB SHARING ON GRID**

In grid computing the job sharing is done in two ways through sun grid engine and through message passing interface.

## **SUN GRID ENGINE**

Sun grid engine is a grid resource and application manager it is helpful in sharing large applications among the grid environment. There is a queue setup in it due to which it keeps jobs in the queue, every job has a priority and on the bases of that priority the job is handled by the queue for execution on first come first serve basis.

## **USERS AND USER CATEGORIES**

Users of the grid engine system fall into following categories. Users in each category have access to their own set of grid engine system commands.

- Managers–Managers have full capabilities to manipulate the grid engine system. By default, the super users of all administration hosts have manager privileges.
- Operators–Operators can perform many of the same commands as managers, with the exception of making configuration changes, for example, adding, deleting, or modifying queues.
- Owners–Queue owners can suspend or enable the queues that they own. Queue owners can also suspend or enable the jobs within the queues they own. Queue owners have no other management permissions.

## **MESSAGE PASSING INTERFACE**

It is the interface that supports parallel computing it supports C and C++ languages the jobs written in these languages can be shared among the processes. It has libraries and functions which on the execution of jobs share the job among various resources and then recompile them for monitoring the output.

## **RESOURCE MONITORING**

The process of evaluating system to determine whether the development of the system satisfies the conditions imposed at the start of that phase. To confirm that the system is being built according to the user requirements and design specifications. In other words, to ensure that system development meets their specified requirements. During the development of this project each step is verified. All the actual outputs are compared with the expected outputs.

- Installation of frontend is verified by issuing the commands for installed packages.
- Installation of compute nodes is verified with the expected outputs shown in user manuals.
- Job sharing in message passing interface is viewed and found that they are executing as per expectations.

- Job sharing through sun grid engine is monitored through its graphical user interface. Job submission, job suspension, job pending, queue instances etc. Can be monitored and verified that jobs are running in respond to our inputs.

To ensure that the system actually meets the user’s needs, and that the specifications were correct in the first place. In other words, to validate that the system fulfills its intended use when placed in its intended environment. Our system can be validated using ganglia roll in Rocks Cluster 6.1. Ganglia roll monitors the health of our grid environment. It monitors CPU load, free memory, disk usage, network I/O, operating system version etc. The figure below shows total amount of CPU’s available in grid environment, hosts that are up while executing any task and the utilization graph of resources.

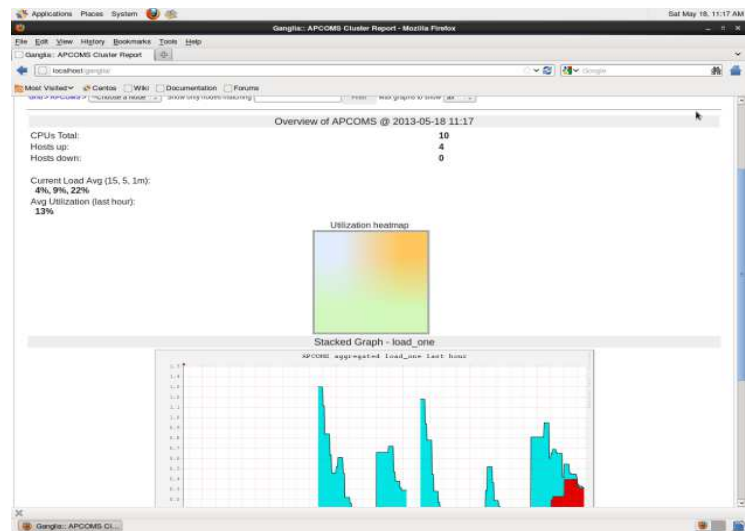


Figure 6: Total Amount of Cpu’s Available in Grid Environment

Figure below shows that four hosts are up, and the utilization map’s color has changed with the execution of application.

