Grid Computing

V Rajaraman

A computing grid interconnects resources such as high performance computers, scientific databases, and computer-controlled scientific instruments of cooperating organizations each of which is autonomous. It precedes and is quite different from cloud computing, which provides computing resources by vendors to customers on demand. In this article, we describe the grid computing model and enumerate the major differences between grid and cloud computing.

V Rajaraman is at the Indian Institute of Science, Bengaluru. Several generations of scientists and engineers in India have learnt computer science using his lucidly written textbooks on programming and computer fundamentals. His current research interests are parallel computing and history of computing.

Introduction

We discussed in an earlier article [1] how the all-pervasive Internet, rapid increase in the speed of computers, size of storage space, and network bandwidth led to the emergence of cloud computing. The same improvements, particularly the enormous increase in the communication bandwidth and the steep reduction in the cost of accessing computing resources situated at geographically dispersed locations, led to the emergence of grid computing. There have been two major shifts in scientific research. One of them is 'Big Science' which attempts to solve the so-called 'grand challenge' problems [2] as a cooperative endeavour involving scientists working in several universities and research laboratories spread across the world, many of them with a high performance computer called a supercomputer, specialized instruments, software systems, and specialized databases. The second is the emergence of realistic computer simulation as an important tool in scientific research. Formulating hypotheses and validating them by experiments was the major paradigm used by scientists for over a century. High cost and large amount of time required to perform experiments and the emergence of supercomputers led to the use of computer simulation as an intermediate step before embarking on costly experiments. Realistic simulation of models, even at an atomic scale, became feasible with the emergence of supercomputers.

Keywords
Grid computing, cooperative computing, volunteer computing, comparison with cloud computing.

Ian Foster and his group [3] at the Argonne National Laboratory, USA and Carl Kesselman at the University of Southern California conceived the idea of interconnecting resources of cooperating organizations in the late 90s. They called it the *computing grid*.

Supercomputers are expensive and not every university can afford one. It is important to allow access to these machines to scientists who need them. Scientific instruments have become highly specialized and very expensive. There is a need to share them among cooperating scientists. Fortunately, most sophisticated scientific instruments today are computer-controlled and may be accessed and used from remote locations provided there is cooperation among the institutions. In practice this has not happened due to practical difficulty of using instruments remotely. Specialized application software takes a long time to develop and the development cost is very high. Not every organization has the capability to develop them. Thus, there is a need for sharing them. Huge volumes of data are generated by experiments such as Large Hadron Collider, Laser interferometer Gravitational Wave Observatory, and the Sloan Digital Sky Survey which need analysis by teams of scientists working worldwide. Thus, the need for cooperation and sharing resources among scientists was widely felt. Ian Foster and his group [3] at the Argonne National Laboratory, USA and Carl Kesselman at the University of Southern California conceived the idea of interconnecting resources of cooperating organizations in the late 90s. They called it the *computing grid*. The term 'grid' was borrowed from 'power grid', where widely distributed power generating stations are interconnected to supply electrical power to consumers. The history of the development of grid computing is given in *Box* 1.

Defining Grid Computing

After conceiving the idea of grid computing, Ian Foster and his group started a project called *Globus* as a cooperative effort of several interested organizations (mainly universities and national laboratories in USA) to implement the idea. The Globus project (www.globus.org) defined grid computing as:

"An infrastructure that enables the integrated, collaborative use of high-end computers, networks, databases, and scientific instruments owned and managed by multiple organizations".

Box 1. Timeline of Grid Computing Development

1965	Multics project at MIT introduces time-shared computer systems
1980	ARPA project leads to the emergence of the Internet
1990	Emergence of parallel and distributed computing
1992	Meta computing – analogy with power grid
1995	I-WAY project demonstrates distributed virtual supercomputing by hosting 60 applications on a national test bed.
1996	Globus project initiated
1997	Globus application demonstrated – called computational grid
1997	Legion project – Vision of world wide virtual computers
1997	UNICORE (*U*niform *I*nterface to *C*omputing *R*esources) initiated in Germany
1998	A book titled "The Grid: Blue Print for a New Computing Infrastructure - I.Foster and C.Kesselman (editors)" published by Morgan Kaufman
2004	Global grid forum formed
2006	National grid computing initiatives – Garuda (India), Data Grid, Tera Grid (NSF-USA), NaReGI (Japan), NES (UK), European Grid Infrastructure

It was later refined to include the fact that there was no centralized control. The addition to the definition stated:

"A system that coordinates resources that is not subject to centralized control using standards, open general purpose protocols, and interfaces to deliver non trivial quality of service."

