IJCSMS International Journal of Computer Science & Management Studies, Vol. 11, Issue 01, May 2011
ISSN (Online): 2231 –5268
www.ijcsms.com

Grid Technology

Madhu Bhatia

Department of Information Technology, Vaish College of Engineering, M.D.U, Rohtak (India)

Abstract

Grid computing is a means of allocating the computational power of a large number of computers to complex difficult computation or problem. Grid computing is a term referring to the combination of computer resources from multiple administrative domains to reach a common goal. Grid is a form of distributed computing mainly to virtualizes and utilize geographically distributed idle resources. A grid is a distributed computational and storage environment often composed of heterogeneous autonomously managed subsystems. This paper contains the most important aspects of Grid computing. Grid Computing allows high performance distributed system to act as a single computer. An overview of grid structure and techniques is given in order to understand the way Grid work.

Keywords: *Grid Computing, Grid Infrastructure.*

1. INTRODUCTION

The rapid development in computing resources has enhanced the performance of computers and reduced their costs. This availability of low cost powerful computers coupled with the popularity of the Internet and high-speed networks has led the computing environment to be mapped from distributed to Grid environments. Grid computing has emerged as a potential next generation platform for solving large-scale problems in several domains of science and engineering. It generally involves millions of heterogeneous resources scattered across multiple organizations and administrative domains [1, 2]. Management and scheduling of resources in such large-scale distributed systems is complex. Grid provides information and application services to utilize the available resources and meet user demands [3]. Grids use a layer of middleware to communicate with and manipulate heterogeneous hardware and datasets. A grid is defined as a group of nodes that are connected to each other through a number of connecting paths for communication [2]. A grid is a large distributed kind of network that is connected to each other like internet environment from multiple administrative domains (Fig.1)

A grid is having the following characteristics.

1. Resources are connected via a network.
2. Resources are geographically distributed.
3. Resources are managed transparently for performance and fault tolerance.
4. Creates the illusion of virtual organizations and projects without central administration [12].
5. Explicit trust relationship between users and resources.

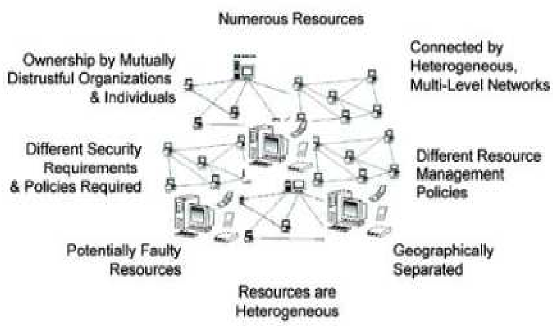

Fig.:-1

It is a type of distributed system that supports sharing and uses of the resources like processor speed, memory, storage, etc. independently from their location. For example the grid connects nearly all university owned computers, including those in labs, in the library, as well as faculty and staff offices. The software that runs in grid gives local users priority for those machines, but when they are idle, their processors can be used over the grid. A typical college or university campus, most of the computers are idle for a long period. A grid can provide significant processing power for users with extraordinary demands. For example animation software which is used by students needs a larger amount of processing power. Grids use a layer of middle ware to communicate with and manipulate heterogeneous hardware and data sets. Grid computing is more powerful for solving complex computational problems than single computer or super computer because of its property of resource sharing. Grid computing solves large computational problems that require a larger number of computer processing cycles and data at same time by sharing the resources from multiple nodes.

Grid computing is having following properties

1. The resources in grid computing are not centrally controlled.
2. Various standards are used.
3. Different hardware and different resources are used by different nodes
4. Every node in grid computing having its scheduling policy.
5. Any time any computer resource can add or delete.
6. Non-trivial qualities of services are achieved.

2. PROPERTIES

Grids, by their very nature, have a set of latent properties that extend those of a traditional computer and make grid computing compelling. These include, but are not necessarily limited to:

Massive scaling and throughput: Network connectivity enables groups of hardware and software components to be effectively combined to achieve greater performance and scaling. You can apply far more resources to a particular workload across a network than you ever could within a single, traditional computer. This may result in greater throughput for a transactional application or perhaps a shorter completion time for a computationally intensive workload.

Inherent resilience and availability: The use of multiple, replicated components within a grid enable greater resilience and availability.

