ISSN 2250-3153

Grid Computing Architecture and Benefits

Shruti N. Pardeshi[1], Chitra Patil[2], Snehal Dhumale[3]
Lecturer,computer department ,SSBT's COET,Bambhori

Abstract- Grid Computing has become another buzzword after Web 2.0. However, there are dozens of different definitions for Grid Computing and there seems to be no consensus on what a Grid is.

This paper describes about Grid Computing. In various ways we are trying to explain grid computing along with its architecture and the standards available for grid computing. Then at last we have discussed about the benefits of grid computing.

Index Terms- Grid Computing, connectivity layer, resource layer, application layer, Open Grid Service Architecture (OGSA). Open Grid Services Interface (OGSI), OGSA-DAI (data access and integration), and Web Services Resource Framework (WSRF).

I. Introduction

GRID computing [1, 2] is a technology for coordinating large scale resource sharing and problem solving among various autonomous group. Grid technologies are currently distinct from other major technical trends such as internet, enterprise distributed networks and peer to peer computing. Also it has some embracing issues in QoS, data management, scheduling, resource allocation, accounting and performance.

Grids are built by various user communities to offer a good infrastructure which helps the members to solve their specific problems which are called a grand challenge problem.

A grid consists of different types of resources owned by different and typically independent organizations which results in heterogeneity of resources and policies. Because of this, grid based services and applications experience a different resource behavior than expected.

Similarly, a distributed infrastructure with ambitious service put more impact on the capabilities of the interconnecting networks than other environments.

Grid High Performance Network Group [3] works on network research, grid infrastructure and development. In their document the authors listed six main functional requirements, which are considered as mandatory requirements for grid applications.

They are:

i) High performance transport protocol for bulk data transfer,
ii) Performance controllability,
iii) Dynamic network resource allocation and reservation,
iv) Security,
v) High availability and
vi) Multicast to efficiently distribute data to group of resources.

Grid computing can mean different things to different individuals. The grand vision is often presented as an analogy to power grids where users (or electrical appliances) get access to electricity through wall sockets with no care or consideration for where or how the electricity is actually generated.

In this view of grid computing, computing becomes pervasive and individual users (or client applications) gain access to computing resources (processors, storage, data, applications, and so on) as needed with little or no knowledge of where those resources are located or what the underlying technologies, hardware, operating system, and so on.

Grid computing could be defined as any of a variety of levels of virtualization along a continuum. Exactly where along that continuum one might say that a particular solution is an implementation of grid computing versus a relatively simple implementation using virtual resources is a matter of opinion. But even at the simplest levels of virtualization, one could say that grid-enabling technologies

II. Architecture

Grids started off in the mid-90s to address large-scale computation problems using a network of resource-sharing commodity machines that deliver the computation power affordable only by supercomputers and large dedicated clusters at that time.
The major motivation was that these high performance computing resources were expensive and hard to get access to, so the starting point was to use federated resources that could comprise compute, storage and network resources from multiple geographically distributed institutions, and such resources are generally heterogeneous and dynamic.

Grids focused on integrating existing resources with their hardware, operating systems, local resource management, and security infrastructure. In order to support the creation of the so called "Virtual Organizations"—a logical entity within which distributed resources can be discovered and shared as if they were from the same organization, Grids define and provide a set of standard protocols, middleware, toolkits, and services built on top of these protocols. Interoperability and security are the primary concerns for the Grid infrastructure as resources may come from different administrative domains, which have both global and local resource usage policies, different hardware and software configurations and platforms, and vary in availability and capacity.

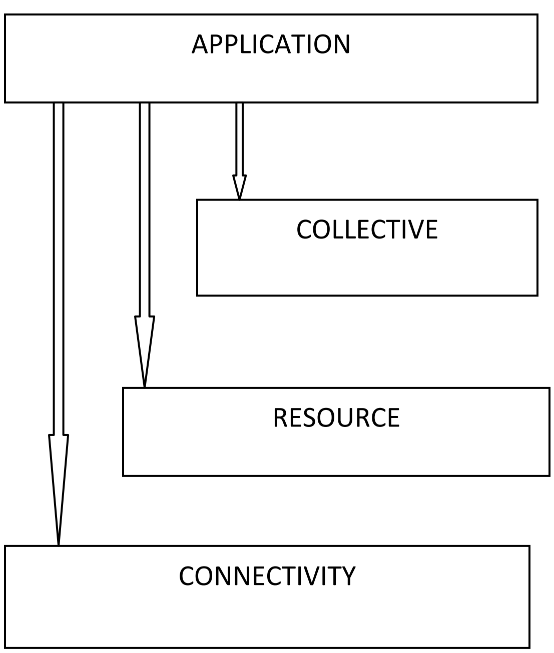
Grids provide protocols and services at five different layers as identified in the Grid protocol architecture (see Figure 1). At the **Fabric layer**, Grids provide access to different resource types such as compute, storage and network resource, code repository, etc. Grids usually rely on existing fabric components, for instance, local resource managers.

