iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 **Issue:** IX **Month of publication:** September 2014

DOI:

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Grid Computing: An Introduction

Mansi Kakkar[1], Mandisha Sharma[2], Manish Sharma[3]

[1,2,3]*Department of Computer Science Engineering, Dronacharya College of Engineering, Gurgaon*

Abstract: Grid computing is a computing framework to meet growing demands for running heterogeneous grid enables application or the traditional "supercomputer" model for research and industry enterprises requiring increasingly large amounts of processing power and storage. A grid system is composed of computers which are separately located and connected with each other through a network. Grids are systems that involve resource sharing. This article introduces grid computing, lays out taxonomy for comparing and analyzing the existing grid resource management systems, discusses the optimization and coordination of computational grids, and presents quantitative methods for managing grid resources. Local scheduling, external scheduling, and coordination of grid endpoints are highlighted as methods of resource management. Techniques and ongoing research projects are also discussed for improving the economy of resource bundles in flat, hierarchal, and cell structure grids.

Keywords: Grid Computing, super computers.

I. INTRODUCTION

The term `Grid Computing' is relatively new and means a lot of different things to a lot of different people [1]. It has been used as a buzzword for any new technology to do with computing, especially computer networking, and therefore it has been over-hyped as the solution to just about every computing problem. One of the goals of this paper is to give a clear definition of what Grid Computing is and why it is required.

1.1 What is Grid Computing?

Grid computing is the collection of computer resources from multiple locations to reach a common goal. The grid can be thought of as a distributed system with non-interactive workloads that involve a large number of files. Grid computing is distinguished from conventional high performance computing systems such as cluster computing in that grids tend to be more heterogeneous and geographically dispersed (thus not physically coupled)[2]. Although a single grid can be dedicated to a particular application, commonly a grid is used for a variety of purposes. Grids are often constructed with general-purpose grid middleware software libraries. Grid size varies a considerable amount. Grids are a form of distributed computing whereby a "super virtual computer" is composed of many networked loosely coupled computers acting together to perform large tasks. For certain applications, "distributed" or "grid" computing, can be seen as a special type of parallel computing that relies on complete computers (with onboard CPUs, storage, power supplies, network interfaces, etc.) connected to a network (private, public or the Internet) by a conventional network interface, such as Ethernet. This is in contrast to the traditional notion of a supercomputer, which has many processors connected by a local high-speed computer bus.

Fig 1: Grid Computing

1.2 Types of grid:

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

On the basis of use grid computing can be divided into different types [3]:

- Computational grids: These type of grid are meant to provide secure access to computational resources, sufficient enough to perform processing of computational problems which otherwise would have required high computing power machines.
- Collaboration grid: With the advances in network hardware resources and internet services, demand for better collaboration has increased. Such desired collaboration is best possible with these kinds of grids.
- Utility Grid: In this type of grid not only CPU cycles are shared, also other software's and special peripherals like sensors are also shared.
- Network grid: Even if we have computational machines with enough computational power as a part of grid but with poor network communication one can't utilize those machines optimally. Network grid provides high performance communication using data caching between nodes there by speed-up communication with each cache nodes acting as router.
- Data grid: There are two things, data and computation over that data. Data grid provides the support for data storage other data related services like data discovery, handling, publication, etc.

II. WHY DO WE NEED GRID COMPUTING?

Although the amount of computing power available to both researchers and businesses is growing at an amazing rate, and has been growing quickly for some time, the demand for computing power is never satisfied. New projects in business and in the sciences require unprecedented amounts of computing power that, even given Moore's Law, will not be fulfilled in the near future. The rate of increase in network bandwidth is increasing at a rate faster than that of processor speed which means that the way to make best use of computing power is to network many computers together in an efficient fashion [4]. Grid Computing is currently seen as the best way to do this.
The New York Times recently published an article which argued that All Science Is Computer Science". This claim was made because every traditional science - Physics, Chemistry, Mathematics, Biology, Astronomy, and many others - is relying more and more on computers and computational power. Although new insights are still needed to generate new research in these fields, the limiting factor in many of the experiments is computational power. Grid Computing is therefore seen as the computing technology enabling the advancement of all sciences.