Another definition which emphasizes aggregation of distributed high-performance computers to obtain extremely high computing capacity to solve a single problem is given below:

"Grid computing provides the ability using a set of open standards and protocols to gain access to applications and data, processing power, storage capacity, and vast array of computing resources over the Internet. It is a type of parallel and distributed system that enables the sharing, selection, and aggregation of resources distributed across multiple administrative domains based on their availability, capacity, performance, cost, and users' quality of service requirements."

Grid computing is a system that coordinates resources that is not subject to centralized control using standards, open general purpose protocols, and interfaces to deliver non trivial quality of service.

Based on these definitions, the important keywords are:

- The resources are owned and operated by independent organizations *not subject to centralized control*.

Computers belonging to member organizations may be configured to work cooperatively (in parallel) to solve a single large problem.

- Resources may be aggregated, if necessary, to deliver the requested computing power.
- Open (i.e., not proprietary) standards and protocols are to be used to allow cooperation of member organizations.
- It is a type of parallel and distributed system.
- Quality of Service should be ensured.

We may infer from these definitions and keywords that a grid computing system is one which has the following features:

- Facilities, namely, high performance computing, scientific instruments, specialized software, and databases that are owned by organizations (mostly universities and research laboratories) which have their own policies for use of these facilities. They permit other organizations to use these facilities on specified terms.
- Computers belonging to member organizations may be configured to work cooperatively (in parallel) to solve a single large problem.
- The organizations agree to cooperate and form a grid which now belongs to a 'virtual organization'. As part of the cooperation agreement, policies are formulated to ensure a specified QoS, the parameters of which may include a specified time within which the requested resource should be made available. The availability should be guaranteed for a specified time.
- Normally, remote facilities will be used by an organization only when it is not locally available.
- An organization using a remote facility should not compromise its own security or that of the host organization.
- There must be a system which allows an organization which is a member of a grid to know what facilities are available in the grid, the policies of usage, and QoS parameters.

The organizations agree to cooperate and form a grid which now belongs to a 'virtual organization'. As part of the cooperation agreement, policies are formulated to ensure a specified. QoS, the parameters of which may include a specified time within which the requested resource should be made available. The availability should be guaranteed for a specified time.

Software Components of a Grid Infrastructure

Even though the original conception of grid computing encompassed the use of scientific instruments from remote locations, almost all major implementations of grid computing emphasize only the use of computing resources due to practical difficulty of accessing instruments from remote locations as pointed out earlier. Thus, the discussions in this article are primarily focused on sharing computing resources. The important software components required to support a grid computing infrastructure must perform the following functions:

Software components required to support a grid computing infrastructure are organized as a layered architecture with each layer supporting a set of software components.

- Define the communication protocol to be used. Authenticate a user and determine the user's access rights to specific resources and the prescribed limits on the use of these resources.
- Maintain a directory of resources and the cost for their use. Cost may not be money but specified hours of use of the resource based on project approval by a peer group to whom the project proposal is submitted.
- Interpret user's request for resources and forward the request in a specified form to a resource manager. Whenever appropriate, quote the cost for using the resources.
- Give time estimate for completing the job. (This may be one of the QoS parameters).
- Schedule user's job in one or more computers and monitor its progress.
- Ensure security and privacy of data sent/received from the grid infrastructure by encrypting data which is the responsibility of the user. Provide access to only registered users.
- Log diagnostic messages and forward them to the user. Send the results to the user when the job is completed.

These software components are organized as a layered architecture shown in *Table* 1 [4]. In a layered architecture, each layer supports a set of software components. The upper layer components depend on the services provided by the lower layers. (See *Figure* 1).