General-purpose components such as GARA (general architecture for advanced reservation) [4],and specialized resource management services such as Falkon [5]

Connectivity layer defines core communication and authentication protocols for easy and secure network transactions. The GSI (Grid Security Infrastructure) [6] protocol underlies every Grid transaction.

The **Resource layer** defines protocols for the publication, discovery, negotiation, monitoring, accounting and payment of sharing operations on individual resources.

The GRAM (Grid Resource Access and Management) [7] protocol is used for allocation of computational resources and for monitoring and control of computation on those resources, and GridFTP [8] for data access and high-speed data transfer.
The **Collective layer** captures interactions across collections of resources, directory services such as MDS (Monitoring and Discovery Service) [9] allows for the monitoring and discovery of VO resources, Condor-G and Nimrod-G are examples of co-allocating, scheduling and brokering services, and MPICH [10] for Grid enabled programming systems, and CAS (community authorization service) [11] for global resource policies.



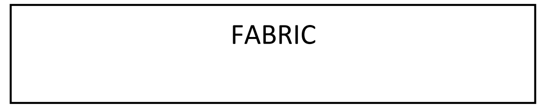


Fig 1. Grid Protocol Architecture

The **Application layer** comprises whatever user applications built on top of the above protocols and APIs and operate in VO environments.

III. Standards for grid Environments

OGSA

The Global Grid Forum has published the Open Grid Service Architecture (OGSA). To address the requirements of grid computing in an open and standard way, requires a framework for distributed systems that support integration,virtualization, and management. Such a framework requires a core set of interfaces, expected behaviors, resource models, and bindings.[12]

OGSA defines requirements for these core capabilities and thus provides a general reference architecture for grid computing environments. It identifies the components and functions that are useful if not required for a grid environment.

OGSI

As grid computing has evolved it has become clear that a service-oriented architecture could provide many benefits in the implementation of a grid infrastructure. The Global Grid Forum extended the concepts defined in OGSA to define specific interfaces to various services that would implement the functions defined by OGSA.

More specifically, the Open Grid Services Interface (OGSI) defines mechanisms for creating, managing, and exchanging information among Grid services. A Grid service is a Web service that conforms to a set of interfaces and behaviors that define how a client interacts with a Grid service. These interfaces and behaviors, along with other OGSI mechanisms associated with Grid service creation and discovery, provide the basis for a robust grid environment. OGSI provides the Web Service Definition Language (WSDL) definitions for these key interfaces.

OGSA-DAI

The OGSA-DAI (data access and integration) project is concerned with constructing middleware to assist with access and integration of data from separate data sources via the grid. The project was conceived by the UK Database Task Force and is working closely with the Global Grid Forum DAIS-WG and the Globus team.[12]

GridFTP

GridFTP is a secure and reliable data transfer protocol providing high performance and optimized for wide-area networks that have high bandwidth. As one might guess from its name, it is based upon the Internet FTP protocol and includes extensions that make it a desirable tool in a grid environment. The GridFTP protocol specification is a proposed recommendation document in the Global Grid Forum (GFD-R-P.020).[12]

GridFTP uses basic Grid security on both control (command) and data channels. Features include multiple data channels for parallel transfers, partial file transfers, third-party transfers, and more.

WSRF

Web Services Resource Framework (WSRF). Basically, WSRF defines a set of specifications for defining the relationship between Web services (that are normally stateless) and stateful resources.[12]

Web services related standards

Because Grid services are so closely related to Web services, the plethora of standards associated with Web services also apply to Grid services. We do not describe all of these standards in this document, but rather recommend that the reader become familiar with standards commonly associate with Web services,such as:
_ XML
_ WSDL
_ SOAP
_ UDDI

IV. Benefits of Grid Computing

Exploiting underutilized Resources:

One of the basic uses of grid computing is to run an existing application on a different machine. The machine on which the application is normally run might be unusually busy due to a peak in activity. The job in question could be run on an idle machine elsewhere on the grid.