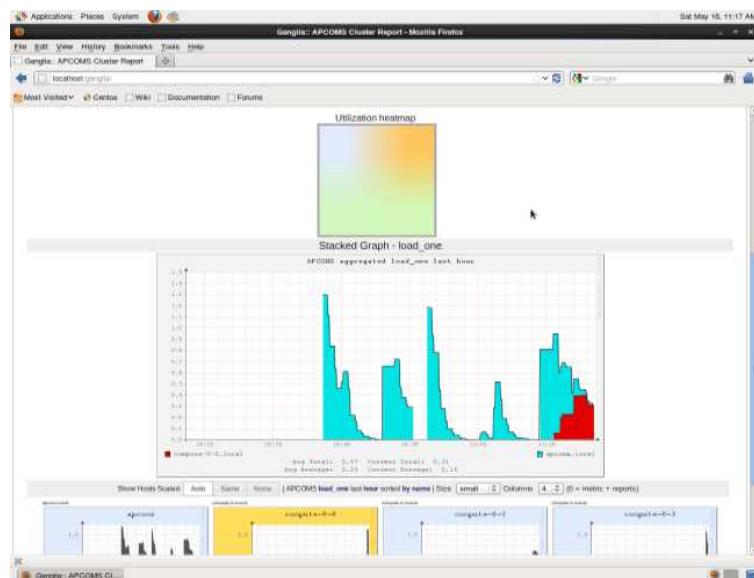


Figure 7: Utilization Map

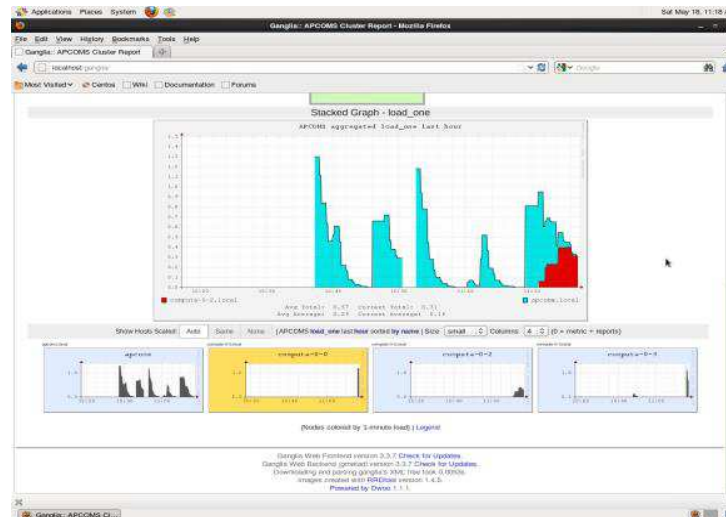


Figure 8: Showing Number of Hosts up

## CONCLUSIONS

The basic aim of deploying a grid environment is to replace the orthodox concepts of computations which were done through supercomputers. Grid has made the computations much easy. Our research on the working of grid environment is successfully completed. Now we can run the jobs according to our needs and can also monitor resource utilization in grid environment.

## REFERENCES

1. Poonam Dabas and Anoop Arya\*. Grid Computing: An Introduction. Volume 3, Issue 3 March 2013, pages 446-500. International Journal of Advanced Research in Computer Science and Software Engineering.
2. Brijender Kahanwal and Tejinder Pal Singh. The Distributed Computing Paradigms: P2P, Grid, Cluster, Cloud, and Jungle. Volume 1, Issue 2, Page No.183-187 July-August 2012. International Journal of Latest Research in Science and Technology.
3. Seyyed Mohsen Hashemi and Amid Khatibi Bardsiri. Cloud Computing Vs. Grid Computing. Vol. 2, No.5, May 2012, pages 188-194. ARPN Journal of Systems and Software.
4. Ioan Raicu, Ian T. Foster, and Yong Zhao. Many Task Computing for Grids and Supercomputers. 2008. IEEE.
5. Georgiana Marin. Grid Computing Technology. Volume 2, March 2011, pages 13-22. Database Systems Journal.
6. Rocks cluster 6.1 tutorials.
7. Sun grid engine tutorials.
8. MPI tutorials.



# Best Journals

Knowledge to Wisdom

Submit your manuscript at [editor.bestjournals@gmail.com](mailto:editor.bestjournals@gmail.com)

Online Submission at [http://www.bestjournals.in/submit\\_paper.php](http://www.bestjournals.in/submit_paper.php)