GRID - distributed computing power at home

Jan Hlinovsky
Helsinki University of Technology
jan.hlinovsky@hut.fi

Abstract

Grid computing is a concept for large scale sharing of resources. A large potential of computing power is staying unused at homes, and the growing number of broadband Internet connections is making the use of this potential feasible. This paper will shortly examine the architecture of a Grid system and the techniques necessary for it to work. Then it will focus on the Grid concept from the home user's perspective, coming up with scenarios and business models. Last, some thoughts on security requirements that these models suggest will be presented.

1 Introduction

Distributed computing is a subject that has been studied a lot in the past, but recently a new field, Grid computing, has drawn attention back to the area. Grid computing differs from traditional distributed computing in its goals: sharing resources on heterogeneous platforms on a geographically wide area is now coming possible through advances in network technology.

Originally the term Grid referred to a distributed computing infrastructure that was proposed for scientific use [4]. Later the term has grown to include many kinds of advanced distributed computing schemes. The term Grid comes from an analogy to the electric grid — you can just plug in when you need power.

The object of this paper is to consider Grid techniques from the home user's perspective. First this paper will examine the basic building blocks of a Grid architecture. Special attention will be given to the Globus Toolkit, which is the most widely used implementation of a Grid infrastructure. This technical Grid walk-through will be in chapter 2. Next, in chapter 3, different scenarios in which home users participate in virtual organizations will be considered. The emphasis will be on sharing computational resources. The chapter will then continue with a few business models, modified from the scenarios. Chapter 4 will discuss some security requirements that rise from the scenarios and business models. Finally, chapter 5 will present conclusions.

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

2 Grid architecture

The ideology of Grid architecture does not strictly follow neither the traditional client-server paradigm nor the more modern peer-to-peer approach. The basic idea is sharing resources on an equal ground, but some centralized services are needed for this to happen in practise. The architectural design can be thought to consist of the communication protocols on one hand, and the interfaces to actual applications (the API's) on the other. Together these form what is usually called "middle-ware", i.e. services that are needed in order to run applications over a network.

The so called "Grid problem" is how to share and use resources in a dynamic, scalable and multi-institutional virtual organizations in a controlled and coordinated way [4]. A virtual organization (VO) is a set of participants that have agreed on conditions and rules for sharing resources [4]. A virtual organization might comprise of a single company's local network, but it might as well be the customers of an ISP or just a group of individuals from around the world willing to take part in a cyber-community.

2.1 Grid protocol architecture

Requirements for an operational Grid architecture include:

- For resource sharing:
  - finding available resource
  - agreeing on service
  - transferring data, program code etc.
  - using resources in a controlled way
  - error recovery
  - security
  - portability for different platforms
  - possible compensation for resource usage
- For a virtual organization
  - information of members (name, status, configuration, available services ...)
  - joining and leaving a VO, creating a VO
  - scheduling, load management, control of resources and policies
  - scalability

There have been several candidates [1] for protocols to fulfil these needs, but the most widely adopted solution is the Globus Toolkit [3]. Globus is an open source software toolkit that implements basic building blocks for a Grid infrastructure, including central services and protocols. It is based on existing Internet protocols. Other notable projects for building a Grid infrastructure are Legion [7] and JXTA [8]. JXTA is a set of general peer-to-peer communication and resource sharing protocols that has potential of becoming a popular Grid framework in the future.

2.2 Globus Toolkit [3, 4, 5]

The Globus Toolkit is a large and constantly developing project, but the central parts are:

- the Grid Security Infrastructure (GSI)
- the Grid Resource Allocation and Management (GRAM) protocol and service
- the Meta Directory Service (MDS)

The Globus is built on existing TCP/IP protocols, it takes advantage of for instance Internet's routing and domain name service systems. Some services are modified, there is, for example, a GridFTP protocol for efficient data transport that uses the Grid Security Infrastructure (making it more secure than the traditional FTP) and has some other additional features.