Parallel CPU capacity

The potential for massive parallel CPU capacity is one of the most common visions and attractive features of a grid. In addition to pure scientific needs, such computing power is driving a new evolution in industries such as the bio-medical field, financial modeling, oil exploration, motion picture animation, and many others.

Virtual resources and virtual organizations for collaboration

Another capability enabled by grid computing is to provide an environment for collaboration among a wider audience. In the past, distributed computing promised this collaboration and achieved it to some extent.[12]

Access to additional resources

As already stated, in addition to CPU and storage resources, a grid can provide access to other resources as well. The additional resources can be provided in additional numbers and/or capacity.[12]

Resource balancing

grid federates a large number of resources contributed by individual machines into a large single-system image. For applications that are grid-enabled, the grid can offer a resource balancing effect by scheduling grid jobs on machines with low utilization.[12]

Reliability

High-end conventional computing systems use expensive hardware to increase reliability. They are built using chips with redundant circuits that vote on results, and contain logic to achieve graceful recovery from an assortment of hardware failures.

Management

The goal to virtualized the resources on the grid and more uniformly handle heterogeneous systems will create new opportunities to better manage a larger,
more distributed IT infrastructure. It will be easier to visualize capacity and utilization, making it easier for IT departments to control expenditures for computing resources over a larger organization

V. Conclusion

In this we have described about grid computing and its architecture. All the protocols considered under grid computing and benefits of grid computing. In future we will discuss about the security issues of grid computing and try to provide specific solution for the problem of security.

References

[1] Foster and C. Kesselman, "The Grid: Blue print for a new computing infrastructure", Morgan Kaufmann Publications (1999).

[2] Foster, C. Kesselman, J. M. Nick and S. Tuecke, "The physiology of the Grid: An open grid services architecture for distributed systems integration", Grid Forum white paper, 2003.

[3] Volker Sander, "Networking Issues for Grid Infrastructure", GFD-I.037, Nov, 22, 2004.

[4] I. Foster, C. Kesselman, C. Lee, R. Lindell, K. Nahrstedt, A.Roy. "A Distributed Resource Management Architecture that Supports Advance Reservations and Co-Allocation", Intl Workshop on Quality of Service, 1999.

[5] I. Raicu, Y. Zhao, C. Dumitrescu, I. Foster, M. Wilde. "Falkon:a Fast and Light-weight tasK executiON framework",IEEE/ACM SuperComputing 2007.

[6] The Globus Security Team. "Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective," Technical Report, Argonne National Laboratory, MCS, 2005.

[7] I. Foster, C. Kesselman. "Globus: A Metacomputing Infrastructure Toolkit", Intl J. Supercomputer Applications, 11(2):115-128, 1997.

[8] B. Allcock, J. Bester, J. Bresnahan, A. L. Chervenak, I. Foster,C. Kesselman, S. Meder, V. Nefedova, D. Quesnal, S. Tuecke."Data Management and Transfer in High Performance Computational Grid Environments", Parallel Computing Journal, Vol. 28 (5), May 2002, pp. 749-771.

[9] J. M. Schopf, I. Raicu, L. Pearlman, N. Miller, C. Kesselman, I.Foster, M. D'Arcy. "Monitoring and Discovery in a Web Services Framework: Functionality and Performance of Globus Toolkit MDS4", Technical Report, Argonne National Laboratory, MCS Preprint #ANL/MCS-P1315-0106, 2006.

[10] N. Karonis, B. Toonen, and I. Foster. MPICH-G2: A Grid- Enabled Implementation of the Message Passing Interface.Journal of Parallel and Distributed Computing, 2003.

[11] I. Foster, C. Kesselman, L. Pearlman, S. Tuecke, and V. Welch."The Community Authorization Service: Status and Future," In Proc. of Computing in High Energy Physics (CHEP), 2003.

[12] Introduction to Grid Computing by Bart Jacob
Michael Brown, Kentaro Fukui, Nihar Trivedi

Author details :

Shruti N. Pardeshi was born in Jalgaon in 1988.She received her B.E. degree in Computer from North Maharashtra University, Jalgaon in 2009 and currently working as a lecturer in computer department ,SSBT's COET ,Bambhori , Jalgaon.

Chitra J. Patil was born in Jalgaon in 1991.She received her B.E. degree in Computer from North Maharashtra University, Jalgaon in 2012 and currently working as a lecturer in computer department ,SSBT's COET ,Bambhori , Jalgaon.

Snehal Dhumale : was born in Bhusawal in 1990..She received her B.E. degree in Information Technology from pune University, Jalgaon in 2011 and currently working as a lecturer in computer department ,SSBT's COET ,Bambhori , Jalgaon.