SLA Management and Resource Modelling for Grid Computing

Adrian Li Mow Ching, Dr Lionel Sacks † and Paul McKee‡

† Electrical and Electronic Engineering, University College London, ‡ BTexact Technologies

a.ching@ee.ucl.ac.uk

Abstract: Current implementations of grids exist in the research and academic communities, where applications are generally characterised as being computationally intensive usually involving large amounts of data. For these purposes and in this environment a 'best effort' resource guarantee is sufficient. However, as grid uses mature in both the academic and commercial arenas, grid providers will require some form of service level management to address these issues. This paper presents a service level approach to grid resource management for SLAs, focussing on the management of SLAs and modelling of resources required by these processes.

1 Introduction

Grid computing enables collaboration and sharing of resources. Grid resources are geographically distributed, heterogeneous and subject to local administration. A grid consists of many machines, spanning across single machines, local area networks and wide areas networks, with the goal of providing one unified resource.

Current implementations of grids exist predominantly in the research and academic communities [1], where applications are generally characterised as being computationally intensive usually involving large amounts of data. For these purposes and in this environment a 'best effort' guarantee is sufficient. However, the shift in grids towards businesses and commercial services creates a new set of demands that have a stricter set of requirements. A commercial service requires an agreement that specifies what the user receives from the offered resources and the relevant performance guarantees. This will be governed and defined in the form of a Service Level Agreements (SLA). An important part of SLA management is how to manage a SLA in relation to the underlying grid resources. Most providers want to have high utilisation in order to maximise revenue, but achieving this is not straight forward because of the volatile nature of grid applications and the current lack of descriptive understandings of grid resources. Also the way in which grid resources are expressed in SLA terms (jobs, applications) are very different to low level raw resource descriptions (processes, memory, processors) required by resource managers.

Whilst many grid providers [2] [3] [4] provide commercial solutions, the area of SLA resource management has not been fully addressed. This is one of the many barriers preventing commercial inter-enterprisegrids. This paper presents a work in progress that takes a service level approach to SLA fulfilment and assurance management in a commercial grid computing environment. The rest of the paper is structured as follows: Section 2 describes the role of SLAs within grid computing, section 3 then outlines other work related to SLA management and resource modelling. Section 4 gives an overview of the SOGRM project. Section 5 introduces the SLA management process and the resource modelling, with a conclusion in section 6.

2. Service Level Agreements for a commercial Grid environment

Current usage and implementations of the grid are based on best effort computing, which is sufficient for the demands of today's uses. An application is run on a grid with no guarantee of successful completion, performance and quality of execution, or any prior indication of the time taken to complete the task. As grid services mature in both the academic and commercial arenas, grid providers will require some form of service level management to address these issues. This will exist between users and providers in the form of Service Level Agreements (SLA). A SLA is an agreement between two parties that defines the guarantees relating to a service that a provider will supply to a user. In a grid SLA the following is defined:

- Application - What the user wants to run
- Resource requirements - The required capability of the resource
- Completion time - Some form of indication when the task should be finished
- Quality of service - The level of guarantee that a task will be completed by the indicated time (i.e. Gold = 95%, Silver = 70% and Bronze = no guarantee)

Although there are many other attributes within an SLA, we have highlighted these as the relevant ones for the management issues with which we are concerned. From a service management and a business perspective, the fulfilment and assurance of SLAs is key. A provider needs to know if it can accept a new SLA based on its resource commitments, whilst at the same time wanting to achieve optimal resource utilisation to gain the maximum revenue. It will also need to have assurances that the service it supplies falls within the bounds defined in the SLA, the resources provided must be capable of completing the task within the time specified.

3. Related work

Various aspects of SLA management within grid computing have been addressed. The specification and monitoring of commercial SLAs was approached in [6]. A language was described for unambiguous and precise specification of SLAs, and a monitoring engine that aggregates measurements from multiple sites. The SNAP protocol [7] was developed for negotiating SLAs and coordinating resource management in distributed systems. It presents a model for managing the process of negotiating access to and the use of resources through the definition of a framework within which reservation, acquisition and task submission can be expressed for any resource in a uniform fashion. The authors of [8] developed a prototype of an ontology-based resource matchmaker, which exploits existing semantic web technologies such as RDF and RDF schema to build a rule-based matchmaker for the requesting of resources.

4. SOGRM

This work is related to the EPSRC funded Self Organised Grid Resource Management (SOGRM) project. The aim is to develop techniques for distributed resource management for GRID as a contribution to the development of a network-wide service architecture [9]. The various resource management techniques are based on the architecture shown in Figure 1.