Mutability and flexibility that result in greater efficiency and agility: Grids also offer the opportunity for true economies of scale through the sharing of a large pool of resources among many sets of workloads with differing profiles. There is sharing in the sense that a single resource may be shared by multiple applications at the same time—for example, two applications, or software components, consolidated onto a single physical server—and sharing in the sense that a resource may be repurposed over time, so that one application replaces another as needs change. This leads to potentially greater efficiency or utilization. In addition, the mechanisms that enable repurposing of resources give the grid a mutability that can translate into greater agility or responsiveness.

Service-oriented. The final distinguishing property of grid computing is a focus on managing applications and services rather than the individual resources within the fabric. The sheer number of resources that must be coordinated becomes so vast that more abstract management objects, such as a tier of a service or a complete business application, must be used so that management can scale.

3. ISSUES

In any real time distributed system there are three main issues.

1. Feasibility- this means that a task running should be finished on its deadline even though there is a fault in the system. Dead line in real time system is the major issue because there is no meaning of such a task which is not finishing before its deadline. So the question is that which method is to be applied by which the task can finish on deadline in the presence of fault.

2. Reliability- in real time distributed system reliability means availability of end to end services and the ability to experience failures or systematic attacks, without impacting customers or operations.

3. Scalability–it is about the ability to handle growing amount of work, and the capability of a system to increase total throughput under an increased load when resources are added.

Grid computing operates on these technology principles.

1. Standardization-IT departments have enjoyed greater interoperability and reduced their systems management overhead by standardizing on operating systems, servers, storage hardware, middleware components, and network components. Standardizing also helps reduce operational complexity in the data centre by simplifying application deployment, configuration and integration.

2. **Virtualization-**Virtualizing IT resources means that applications are not tied to specific server, storage, or network components and can use any virtualized IT resource. Virtualization occurs through a sophisticated software layer that hides the underlying complexity of IT resources and presents a simplified, coherent interface used by applications and other IT resources.

3. Automation-Because of the potentially large number of components—both virtual and physical—grid computing demands large-scale automation of IT operations. Each component requires configuration management, on-demand provisioning, top-down monitoring, and other management tasks. A grid management solution must ensure that infrastructure cost savings do not evaporate as a result of hiring additional staff to manage the grid. IT administrators need a top-down view from the end-user or application level so they can effectively measure service levels and proactively resolve problems. Combining these capabilities into a single, automated, integrated solution for managing grids gives organizations a maximum return on their grid investment.

4. KEY COMPONENTS OF GRID COMPUTING

- Resource management: the grid must be aware of what resources are available for different tasks.
- Security management: the grid needs to take care that only authorized users can access and use the available resources.
- Data management: data must be transported, cleansed, and processed.
- Services management: users and applications must be able to query the grid in an effective and efficient manner.

5. GRID ARCHITECTURE MODEL

A new architecture model and technology has been developed for the establishment and management of cross-organizational resource sharing. This new architecture, called *grid architecture*, identifies the basic components of a grid system.

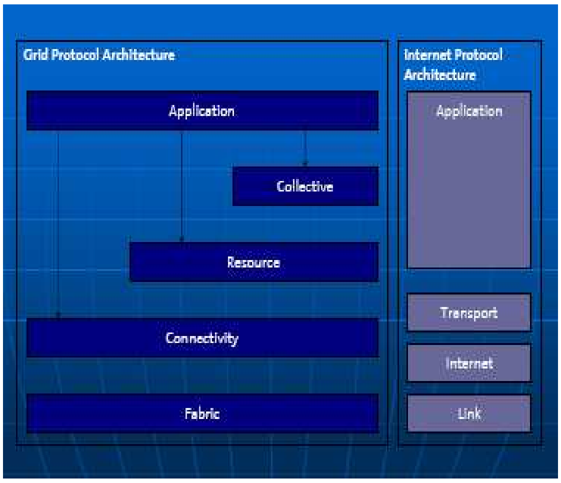

Figure 2: Grid Architecture Model

The grid architecture defines the purpose and functions of its components, while indicating how these components interact with one another. The main focus of the architecture is on interoperability among resource providers and users in order to establish the sharing relationships. This interoperability, in turn, necessitates common protocols at each layer of the architectural model, which leads to the definition of a grid protocol architecture. This protocol architecture defines common mechanisms, interfaces, schema, and protocols at each layer, by which users and resources can negotiate, establish, manage, and share resources. Figure 2 shows the component layers of the grid architecture and the capabilities of each layer. Each layer shares the behavior of the underlying component layers. The following describes the core features of each of these component layers, starting from the bottom of the stack and moving upward.