III. COMPARISON OF GRIDS AND CONVENTIONAL SUPERCOMPUTERS

"Distributed" or "grid" computing in general is a special type of parallel computing that relies on complete computers (with onboard CPUs, storage, power supplies, network interfaces, etc.) connected to a network (private, public or the Internet) by a conventional network interface producing commodity hardware, compared to the lower efficiency of designing and constructing a small number of custom supercomputers. The primary performance disadvantage is that the various processors and local storage areas do not have high-speed connections. This arrangement is thus well-suited to applications in which multiple parallel computations can take place independently, without the need to communicate intermediate results between processors. [5] The high-end stability of geographically dispersed grids is generally favorable, due to the low need for connectivity between nodes relative to the capacity of the public Internet.

There are also some differences in programming and deployment. It can be costly and difficult to write programs that can run in the environment of a supercomputer, which may have a custom operating system, or require the program to address concurrency issues. If a problem can be adequately parallelized, a "thin" layer of "grid" infrastructure can allow conventional, standalone programs, given a different part of the same problem, to run on multiple machines. This makes it possible to write and debug on a single conventional machine, and eliminates complications due to multiple instances of the same program running in the same shared memory and storage space at the same time.

IV. GRID ARCHITECHTURE

In grid computing infrastructure resources belong to and come from physically scattered administrative domains to collectively provide various resources to the users. In a grid, computing

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

nodes might not be placed at common physical location but can be independently operated from different locations. Each computer on the grid is a distinct computer [6]. Collection of servers clustered together to work out a common problem forms a grid [7]. The computers joined to form a grid may even have different hardware and operating systems. Grid consists of a layered architecture model providing protocols and service at five different layers represented by Fig.2.

- Fabric layer: It provides shareable resources such as network bandwidth, CPU time, memories, scientific instruments like sensors, telescope, etc. Data received by sensors at this layer can be transmitted directly to other computational nodes or can be stored in the database over grid. Standard grid protocols are responsible for resource control. Accomplishment of sophisticated sharing operation is the measure for quality of this layer. Operating system, queuing systems and processing kernels also form the part of this layer.
- Connectivity layer: This layer specifies the protocols for secure and easy access. Protocols related to communication and authentication required for transactions are placed in this layer. These communication protocols permit the exchange of data between resource layer and fabric layer. Authentication protocols are meant to provide secure cryptographic mechanisms for identification of users and resources. E.g. GSI –Grid Security Infrastructure (built around existing TLS protocols).
- Resource layer: This layer specifies the protocols for operating with shared resources. Resource layer build on the connectivity layer's communication and authentication protocols to define Application Program Interfaces (API) and software development kit (SDK) for secure negotiation, accounting, initiation, control, monitoring and payment of sharing resources. E.g. GRIP (Grid resource Information Protocol; based on LDAP), GRAM [8] (Grid Resource Access and Management) for allocation and monitoring of resources.
- Collective layer: Any collaborative operations in the shared resources are placed in this layer and it coordinates sharing of resources like directory services, co-allocation, scheduling, brokering services, and data replication services.
- Application layer: It is the topmost layer of the grid layered architecture. This layer consists of application which the user will implement and provides interface to the users and administrators to interact with the grid.