The software suites needed to support grid infrastructure described so far are so huge that it is impractical for a single organization to develop them. A large number of research groups working in many universities, research laboratories, and industry

Name of Layer	Components in Layer
Application Layer	Scientific and engineering applications (e.g., Tera grid secure gate way)
	Collaborative computing environment (e.g., National Virtual Observatory)
	Collaborative databases (e.g., data generated by the Large Hadron Collider)
Collective Layer	Directory of available services
	Resource discovery
	Brokering (resource management, selection, aggregation)
	Diagnostics
	Usage policy implementation
Resource Layer	Resource allocation
	Monitoring use of resources
	Calculating usage cost
	Collecting payment
	Monitoring Quality of Service
Connectivity Layer	Communication protocols
	User authentication protocols
	Secure access to services
	Encryption, privacy, protection from malicious software
Physical Layer	High performance computers
	Networks
	Storage media
	Instruments
	Databases
	Software library

Table 1. Layered architecture of grid infrastructure.

participants developed an open source tool kit called Globus, which consists of large suites of programs (often known as *middleware*). A global *open grid forum* (www.ogf.org) meets regularly to evolve standards to be followed in creating programs to be included in the Globus kit.

Invoking a Grid Infrastructure

The steps to be followed by a user to utilize the grid infrastructure are:

1. The application is developed as a distributed program using tools provided in the application layer. If it is a request to use other resources, the appropriate protocol is used to submit it.

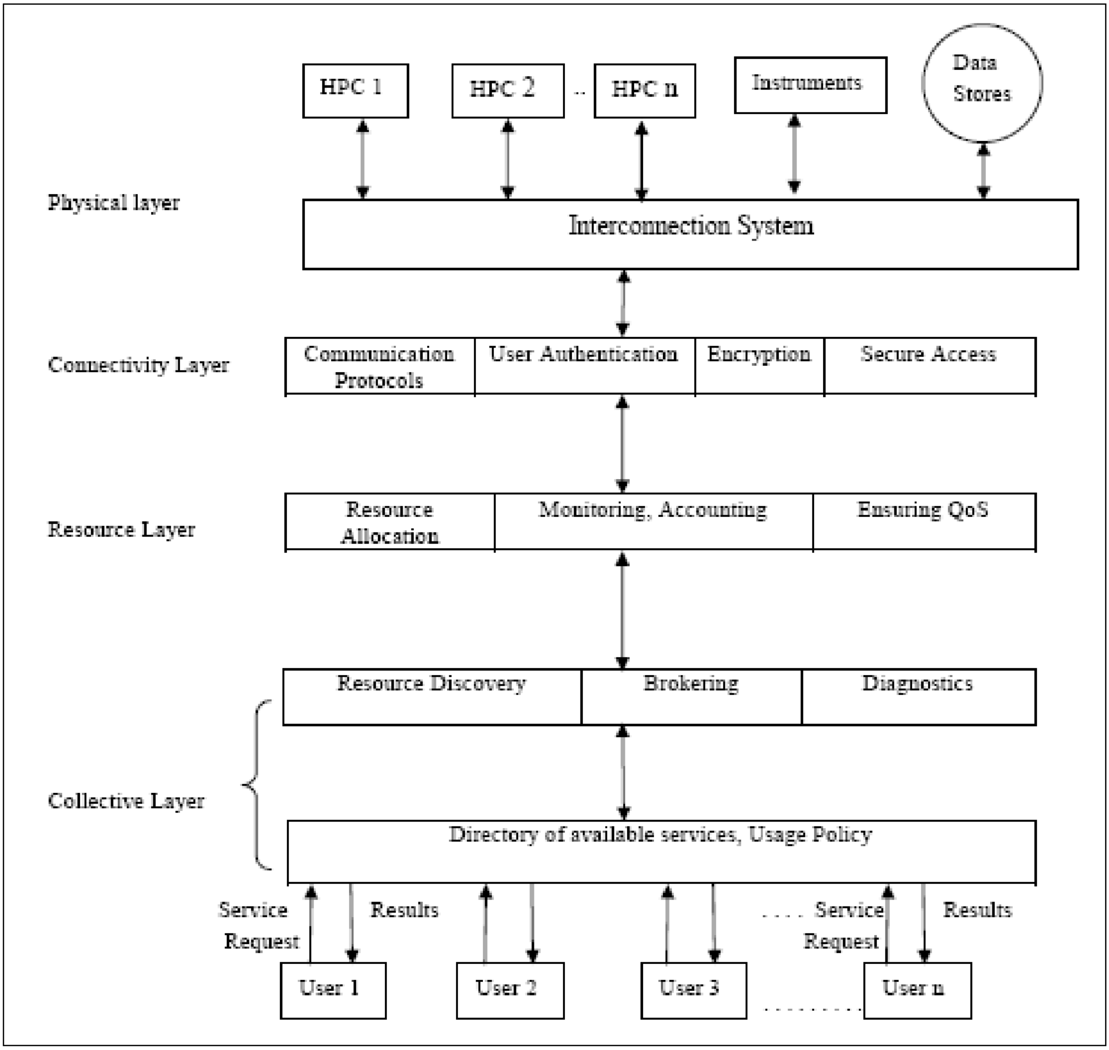


Figure 1. Layered architecture of grid computing. (HPC stands for High Performance Computer.)