- *Fabric layer*—the fabric layer defines the interface to local resources, which may be shared. This includes computational resources, data storage, networks, catalogs, software modules, and other system resources.
- *Connectivity layer*—the connectivity layer defines the basic communication and authentication protocols required for grid-specific networking-service transactions.
- *Resource layer*—this layer uses the communication and security protocols (defined by the connectivity layer) to control secure negotiation, initiation, monitoring, accounting, and payment for the sharing of functions of individual resources. The resource layer calls the fabric layer functions to access and control local resources. This layer only handles individual resources, ignoring global states and atomic actions across the resource collection pool, which are the responsibility of the collective layer.
- *Collective layer*—while the resource layer manages an individual resource, the collective layer is responsible for all global resource management and interaction with collections of resources. This protocol layer implements a wide variety of sharing behaviours using a small number of resource-layer and connectivity-layer protocols.
- *Application layer*—the application layer enables the use of resources in a grid environment through various collaboration and resource access protocols.

6. GRID COMPUTING INFRA - STRUCTURE

Grid computing application projects can create their own Grid, that is, a network of resources for that particular project. Grid networks have been set up at a local level, national level, and international level throughout the world to promote Grid computing.

National Grids: Many countries have initiated national Grid computing projects around their high speed networks.Some national Grid networks include Grid-Ireland (Ireland),NorduGrid (Scandinavian grid), DutchGrid (Netherlands), PIONIER (Poland) ACI (France).A national Grid provides collective national computing resources to address major scientific and engineering problems and also problems of national interest such as studying or predicting earthquakes, storms, major environmental disasters, global warming, and terrorism.

Multi-National Grids: In the period 2000-2005, several efforts were started to create Grids that spanned across many countries. For example, ApGrid, a partnership for Grid computing in the Asia Pacific region involved Australia, Canada, China, Hong Kong, India, Japan, Malaysia, New Zealand, Philippines, Singapore, South Korea, Taiwan, Thailand, USA, and Vietnam. There have been several initiatives for European countries to collaborate in forming Grid-like infrastructures to share compute resources funded by European programs.

Campus Grids: Educational institutions also recognized the advantages of Grid technology for cost-effectively sharing computing resources and several institutions set up campus wide grids. Since most campuses nowadays provide wireless connectivity, a wireless Grid is possible for harnessing wireless resources.

7. GRID MIDDLEWARE

The collection of APIs, protocols and software that allows to create and use a distributed system represents a grid middleware. But in a higher level than the network transport protocols, there is a great variety stands on a lower level than the user‘s application of middleware packages, a lot of projects are developing middleware that allow creating some production grids. The most important project is the Globus project. Some other important projects are Legion and Condor, Akenti, NetSolve, Ninf.

Globus Toolkit: A grid software infrastructure‘s approach is represented by the Globus project. Globus develops an integrated set of grid basic services, the set being called Globus toolkit. Globus is different than the other architectures by three aspects: by its services model that can allow the applications to use the grid services without adopting a particularly programming model; by its specialized mechanism that can coexist with (sometimes replace) the mechanism provided by the commodity computing; and by the support offered to a application high demands process approach based on information. The toolkit has a set of components that implements the basic services for security, resource allocation, resource management, communication, etc. The Globus services are listed below:

Service	Service	Description
Resource administration	GRAM	Resource allocation and process management
Communication	Nexus	Unicast and multicast communication services
Information	MDS	Distributed access to information structure and state
Security	GSI	Authentication and security services
Health and status	HBM	Surveying the systems components health and status
Data remote access	GASS	Data remote access via sequential and parallel interfaces
Exe management	GEM	Exe building, keeping in cache memory and localization

8. LIMITATIONS OF GRID COMPUTING

One of the goals of the grid computing is to utilize grid technologies to allow organizations to securely, efficiently, and reliably share both distributed computing power and distributed data. Therefore, an organization that is part of a grid can submit and run jobs on high performance clusters at multiple locations. Today, many businesses are using grid computing technologies, since they are becoming more geographically dispersed and dependent on external service providers [4].

The challenge is that participating organizations often have different security protocols on their respective campuses. These security protocols involve layers of firewalls with configurations that are unknown to the incoming traffic. Clearing all the firewalls and security obstacles presents a major challenge.