2.2.1 Layer model [4]

The architects of Globus Toolkit present their view of Grid architecture as a layer model. The layers in the model are:

- *Fabric:* The fabric layer includes the local resources, such as computers, databases, networks and their management mechanisms, operating system services etc. These are basically taken as existing infrastructure.
- *Connectivity:* This layer handles communication and authentication. The Globus toolkit uses available TCP/IP protocols and implements the Grid Security Infrastructure (GSI).
- *Resource:* The resource layer uses the services of the connectivity layer to handle sharing of an individual resource. It also communicates directly with the fabric layer to access the local resource. The Grid Resource Allocation and Management (GRAM) is the central part of Globus on this layer.
- *Collective:* As the layer between resource and application layers the collective layer deals with control and interaction between multiple resources as well as with allocation of resources according to the needs of applications. Services that belong to this layer include resource directory services, scheduling and allocation, monitoring, and accounting. The Globus Toolkit implements the Meta Directory Service (MDS), and there are several individual projects working on the other services, such as Nimrod/G [2] that implements scheduling and allocation.
- *Application:* The applications that the users run in virtual organizations are located in the highest layer.

The layers in this model do not map easily to the TCP/IP model, although [4] presents a such mapping. The application layer of TCP/IP model includes services from all layers of Grid layer model, such as DNS (connectivity layer) and SNMP (fabric layer). The Grid layer model should be seen as a description of logical relationships, not as a description of the implementation of the protocol stack.

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

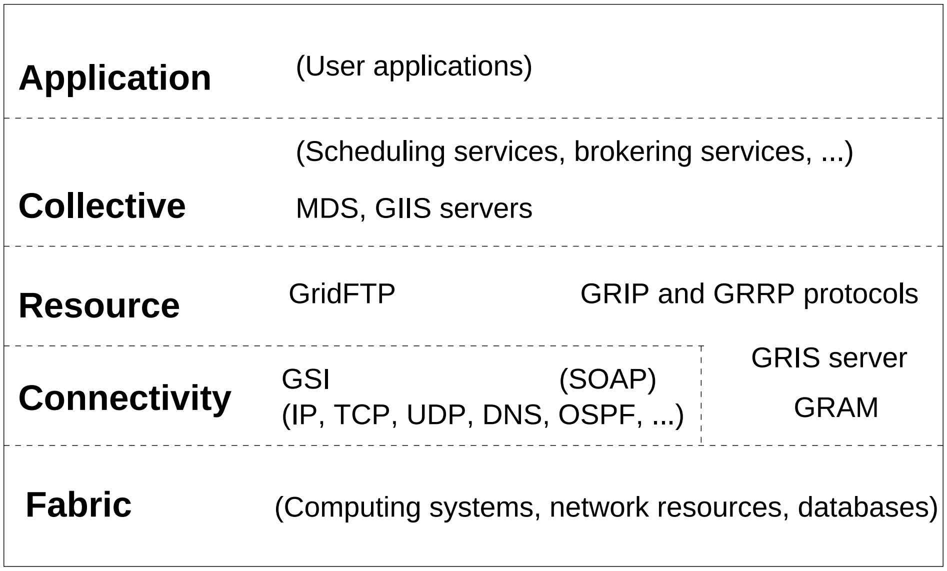

Figure 1: The layered approach to Grid architecture. The items in parentheses are not part of the Globus Toolkit.

2.2.2 Grid Security Infrastructure

The Grid Security Infrastructure offers single-sign-on authentication and some support to delegation of authority. It also provides message confidentiality and integrity. The GSI requires an operational Public Key Infrastructure and it uses X.509 certificates. The authentication protocol is not fixed, but usually the GSI uses TLS (Transport Layer Security) and adds single-sign-on and delegation functionality to that.

2.2.3 Grid Resource Allocation and Management

The Grid Resource Allocation and Management protocol (GRAM, also called Grid Resource Access and Management protocol in [4]) is used for remotely allocating resources and for controlling and monitoring them. The protocol uses the Resource Specification Language (RSL) for communications. RSL uses attribute-value pairs, the syntax is similar to LDAP (for example &(executable="/bin/ps")(arguments="-a")).

The GRAM service includes a "gatekeeper" unit that is the point of entry to the system. This is where the authentication is made, connections are established, and processes started and controlled.

2.2.4 Meta Directory Service

The Meta Directory Service (MDS) is the answer to the problem of keeping track of available resources. It uses two protocols, Grid Resource Registration Protocol (GRRP) for discovering resources and Grid Resource Inquiry Protocol (GRIP) for communicating information about resources. The GRIP is based on LDAP protocol, and, actually, the LDAP

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

service is the basis for the whole MDS service.

There are two kinds of MDS servers: Grid Resource Information Service (GRIS) servers on the lowest level (specific resources level) and Grid Index Information Service (GIIS) servers on higher levels. The GIIS servers collect data from GRIS servers beneath them using LDAP as the inquiry protocol. There is a global naming scheme which is, again, adapted from LDAP.