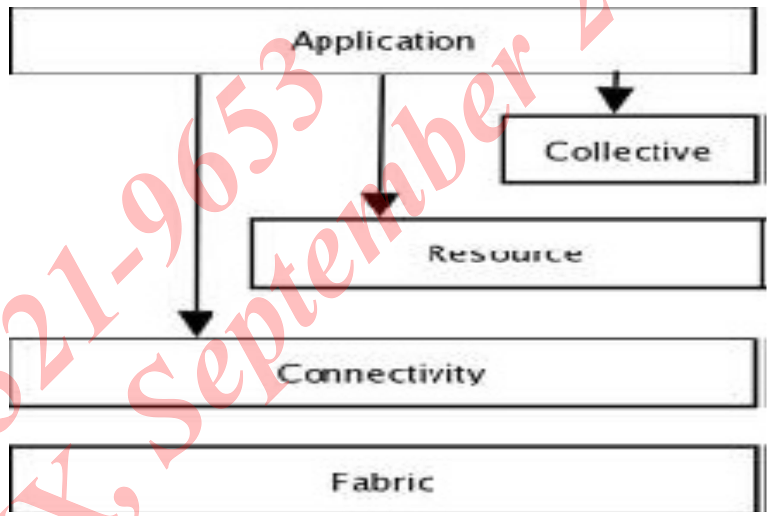


Fig 2: the layered Grid Architecture

V. CURRENT ISSUES IN GRID COMPUTING

Grid Computing is still very much in its development stage and there are a number of issues that must be addressed or resolved before it can be considered as a stable technology. Some of these issues are discussed below.

1.3 The Grid versus Many Grids:

A distinction must be made between the idea of a single, worldwide, ubiquitous grid and the idea of many separate grids located in businesses and on university campuses. The original intention of Grid Computing was that it would follow the same architecture as the electricity grid. This means that whenever and wherever you needed compute power you would simply \plug in" to *The* Grid and the processing would be done. There would be no need to know where the computing was being done - just as there is no need for me to know where the power that is lighting this room is coming from - only that it was being done.

Vol. 2 Issue IX, September 2014

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

In the same way that I don't need to know whether the electricity lighting this room is coming from a hydro-electric power plant in Fordland or a wind turbine in Wellington, I wouldn't care if my complicated simulation were being run on a spare machine next door or on an idle server somewhere on the other side of the world. In fact, *The* Grid could be viewed as a Grid of Grids; in much the same way as the Internet is a network of networks. Although work is still being done toward creating a single Grid, it is already the case that there are many disparate grids worldwide that are all completely isolated from each other.

1.4 No-one wants to share:

One of the biggest problems facing Grid Computing is not a technological one but a social one. Even when the technology exists for Grid Computing to work easily and flawlessly, people are still required to donate their spare CPU cycles or Grid Computing will not work at all. Although one of the major points of Grid Computing is that only spare cycles will be used, it still goes against human nature to allow others to access their computers and run programs on them. A fear of viruses is no doubt a valid threat as what has been viewed as a secure system in the past has been shown not to be so, so much work must go into developing a security infrastructure that can be completely trusted.

1.5 Grid Economics:

Before all the separate grids can be connected into one `super grid' some sort of billing system must be established that is accepted and trusted by everyone. It is unlikely for a worldwide Grid to take off and make use of almost all spare CPU time without some incentive for people to make their computers available. However, in order for a world-wide billing system to work, there will need to be some way of accurately keeping track of the CPU time used, the CPU time provided by each user and a way of transferring payment between users. The development of such a system in a way that is scalable and trusted by everyone is necessary before a global Grid can become the reality. The development of such as system could lead to some sort of global bidding system for compute power - which would fluctuate like the stock market. The value of CPU time would vary over time according to supply and demand. Daytime hours in the North America during the working week would probably have the highest demand so would cost more, but could make use of the servers in Europe and Asia that are not handling their peak capacity. The analogy of the Computer Grid with the electricity grid can be expanded further - just like it is possible to feed power back into the electricity grid - it will be possible to feed computing power back into the Computer Grid.

1.6 Performance Forecasting

One of the problems with scheduling resources on a Grid is that it is hard to know how long a resource will be available for or how good its performance will be if it is used. Researchers have implemented a tool known as *Every Ware* which contains, amongst other things, a performance forecasting mechanism. With accurate forecasting, scheduling becomes simpler because it is known that a given resource will react fast to requests or process data quickly. Without accurate performance forecasting a scheduler could schedule a remote set of CPUs to try and speed up processing but actually make it slower because those CPUs do not perform as well as expected.
There is still work to be done in this area, however, as the performance forecasting needs to be incorporated into scheduler algorithms and the accuracy of performance forecasting can no doubt be improved.