Another limitation is that grid computing involves distributing the main computations over some number of clusters that might be heterogeneous and often of different size. Thus, the performance gains are limited to the performance, availability, and reliability of the local cluster(s).[11] Depending on the organization, the size of a local cluster may be restricted by factors such as budget, staff, and space.

9. CONCLUSION

The theory behind "Grid Computing." is not to buy more resources but to borrow the power of the computational resources you need from where it's not being used”. Grid Resources fall into the categories of computation (i.e. a machine sharing its CPU) storage (i.e. a machine sharing its RAM or disk space), communication (i.e. sharing of bandwidth or a communication path), software and licenses and special equipment (i.e. sharing of devices)[12].Grid Computing enables aggregation and sharing of geographically distributed computational, data and other resources as single, unified resource for solving large scale compute and data intensive computing application. Management of these resources is an important infrastructure in the grid computing environment. It becomes complex as the resources are geographically distributed, heterogeneous in nature, owned by different individual or organizations with their own policies, have different access and cost models, and have dynamically varying loads and availability.

References

[1] Heba Kurdi, Maozhen Li, and Hamed Al-Raweshidy, AClassification of Emerging and Traditional GridSystems,IEEE Distributed Systems Online, vol. 9, no. 3, 2008, art. no.0803-3001.

[2] I. Foster, C. Kesselman, and S. Tuecke, The anatomy of the grid, enabling scalable virtual organization , International Journal on Supercomputer Applications, vol. 15, no. 3, 2001.

[3] I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke, The physiology of the grid, an open grid services architecture for distributed systems integration.

[4] [Lingfen Sun and Emmanuel C. Ifeachor, The impact of grid on healthcare.

[5] Pin Hu, Lingfen Sun and Emmanuel Ifeachor, An Approach to Structured Knowledge Representation of Service-oriented Grids.

[6] S. M. Park, Y.-B. Ko. and J. H. Kim, "Disconnected Operation Service in Mobile Grid Computing", International Conference on Service Oriented Computing (ICSOC' 2003), Trento, Italy, Dec 2003.

[7] Foster, I. and Kesselman, C, "The Grid: Blueprint for a New Computing Infrastructure," Morgan Kaufmann, 1999.

[8] A. Abbas - Grid Computing: A Practical Guide to Technology and Applications, Charles River Media, 2004

[9] Heba Kurdi, Maozhen Li, and Hamed Al-Raweshidy, A Classification of Emerging and Traditional Grid Systems, IEEE Distributed Systems Online, vol. 9, no. 3, 2008, art. no.0803-3001.

[10] Mehdi Bahrami and Ahmad Faraahi, "AGC4ISR,New Software Architecture for Autonomic Grid Computin", 2010 International Conference on Intelligent Systems, Modelling and Simulation 978-0-7695-3973-7/10 © 2010 IEEE

[11] H.M. Faheem, "Accelerating Motif Finding Problem using Grid Computing with Enhanced Brute Force", ISBN 978-89-5519-146-2

[12] Inderpreet Chopra, "Analysing the need for Autonomic Behaviour in Grid Computing", 978-1-4244-5586-7/10/ 2010 IEEE

[13] Naidila Sadashiv, "Cluster, Grid and Cloud Computing: A Detailed Comparison", The 6th International Conference on Computer Science & Education (ICCSE 2011) August 3-5, 2011. SuperStar Virgo, Singapore

[14] GlobusSimpleCA,http://www.globus.org/security/simple-ca.html, 2004.

[15] Browne, J.C. "Grid computing as applied distributed computation: a graduate seminar on Internet and Grid computing", Cluster Computing and the Grid, 2004. CCGrid 2004. IEEE Int'l Symposium, 19-22 April 2004, pp 239 – 245.

[16] Irving, M; Taylor, G.; Hobson and P."Plug in to Grid Computing", *Power and Energy Magazine*, IEEE, Volume 2, Issue 2, Mar-Apr 2004, pp 40 – 44.

[17] *IT Professional* "Grid computing 101: what's all the fuss about?**",** Volume 6, Issue 2, March-April 2004, pp 25 – 33.

[18] Dai, Y.S.; Xie, M.; and Poh, K.L. "Reliability Analysis of Grid Computing Systems" Dependable Computing, 2002. Proceedings. 2002 Pacific Rim International Symposium, 16-18 Dec. 2002, pp: 97 – 104.

[19] A Networking Approach to Grid Computing , pp 1-100, by Daniel Minoli, John Wiley & Sons.