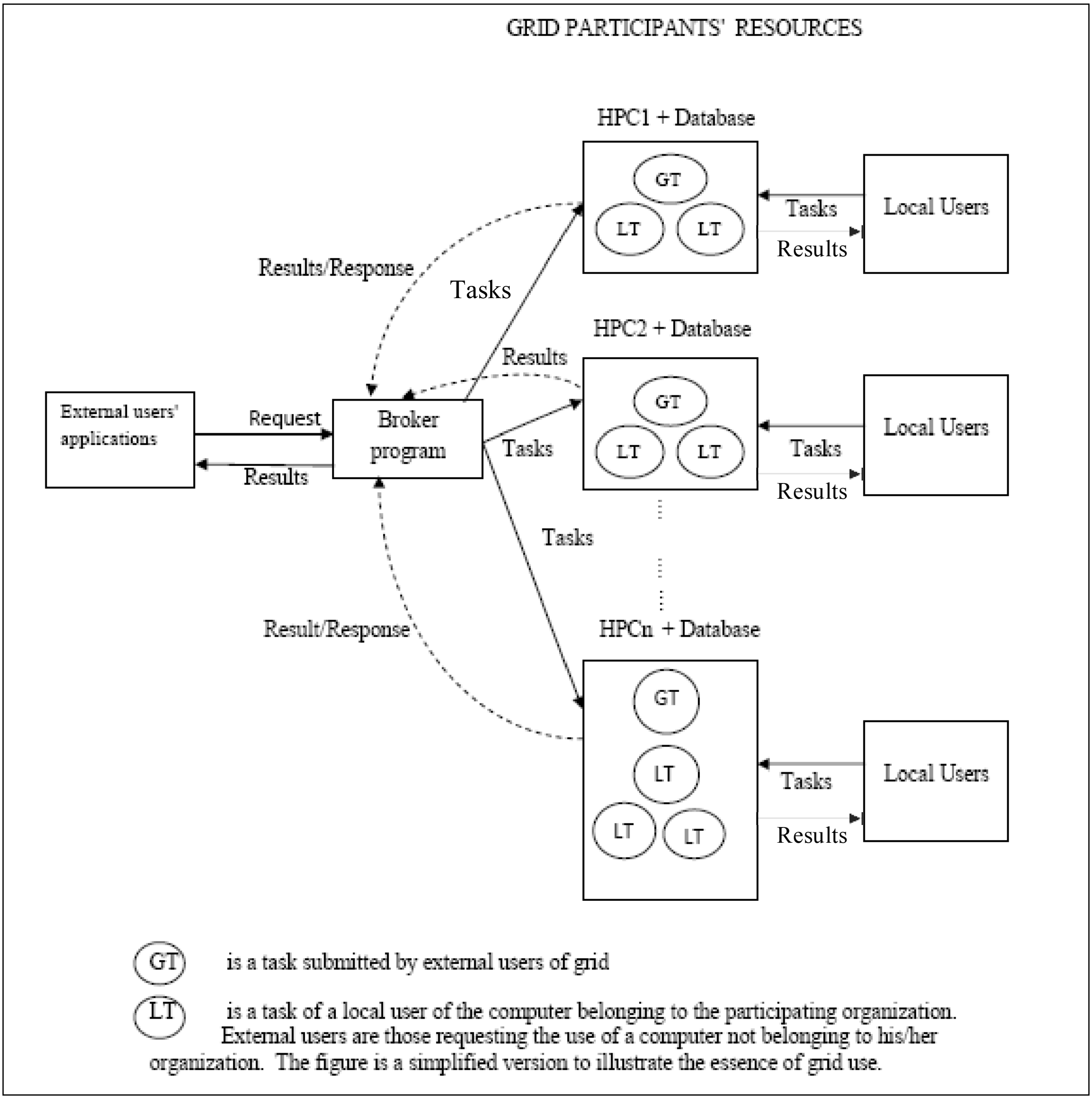
2. The application is submitted to a broker (broker is a program available in the collective layer) specifying computing, storage space, software, and database resources needed. QoS parameters are also specified.