2.3 Service oriented approach

The next step upwards is abstracting everything behind the concept of *Grid service*. The Open Grid Services Architecture (OGSA) [5] is the result of taking the Globus Toolkit and covering it with an abstraction layer using Web Services Description Language (WSDL). Storage and computing resources, networks, devices, programs etc. are all understood as services. (The authors of OGSA chose to call them *services* instead of *objects* "due to the overloaded meaning" of the word object.[5] Well, the word service is not exactly free from overload either . . .)

In comparison to other architectures with similar functionality, such as CORBA, OGSA is simpler and, of course, tailored for Grid services. Many parts of the OGSA are built on XML (Extensible Markup Language), such as the Web Services Description Language (WSDL), Web Services Inspection Language (WSIL), Web Services Flow Language (WSFL) etc. The protocol for transfering all this information on the services is Simple Object Access Protocol (SOAP) which can be carried over several protocols including HTTP and FTP [5].

The obvious merit of the service approach is that it defines a uniform interface to services, which include not only the building blocks created in the Globus project so far, but also higher level services that are currently being built or will be built in the future. Examples of these higher level services are e.g. distributed data management services, workflow management and auditing services [5]. From these the service providers will eventually implement even higher level services that they offer to the customers.

2.4 The home user

So far, the Grids found on paper and in test laboratories have mainly been scientific research projects focusing on scientific applications (e.g. Earth System Grid[10], Grid Physics Network[11] etc.). Now the field has got plenty of attention from the business world, and commercial applications can be expected.

The potential computing power sleeping at home users' desks has been recognized for some time now (c.f. Seti@Home and others), and before long we can expect "Home Grid" program suites that implement the necessary parts of the infrastructure, such as a GRIS server and GRAM and gatekeeper services, so that home users will be able to join virtual organizations easily.

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

3 Scenarios and business models

Computing problems can be divided into parallelizable (such as searching for a pattern in a large set of data) and non-parallelizable (such as counting from 1 to 10000). The parallelizable problems can further be divided into those that need a lot of communication with little delay and those that do not need to communicate much and do not care about latency. The last kind of problems are those that are suitable for Grids, the others are still most effectively handled with sequential or parallel supercomputers.

So the set of problems that benefit from shared computing capacity in a Grid is limited, and, in addition to this, these problems tend to be of such kind that the average home user has no need to compute them. The natural basis for a business model would then be such that home users joining a Grid will offer computing capacity in exchange for storage capacity or some other service, perhaps even a micropayment.

3.1 Scenarios

In order to get an idea of possible roles and interactions in Grid systems including home users, some scenarios will be considered.

- Virtual community

  Computer users with a common interest (hobby, school, residential area, . . .) can create a community and start to deploy Grid technologies. The resource sharing inside the community would probably be mainly storage sharing, but in some cases the community might find use for the computing capacity as well. An option of selling computing capacity to some party outside the community would also exist.

  Example scenarios:
  Example 1: A group of crypto enthusiasts decide to create a Grid that enables them to use the combined computing capacity of their computers to test the strength of their self-made crypto-algorithms. They also share backups of their non-confidential data in the Grid.

  Example 2: A movie fan club forms a Grid that sells the computing capacity to outside companies and uses the profit to buy films in digital form. The films are stored using the shared storing capacity. (Now this example is highly hypothetical and serves only to illustrate the possible uses of Grid technology - the film producers would have to change their attitudes towards intellectual rights for this kind of sharing to be actually possible.)

- Charity

  Enlightened computer users can choose to donate their spare computing capacity to a scientific or humanitarian purpose, much in the way of Seti@Home and Compute-against-Cancer today.

  Example scenario: Home user joins the Largest-Prime Grid and sets the access control to allow starting a process that helps the Virtual Organization to make a new record of largest known prime number. The process will be suspended when there is user activity on the computer. The next day our user finds another respectable way

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

to donate cycles and joins the Dolphin Grid to help decrypting dolphin language. He allows this VO to start a process, too, and sets the local scheduler to allow 60% of the "idle" time to the dolphin process and 40% to the prime number process.

The charity VO can be seen as a kind of community, but the organization is centralized and the "cycle donators" do typically not get any service in return of the usage of their resource. Also the membership times would be shorter, members coming and going all the time, maybe coming again after a while.