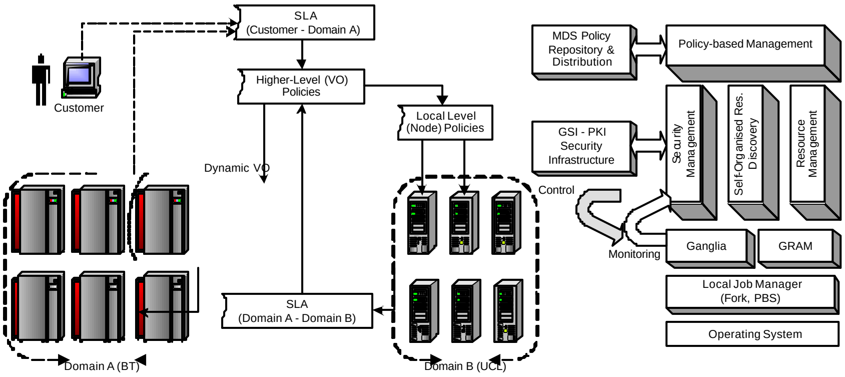

Figure 1 - Overall architecture (a) and the Node architecture (b)

Our scenario includes two grid resource providers and a grid user. The grid customer has an agreement with grid provider A to use its resources. Grid providers A and B also have an agreement to use each other's resources. In a scientific environment this might be for collaboration, in a commercial environment this might be during instances for overflow of resources. These agreements, defined as SLAs, require a combination of service level management at the top level and resource management at the lower levels. At the node level, resources are managed by; the local monitoring and resource modelling which captures resource performance and utilisation, Security Management which uses I^3 (Integrity, Information and Intelligence) to perform Intrusion detection, and Self Organised Resource Discovery which uses a self-organised algorithm to find the optimal available resources for jobs entering the system.

5a. SLA management

In order to understand the characteristics of a grid in relation to SLA attributes, we considered resources in terms of resources allocated or committed to users. It is difficult to model what resources grid users require, as a fuzzy area separates requirements in humanly understandable terms with those requirements in to machine understandable terms for resource management. The requirements defined in an SLA by a user for a job (completion time, application, quality of service) are humanly abstract and do not have a direct mapping to the low level resource characteristics required by grid management systems (number of processes, cpu capability).

Our SLA management process and its role in relation to the other parts of SOGRM resource management is illustrated below in Figure 2.

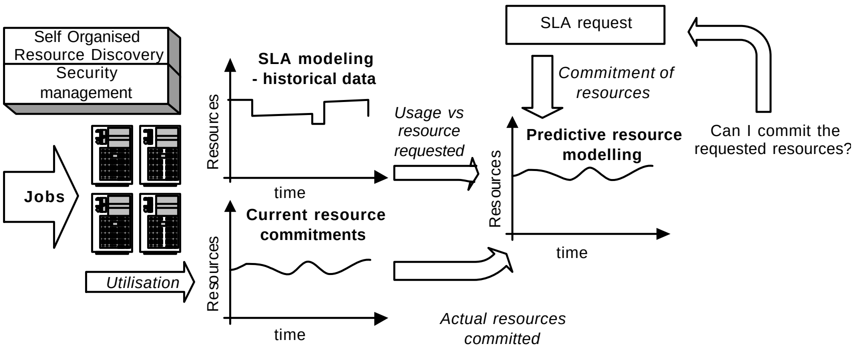
Self Organised Resource Discovery
Security management
Jobs
Utilisation
SLA modeling - historical data
Resources
time
Current resource commitments
Resources
time
Usage vs resource requested
Actual resources committed
SLA request
Commitment of resources
Predictive resource modelling
Resources
time
Can I commit the requested resources?

Figure 2 - SLA management process

When an SLA enters the system, each SLA request is a commitment of resources. At the same time, grid resources which are already loaded with running jobs have a certain level of resource utilisation and resource commitments. By modelling historical SLA data we can gain information regarding the arrivals of incoming resource commitments and the possible patterns of future resource requests over a certain time period. We can also model the amount of resources used in comparison to the amount of resources committed to a user, which indicate when resources allocated to particular users are actually used. Although a user might have a high QoS, and a large amount of resources are initially committed to this user, all these resources are likely to be unutilised for certain periods of time. This information, combined with current usage information can be used to predict a probable resource usage. Furthermore, it is then also possible to predict the amount of spare resources over a future time period based on the difference between our maximum resource commitments and the actual amount of resource we predict will be required. For example, a researcher has a computationally intensive application with a given resource requirements and wants the application to finish within a set time period. The researcher also requests for a high QoS, and so is provisioned a large set of resources. However since our researcher performs the bulk of the intense calculations over night, we can commit all of the provisioned resources over night and say, only half of these resources during the day as we know they will not be used, allowing us to commit those resources elsewhere. Although it might appear that we are over committing resources, our modelling of historical usage tells us that the resources we commit is higher than the resources used and the actual usage is within the bounds of our resource capabilities.