3. The broker finds the optimal set of resources which are available and can be scheduled to meet the user's requirements including the QoS parameters. Then, the broker estimates the cost and the user's capacity to pay based on available credit in the user's account. The cost estimate is usually sent back to the user to obtain his/her consent.

4. The broker schedules the task on the identified resources. The progress of the job is monitored. Any problem encountered during execution is notified to the user.
5. On completion of the job, the results along with a bill are sent to the user.

Local users of the institutions continue to use their normal password access. They access other institutions' computers only when local facilities are inadequate. (See *Figure* 2).

Figure 2. Invoking grid infrastructure.



Enterprise Grid

The grid computing environment described in the previous section assumes that several independent organizations (also called administrative domains) cooperate to form a virtual organization. Large public or private sector companies, state governments, and universities often have computing facilities in geographically distributed locations. We call all these types of organizations as large enterprises. From an enterprise point of view, it is desirable to have a global view and optimize the configuration and the use of all the computers, particularly high performance servers wherever they are. In addition it would be desirable to combine the processing power of idle computers to execute processor-intensive jobs. The research done by the grid computer alliance and the availability of Globus software tools motivated large enterprises to mobilize their resources to create an *enterprise grid*. An enterprise grid is designed to interconnect all computers and other resources, particularly databases to create a system accessible to all the members of the enterprise. Several advantages accrue to an enterprise when they create an enterprise grid. Some of them are:

From an enterprise point of view, it is desirable to have a global view and optimize the configuration and the use of all the computers, particularly high performance servers wherever they are.

- Idle computers in one location may be used by users in other locations of the enterprise.
- High performance computers can be accessed by an employee from a remote location from his/her desktop based on need as determined by the management.
- Reliability and QoS of the enterprise's combined resources are improved.
- Hardware and software costs and operational cost are reduced. In addition, the productivity and use of resources of the enterprise are enhanced.

Other benefits which accrue when an enterprise grid is used are:

- Collaboration between employees, and formation of (virtual) teams spread across the enterprise are facilitated. Collaboration very often leads to innovation.
- It enables easy, seamless, and secure access to remote

An enterprise grid is designed to interconnect all computers and other resources, particularly databases to create a system accessible to all the members of the enterprise.

Several vendors have cooperated to establish an enterprise grid alliance (www.ega.org). The objectives of this alliance are to adopt and deploy grid standards including interoperability.

instruments and sensors.

- In case of failure of a computer at one location, the grid architecture allows migration of applications to other locations of the enterprise. This enhances the disaster recovery capability of the enterprise.
- The overall cost of computer infrastructure of the enterprise is reduced.

Several vendors have cooperated to establish an enterprise grid alliance (www.ega.org). The objectives of this alliance are to adopt and deploy grid standards including interoperability. The alliance works closely with Globus consortium. An enterprise grid need not be confined to a single enterprise. It may encompass several independent enterprises which agree to collaborate on some projects sharing resources and data. There are also examples of enterprises collaborating with universities to solve difficult problems (See www.wrgrid.org.uk/leaflets/DAME.pdf).

Comparison of Grid and Cloud Computing

Cloud computing is defined as [5]: "A method of availing computing resources from a provider, on demand, by a customer using computers connected to a network (usually the Internet)". Grid computing was being used before cloud computing emerged. Clouds are managed by private 'for profit' providers. Grid and cloud provide different computing environments. In what follows, we compare the two environments based on six different attributes: business model, architecture, resource management, security model, programming model, and applications [6].

Even though grid and cloud computing both use the Internet to interconnect geographically distributed computers they differ substantially in many respects.

Business Model

Grid is formed by 'not for profit' cooperating organizations (normally universities and research laboratories). Individual organizations that cooperate are *autonomous* and determine the policies of usage of the infrastructure they contribute to the grid. In other words, it is a federal system. Cloud computing systems, in contrast, is run by vendors that maintain massive infrastructure. Hundred thousand servers could be provided to a single customer 'on demand' by a cloud computing vendor. Cloud computing vendors charge customers based on use. A customer selects one among a set of vendors. As of now, it is difficult for a customer to shift programs and data from one vendor to another.