- Mutual gain

  Example scenario: An ISP offers a broadband network connection to home users. The home users, in return, must install a Grid suite that is remotely configurable by the ISP and they also agree to allow the ISP to execute programs on their host. They make a resource usage agreement with conditions like that the ISP is allowed to control the computing resources for at least n hours every weekday, at some fixed hours, or for at least 2n hours per day in average, whenever the resource is free. Sanctions for not following this resource usage agreement are also agreed upon. The ISP then sells the computing capacity onwards to its regular capacity buying customers according to contracts and auctions the remaining computing capacity.

  Example 2: A user joins a Search-Grid VO and allows a search engine use the idle computing time and additionally to access a small part of the local disc to store cached documents. The user gets free access to the search service which includes dictionaries and other specialized databases. Users that do not belong to the VO have to pay for the service.

- Private Grid

  A company that has the employees' home computers in a VPN could establish a Grid inside the VPN and utilize all the spare processing power both at the company's premises and the employees' homes.

3.2 Business models

The scenarios above give rise to some business models [9]. The purpose of this is to generalize the roles that appear in the scenarios and consider how the parties benefit from the transactions. Here business is understood as any commercial activity, not necessarily directly involving money.

3.2.1 Roles

The concept of resource is taken as the basis for defining roles. The typical roles are:

- *Resource provider*: in our examples of computational resources the resource provider is the home user. The home user owns the computational resources, but in average case the resources are idle most of the time. The goal of the home user is thus to get some benefit from making the resource available to outsiders.

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

- *Resource consumer*: in our examples the consumer of computational resources is some organization or company that has a great need of computational capacity. The consumer typically has some resources by itself, but if the demand is too great for the existing resources there are two options: buy more resources or outsource them. The downside of buying computing arsenal is that it gets old quite fast and supercomputers are never cheap. If the demand for computing resources varies a lot or it is difficult to estimate future needs, then the option of outsourcing computing resources is especially suitable in its flexibility.

- *Resource broker* is basically the Grid operator that registers providers of computational resources and allocates tasks to them. The tasks come from resource consumers. Resource broker naturally takes its share of profit from the transactions that go through it. The resource broker functionality can be handled also by the resource consumer or provider.

3.2.2 Broker models

- Cycle shop:

  The cycle shop makes contracts with resource providers (including home users) concerning the usage of their resources. The compensation for the providers is either some sort of micropayment or e.g. gift certificates to other online shops.

  The cycle shop offers reliable computing service to the consumers. The reliability comes from the large amount of resource providers. There are different levels of service quality; the cheapest service is the one that suffers first if there are problems with the providers' ability to offer agreed resources.

- Auction broker:

  The auction broker is similar to the cycle shop, it aggregates capacity from multiple providers and sells it to the consumers. The difference is that the the price is decided by auction. The agreements between parties are typically shorter termed compared to cycle shop.

3.2.3 Resource trading models

- Symbiosis model:

  This is a generalization of the first example scenario in mutual gain scenarios. The consumer of computing resources offers the providers of that resource some other resource in exchange. This can also be combined with the brokerage idea, if the resource provided by one side does not match the resource needed by the other and vice versa. Typical example is an ISP providing network access, web hosting, portal services etc. and acting as a cycle shop. The central point of the symbiosis model is that the relationship between the provider and consumer (or broker) is static. The benefit to the consumer is that the level of service is predictable. The symbiosis model is most likely to be applied on a geographically restricted area, not only because of its suitability to ISPs but also because it is the most static and binding of the models presented here, which means that it would benefit from operating under a uniform legislative area.

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

- Affiliation model:

  A home user joins some virtual organization that provides a resource-hungry service, such as a search engine. Sharing some computational and/or storage resources allows the user free access to the service provided by the virtual organization. This model is similar to the symbiosis model, but the amount of resources being traded is smaller, which affects the properties of the model. The relationships are more dynamic and it is easier to be affiliated to several services. The home user gets free access to a service he normally would have to pay for and the service can use the resource to enable faster operation and get more paying customers.

3.3 Comparison of models

All the models presented are applicable to home users. One of the biggest differences between the models is the duration of contracts and memberships in the virtual organizations.

The main features of the models are compared in the following table

Model	Broker?	Compensation	Membership time
Cycle shop	yes	micropayment	medium
Auction	yes	micropayment	short
Symbiosis	possibly	service	very long
Affiliation	no	service	long

4 Security requirements

The topic of security in Grid systems as a whole is large and complex enough to be outside the scope of this study. This paper will only consider a few requirements that rise up from the models above and examine the requirements from the perspective of different actors in the models.