5b. Resource modelling

Fundamental to our approach to SLA management is modelling resources in terms of resource commitments. In order to do so we decomposed an SLA in to its low level resource attributes. At this point we argue that more common values for resources such as processor speed or memory size do not give accurate indications as to how well a particular application might execute on a set of resource. To gain a better understanding requires resources to be first of all quantifiably expressed to the degree where their values represent their performance in terms of application requirements, and secondly, expressed in terms of the resources required by different sets of applications. The decomposed SLA and resource relationships we defined are shown below in Figure 3.

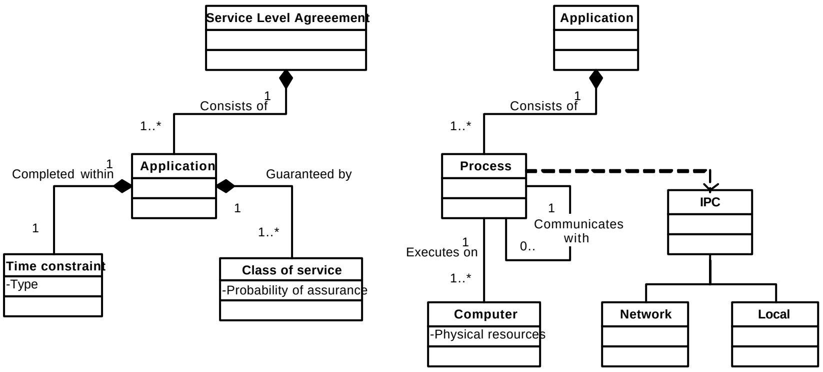

Figure 3 - SLA and resource decomposition models

An SLA is composed of one or more applications, each application has a time constraint and a requested QoS guarantee. This time constraint can be, finish as soon as possible, finish by time t, and finish anytime. The QoS guarantee is an assurance that the application will finish execution by the time specified (there is no guarantee required for an application to finish anytime). In terms of physical requirements, we state that an application is composed of one or more processes, each of which is executed on a set of resources on a computer.

For a given application we can populate its characteristics in to the different attributes specified in our model. Using these defined relationships and assigning values to each part, we intend to investigate the use of logic and probabilistic approaches to allocate loosely defined rules to the relationships. Thus, to a certain level of accuracy we can use the model to express the resource requirements of an application and model grid resources. The requirements of a processes and capability of the resources give an indication of the amount of time needed for an application to finish.

6. Conclusion

This paper has discussed a service level approach to SLA resource management. Decomposing an SLA in to resource commitments, we created a resource model that defines relationships between applications and grid resources in terms of requirements. Furthermore, it is quantifiable in terms of resources committed, and so will allow us to model resources, perform calculations and perform the required decision making of fulfilling incoming SLA requests and factoring in the assurance of these requests.

Acknowledgments.

Adrian Li Mow Ching would like to acknowledge the EPSRC for funding his EngD programme.

References.

[1] The UK e-science grid. http://www.escience-grid.org.uk/
[2] Platform Computing. http://www.platform.org
[3] United devices. www.ud.com
[4] Entropia www.entropia.com
[5] The Global Grid Forum, http://www.gridforum.org/
[6] Sahai A, Graupner S, Machiraju V, van Moorsel A. Specifying and Monitoring Commercial Grids through SLA. CCGrid, May 2003.
[7] Czajkowski, K., Foster, I., Kesselman, C., Sander, V., Tuecke, S.: SNAP: A Protocol for Negotiation of Service Level Agreements and Coordinated Resource Management in Distributed Systems, submission to Job Scheduling Strategies for Parallel Processing Conference (JSSPP), April 2002.
[8] Hongsuda Tangmunarunkit, Stefan Decker, Carl Kesselman: Ontology-based Resource Matching in the Grid - The Grid meets the Semantic Web. 1st Workshop on Semantics in Peer-to-Peer and Grid Computing, May 2003.
[9] I. Liabotis, O. Prnjat, T. Olukemi, A. L. M. Ching, A. Lazarevic, L. Sacks, M. Fisher, P. McKee, "Self-organising management of Grid environments", International Symposium on Telecommunications (IST'2003), Isfahan, Iran, 2003.