Architecture

Grid is formed by interconnecting heterogeneous computers (usually high performance computers). The infrastructure is federated and uses standard protocols allowing interoperability.

Cloud providers typically use uniform architecture computers and configure them as a huge collection of identical servers known as *warehouse scale systems*. Many users may use a single server or several servers simultaneously (called multi-tenancy) as virtual machines appropriate for their application using virutalization sofware. There are many independent vendors of cloud systems and there is no standardization of protocols and data storage models among vendors. Cloud computing vendors provide Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service(SaaS) as explained in [1].

Grid is formed by interconnecting heterogeneous computers (usually high performance computers). The infrastructure is federated and uses standard protocols allowing interoperability. Cloud providers typically use uniform architecture computers and configure them as a huge collection of identical servers known as *warehouse scale systems*.

Resource Management

In a grid system, resources are provided on availability (not on demand). Grid systems are optimized to provide high performance computing to a select group of members. The software systems are designed to provide publicized services in a static environment. A broker has information on facilities available in the grid and schedules jobs to a group of computers, which are allocated exclusively to a user until the user's job is completed. Jobs are not only requests to use computing resources but (in theory) also computer controlled scientific instruments. Most grids, however, emphasize allocation of CPU cycles and database access rather than other resources. A cloud provider, on the other hand, has massive computing infrastructure and can provide computing power on demand. The systems are multi-tenanted. The aim is to optimize the vendors' resource utilization and maximize their profit.

Grids are mainly used for high performance numerical computing. It is a collaborative model among scientists and engineers providing in addition to computing power, specialized software systems, and massive databases which are cooperatively maintained.

Security Model

Grids use public key-based security infrastructure. Users are registered and their credentials are verified before they become members of the grid community.

A cloud computing vendor is not overly concerned with who uses the infrastructure. Multi-tenancy of computers is common and has associated risk. Data may be located anywhere across the world and may lead to complex legal problems if there is a breach of privacy or loss of data.

Resource Allocation

In a grid computing infrastructure, resources are reserved and allocated based on availability. The allocated resources are normally exclusive for a single user. In a cloud infrastructure, resources are allocated on demand. As there are massive resources and virtualization, a customer would expect instant availability of even up to 10,000 servers. Cloud systems are multi-tenanted as this simplifies allocation of resources by vendors.

Applications

Grids are mainly used for high performance numerical computing. It is a collaborative model among scientists and engineers providing in addition to computing power, specialized software systems, and massive databases which are cooperatively maintained.

Cloud computing provides loosely coupled servers to customers on demand. They are mostly used to host web servers, e-commerce applications, big data analytics, and on-line transaction processing.

Private Cloud and Enterprise Grid

A private cloud [1] is created by a vendor for the exclusive use of one enterprise. It normally consists of a uniform set of servers and is homogeneous. The computing infrastructure is owned by the vendor and the enterprise is charged based on use. An enterprise grid, in contrast, is created by one or more enterprises by combining the infrastructure owned by them using software created by the enterprise grid alliance. The computer systems in the grid are heterogeneous. The grid is maintained and managed by the enterprises. The enterprises have full control of their data and other valuable resources. An enterprise may decide to outsource the maintenance of the infrastructure to a vendor. From the point of view of an enterprise, such as a bank, an enterprise grid would be preferred as the bank will have full control of its infrastructure and it can ensure better security and privacy of its customers' data.

Volunteer computing is a model of distributed Client-Server computing in which owners of computers (clients) donate computing resources to a project (they consider worthwhile) that is initiated by a server.

Volunteer Computing and Grid Computing

Volunteer computing is a model of distributed Client-Server computing in which owners of computers (clients) donate computing resources to a project (they consider worthwhile) that is initiated by a server. The idea originated in the mid 90s but became well known with two science projects (Search for Extra Terrestrial Intelligence (SETI) and Protein folding) which required immense computing power. In this model of computing, a server creates a single program multiple data type of parallel program and distributes the program to all the volunteers. Each volunteer is then given a chunk of data to be used as input to the program. Volunteers' computers run the program given to them in the background. Whenever a volunteer is not using his/her computer it will not be switched off and the program will run on the computer. As soon as a computer exhausts the data given to it for processing, it sends a request to the server for more data. If there are a thousand volunteers each having a computer whose speed is 1 megaflop the effective computing speed is ideally 1000 megaflops.