1.7 Security:

One of the reasons that people may not want to make their computer available on a Grid is that they do not trust other users to run code on their machines. Within small scale Grids this is not too much of a problem as Virtual Organizations at least partially eliminate the fear of malicious attacks. This is because in a Virtual Organization you can authorize only those from within a certain trusted organization to be able to access your computer. However, there could potentially be problems with the authorization systems and it is possible that someone from within the organization could act in a malicious way. With larger scale Grids it will be impossible to know and trust everyone who can access a single computer so the Grid infrastructure will have to provide guarantees of security in some way. The Java Sandbox Security Model [10] already provides an environment in which un-trusted users are restricted from making certain system calls which are not considered safe, and from accessing memory addresses outside of a certain range. Any Grid system will have to provide a similar

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

mechanism, so that users will be happy to let others access their computer.

VI CONCLUSION AND FUTURE SCOPE

Despite substantial research in grid computing certain issues are still holding back grid computing from becoming widespread in its use. Today Grid computing has been utilized by most of the scientific domains like astronomy, biological science, climatology, and much more. But the prime requisite for using grid is high speed internet, if one does not have a high speed internet one cannot get the best benefits from grid. This has been holding back grid for long. On one end of grid is high computation and optimized utilization of resources and at other the ability to manage distributed and heterogeneous systems. We need security with high availability of data and resources on demand and at the same time ease of access to implement these. On-demand provisioning of resources and more secure protocols for users and service providers to monitor and arrange payment is still a necessary requirement for commercial utilization of grid. No doubt grid computing is still evolving and better protocols and standards will be implemented in near future, culled from academia and business houses to speed up the grid evolution process. It will be of interest how various researchers would find new ways to improve grid and move along parallel paths. There is a lot of scope for Grid Computing as more and more complex algorithms are being developed. It is being speculated that grid computing through dedicated fiber optic links, might change the current scenario of broadband and other services providing bandwidth ranging from gigabytes to terabytes.

REFERENCES

[1] Jennifer M. Schopf. Grids: The top ten questions. In International Symposium on Grid Computing, March 2003.

[2]What is grid computing? - Gridcafe. E-sciencecity.org. Retrieved 2013-09-18.

[3] Ovaise Khan, "Types of Grid", [Online] Available: http:// thegridweblog.blogspot.in/2005/10/types-of-grids.html.

[4] D. Laforenza. Grid programming: some indications where we are headed. ACM Parallel Computing Special Issue: Advanced environments for parallel and distributed computing, pages 1733{1752, 2002.

[5] Computational problems - Gridcafe. E-sciencecity.org. Retrieved 2013-09-18.

[6] http://cloudcomputingapps.org/what-is-the-difference-betweencluster-computing-and-grid-computing.html