4.1 Computing capacity provider (Home user)

The central requirement, from the home user's point of view, is access control, i.e. that no resources except the explicitly specified ones are shared. The authentication of network connections is provided by the Globus toolkit, but in addition to this the Home Grid suite installed in the home user's computer should have a configurable sandbox-like protection system.

4.2 Computing capacity consumer

If the consumer is going to outsource the computing resources he can not have high requirements for confidentiality. (If confidentiality is needed then only own resources should be used.) The main concerns are then dependability on the outsourced resources, especially availability (does the service provider really deliver what it promised), and on the other hand integrity of the results. Attempts to cheat can, in many cases, be noticed in the results, but in some cases this is difficult. In some simulations, for instance, the real results can be indistinguishable from some random fake results. (If one knows the answer, what's the point of running the simulation?) The involvement of money will make the threat of fraud attempts real. Simplest solution is to add some redundancy: duplicate part of the computations in independent places and compare results.

4.3 Broker

The main requirement on the broker's side is accountability. The broker must keep track of all the providers and consumers and their actions in the virtual organization. Part of this comes from authentication that is included in the basic protocols, but also a lot of logging and auditing needs to be done.

5 Conclusions

Grid computing, which originally started as a project for scientific computing, is now getting adopted to commercial world and will eventually be introduced on home users' desktops. The computational capacity of home computers that is now sitting idle most of the time will then be shared and exchanged to other services or digital money.

This paper has shortly covered the central parts of Grid architecture, and in particular Globus, a widely used implementation of Grid middle-ware. The Grid concept has been examined from the home user's perspective, and scenarios of virtual organizations that home users could join have been made up. A few scenarios have been remodified as business models and compared to each other. Finally, there are some notes about security requirements suggested by these models.

The main outcome of this paper is the discussion of Grid techniques' applicability to home users in form of scenarios and a few business models. The scenarios suggested here do not cover all possibilities, but the author believes them to be realistic and representative. The presented business models are quite basic, and one direction of future studies could be coming up with more models and analysing them, together with the ones presented here, on a deeper level.

References

[1] Baker, M. & Buyya, R. & Laforenza, D., Grids and Grid Technologies for Wide-Area Distributed Computing. <http://www.csse.monash.edu.au/~rajkumar/papers/gridtech.pdf> (referred 4.2.2002)

[2] Buyya, R. & Abramson, D. & Giddy, J., Nimrod/G: An Architecture of a Resource Management and Scheduling System in a Global Computational Grid. HPC Asia 2000. <http://www.csse.monash.edu.au/~davida/papers/hpcasia.ps.Z> (referred 17.4.2002)

HUT TML Course T-110.551 ISBN 951-22-5959-1 ISSN 1455-9749
http://www.tml.hut.fi/Studies/T-110.551/2002/papers/May/index.html

[3] Foster, I. & Kesselman, C. Globus: A Metacomputing Infrastructure Toolkit. International Journal of Supercomputer Applications, 11(2):115-128, 1997.

[4] Foster, I. & Kesselman, C. & Tuecke, S., The anatomy of the grid: Enabling scalable virtual organizations. International Journal of High Performance Computing Applications, 15 (3). 200-222. 2001. <http://www.globus.org/research/papers/anatomy.pdf> (referred 4.2.2002)

[5] Foster, I. & Kesselman, C. & Nick, J. M. & Tuecke, S., The physiology of the grid: An Open Grid Services Architecture for Distributed Systems Integration. *Draft version.* <http://www.globus.org/research/papers/ogsa.pdf> (referred 2.4.2002)

[6] Foster, I. & Kesselman, C. & Tsudik, G. & Tuecke, S., A security architecture for computational grids. In Fifth ACM Conference on Computers and Communications Security, November 1998.

[7] Natrajan, A. & Humphrey, M.A. & Grimshaw, A.S. Grids: Harnessing Geographically-Separated Resources in a Multi-Organisational Context. Presented at High Performance Computing Systems, June 2001. <http://legion.virginia.edu/papers/HPCS01.pdf> (referred 15.4.2002)

[8] Project JXTA. <http://www.jxta.org/> (referred 16.4.2002)

[9] Rappa, Michael. Business Models on the Web. <http://digitalenterprise.org/models/models.html> (referred 2.4.2002)

[10] The Earth System Grid <http://www.earthsystemgrid.org/> (referred 17.4.2002)

[11] The Grid Physics Network <http://www.griphyn.org/> (referred 17.4.2002)