Volunteer computing got a boost when the University of California, Berkeley, developed an open source middleware program called BOINC (Berkeley Open Infrastructure for Network Computing) to support a project called SETI@home.

Volunteer computing got a boost when the University of California, Berkeley, developed an open source middleware program called BOINC (Berkeley Open Infrastructure for Network Computing) to support a project called SETI@home (www.setiathome.

ssl.berkeley.edu). SETI aims to analyze radio signals emanating from outer space for signs of extra terrestrial intelligence. The SETI project team widely publicised the project and requested volunteers to donate idle time on their computers. The response was very good and over 1.5 million computers worldwide have participated in the project during the past 15 years. At present over 150,000 volunteers are participating and the combined processing power is around 650 Teraflops. BOINC has been enhanced over the years and currently supports Windows, MAC OS, Android, Unix Versions and even some gaming computers. This allows a large variety of computers to participate in the project.

Another notable volunteer computing project is Folding@home (https://folding.stanford.edu) which is a protein folding simulation program developed by Pande at Stanford University. It is also amenable to distributed computing using single program multiple data parallel programming paradigm. Around 100,000 volunteers are participating in this project providing 20 Petaflops of computing power. The program to be run on a volunteer's computer is sent by Stanford University along with data. All computers compute independently and send the results to the server at Stanford. Besides these projects, over 50 volunteer computer projects are currently active.

Both grid computing and volunteer computing are distributed computing environments. However, their architecture and usage pattern are quite different as is clear from the above description of volunteer computing. Grid computing is 'give and take' model in which participants give computing resources to members and also take resources from other members. In volunteer computing the initiating institutions 'take' computing power free of cost from donors. Another model is possible (in theory) in which a group of mutually trusting members can each download BOINC and give and take resources from each other which is similar to the philosophy of grid computing.

Both grid computing and volunteer computing are distributed computing environments. However, their architecture and usage pattern are quite different as is clear from the description of volunteer computing.

Conclusions

Both grid computing and cloud computing depend on the availability of high speed Internet that connects massive computing resources. However, they are different computing environments and have their own distinct clientele. As of now cloud computing is widely used with several competing providers of cloud services. This is due to the ease of use of clouds and on demand availability of enormous computing resources. Intense competition among vendors has brought down the cost of cloud computing. Grid computing is not that widely used as it depends on the cooperation of a large number of institutions and those owning the resources tend to favour users belonging to their own institution over 'guests' from other institutions. Grid and cloud computing will coexist and borrow technical ideas from each other.

Acknowledgement

I thank Prof. P C P Bhatt and Dr J Lakshmi for reading this article and their critical comments that helped me improve it.

Suggested Reading

[1] V Rajaraman, Cloud Computing, *Resonance*, Vol.19, No.3, pp.242–258, 2014.

[2] Report of the National Science Foundation Advisory Committee for Cyber Infrastructure Task Force on Grand Challenges, Washington, U.S.A., March 2011. (www.nsf.gov.in/cise/aci/taskforces)

[3] I Foster and C Kesselman, Globus: A Metacomputing Infrastructure Tool kit, *International Journal of Supercomputing Applications*, Vol.11, No.2, pp.115–128, 1997.

[4] F Berman, G Fox and A Hey (Editors), *Grid computing: Making Global Infrastructure a Reality*, Wiley, New York, 2003.

[5] P Mell and T Grance, *The NIST definition of Cloud Computing*, National Institute of Standards and Technology, U.S.A. Special publication 800-145, Sept.2011 (Accessible on the world wide web).

[6] J Foster *et al*, *Cloud Computing and Grid Computing 360 degrees compared*, 2008 (www.arxiv.org/pdf/0901.0731).

[7] V Rajaraman and N Adabala, *Fundamentals of Computers*, 6th edition (Chapter 17 Emerging Computing Environments), PHI Learning, New Delhi, 2015.

[8] B Jacob *et al*, *Introduction to Grid Computing*, IBM Red Books, December 2005.

Address for Correspondence
V Rajaraman
Supercomputer Education and Research Centre
Indian Institute of Science
Bengaluru 560 012, India.
Email: rajaram@serc.iisc.in