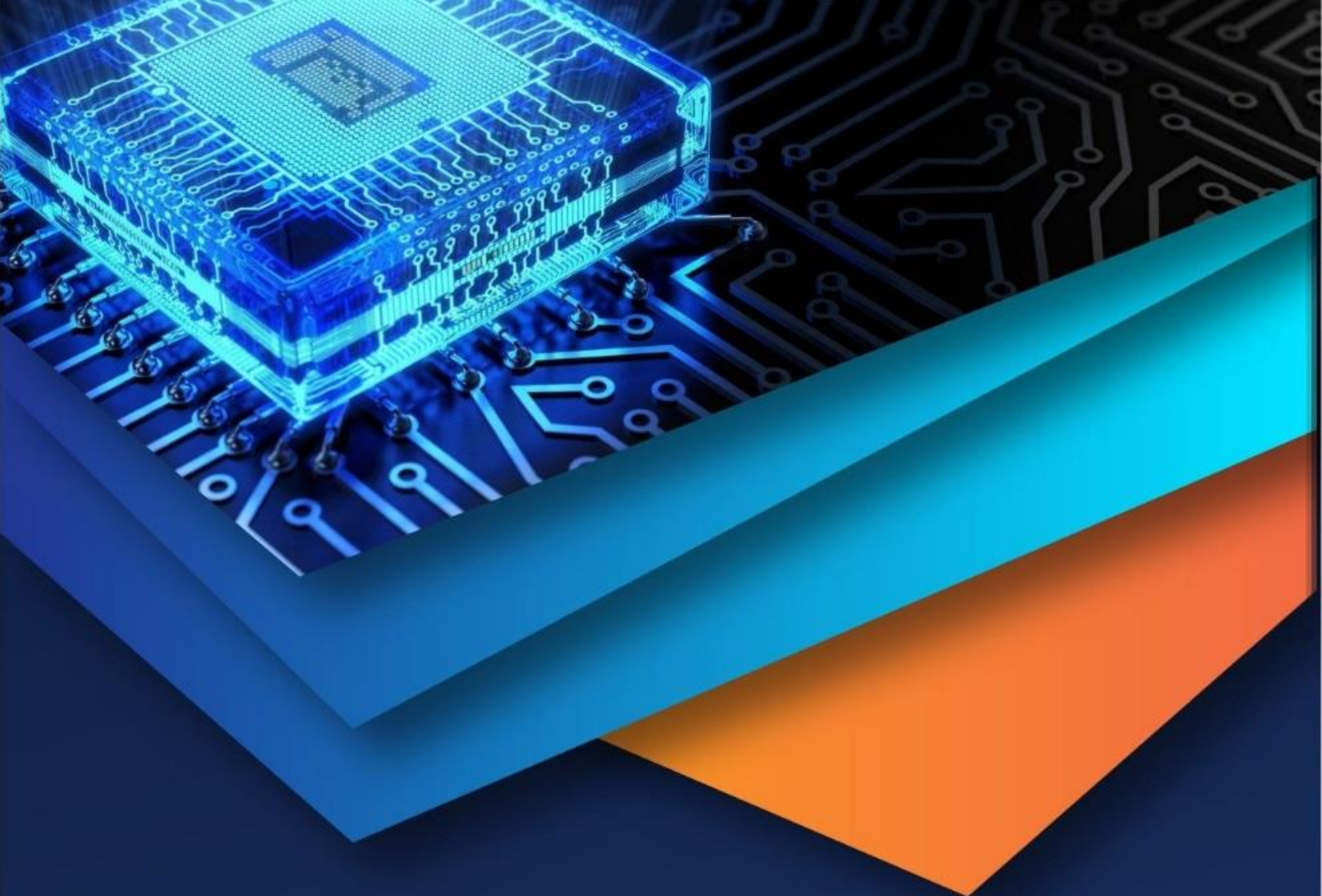
[7]http://www.linkedin.com/answers/technology/enterprisesoftware/TCH_ENT/552522-14941035

[8] Andre Rigland (2010-12-17). "Cloud Computing-How it is Different from Grid Computing". Trial lecturer in Conjunction with PhD thesis defense.

[9]Rich Wolski, John Brevik, Chandra Krintz, Graziano Obertelli, Neil Spring, and Alan Su. Running EveryWare on the computational grid. In Supercomputing Conference on High-performance Computing, 1999.

[10] Bhawna, kompal Ahuja "An Efficient Priority Based Task Management In Grid Computing", International Journal for Research in Applied Science and Engineering Technology (IJRASET), Volume 2 Issue III, March 2014, Page No: 76-79

[11]Li Gong. Java security: present and near future. IEEE Micro, 17(3):14{19, May-June 1997.



doi crossref
10.22214/IJRASET

INDEX COPERNICUS
45.98

ISRA JIF
IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL
IMPACT FACTOR:
7.429

ISI

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY