Towards Quality of Service in the Cloud

Django Armstrong, Karim Djemame*

Abstract

Quality of Service (QoS) plays a critical role in the affective reservation of resources within service oriented distributed systems and has been widely investigated in the now well established paradigm of Grid Computing. The emergence of a new paradigm, Cloud Computing, continues the natural evolution of Distributed Systems to cater for changes in application domains and system requirements. Virtualisation of resources, a key technology underlying Cloud Computing, sets forth new challenges to be investigated within QoS and presents opportunities to apply the knowledge and lessons learnt from Grid Computing.

QoS has been an issue in many of the Distributed Computing paradigms, such as Grid Computing and High Performance Computing. The aim of this paper is to address QoS specifically in the context of the nascent paradigm Cloud Computing and propose relevant research questions. The objectives of this paper are to discuss the confusion surrounding the term "Cloud", the current consensus of what Cloud Computing is and the legacy bequest by Grid Computing to this emergent paradigm. Emphasis is placed on the state of QoS provisioning in Grids and the technology to enable it in Cloud Computing. Finally open research questions within QoS relevant to Cloud Computing are proposed and the direction of various future research is envisioned.

Keywords: Quality of Service, Cloud Computing, Grid Computing, Resource Management, Virtualisation

1 Introduction

Quality of Service (QoS) is a broad topic in Distributed Systems and is most often referred to as the resource reservation control mechanisms in place to guarantee a certain level of performance and availability of a service. The scope of this paper is primarily concerned with the management and performance of resources such as processors memory, storage and networks in Cloud Computing. A defined QoS is not just limited to guarantees of performance and availability and can cover other aspects of service quality, which are outside the scope of this paper, such as security and dependability. The problems surrounding resource reservation are non-trivial for all but the most basic best effort guarantees and the problems behind resource capacity planning are non-deterministic polynomial-time hard to solve.

QoS provides a level of assurance that the resource requirements of an application are strictly supported. QoS models are associated with End-Users and

*School of Computing, University of Leeds, United Kingdom, {d.j.armstrong04,k.djemame}@leeds.ac.uk

Providers (and often Brokers), involve resource capacity planning via the use of schedulers and load balancers and utilise Service Level Agreements (SLA). SLAs provide a facility to agree upon QoS between an End-User and Provider and define End-User resource requirements and Provider guarantees, thus assuring an End-User that they are receiving the services they have payed for.

Section 2 of this paper will demist the term "Cloud", discuss the heritage of Grids in Cloud Computing, and define the paradigms. Section 3 of the paper will explore the history and current state of QoS within Grid Computing, relevant to the scope of this paper. Section 4 will discuss commercial Cloud adopters, describe Cloud Computing research projects and open source solutions and Virtualisation technology in the context of QoS. Finally Section 5 will elaborate on possible trends in QoS research and propose some open research questions formulated from the material discussed in the paper congruous to Cloud Computing.

2 Clouds Vs Grids

Cloud Computing has been described as:

> "the next natural step in the evolution of on demand information technology services and products"[67]

within the field of Distributed Systems and draws heavily on the principles and paradigms of Grid and Utility Computing. As with any service, such as public utilities, guarantees need to be in place that pledge a certain level of performance and involves resources reservation control and monitoring mechanisms for service fulfilment. There has been much confusion over the term Cloud Computing due to its relative infancy within computer science, its extensive generalised use by industry and the lack of consensus on what a Cloud really is. Many definitions have been proposed and are often muddled up with the Grid paradigm.

Before the relevance of QoS within Cloud Computing can be considered, concrete definitions are essential in being able to characterize current Cloud Systems. This will facilitate in reducing the scope of research questions by excluding more generalised definitions of Clouds made by computing experts such as:

> "using the Internet to allow people to access technology-enabled services."[43]

Being able to categorise a cloud by its capabilities is key to formulating a concise definition. A general consensus is held that Clouds fall into at least one of three types of system, dependent on the actors involved and the services they provide[41, 66]. The definitions of these three types of system are:

- Software as a Service (SaaS), defined as a provider that supplies remotely run software packages, via the Internet to consumers, on a utility based pricing model. A typical example application could be an on-line alternative to a word processor or spread sheet.

- Platform as a Service (PaaS), defined as a provider that offers an additional layer of abstraction above the virtualised infrastructure. This provides a

software platform that trades off restrictions in the type of software than can be deployed in exchange for built-in scalability.

- Infrastructure as a Service (IaaS), defined as a provider that provisions compute and storage resource capacity via virtualisation allowing physical resources to be assigned and split dynamically.

These three categories of system are tiered from the bottom up, meaning that for a PaaS provider to function the use of an IaaS provider would be mandatory or alternatively the PaaS provider could deploy and utilise their own IaaS. Previous tiers are obscured from the End-User and services are provided transparently. This allows for increased flexibility, the possibly of an open market and reductions in cost.

The evolution of Cloud computing has its roots in multiple Internet related distributed system technologies and computing paradigms such as Cluster Computing, P2P, Service Computing, Utility Computing and most importantly Grid Computing.

QoS within Grids has been a major topic of interest and continues to be actively researched[37, 36, 51]. This paper is reminiscent to the position that resource management and performance research was at in the early days of Grid Research[59, 55, 50], when the issues were just becoming understood. Grid Computing has been defined as:

> "a system that: coordinates resources that are not subject to centralized control using standard, open, general-purpose protocols and interfaces to deliver nontrivial qualities of service"[42]

and was hailed as the next revolution in computing science after the creation of the Internet and shares many of the same goals as Cloud Computing. Thus the majority of the lessons already learnt within the research topic are highly relevant to Cloud computing. The motivation behind research into Grid Computing was initially the need to manage large scale resource intensive scientific applications across multiple administrative domains that require many more resources than that can be provided by a single computer. Cloud computing shares this motivation but within a new context oriented towards business rather than academic resource management, for the stipulation of reliable services rather than batch oriented scientific applications. This difference in application domain and requirements being pushed by industry does not mean that the scientific community cannot leverage Cloud Computing, far from it, as illustrated by GridBatch[52]. There is much crossover between the two paradigms and many goals are shared.

Cloud Computing will be enabled through the next generation of data centre technology. The current generation of data centres are already leaning heavily towards the virtualisation of compute and storage resources, the technological foundation of a Cloud, enabling the consolidation of proprietary servers running legacy software. This is being achieved through the creation of virtual machines which run on large physical servers utilising the latest technology. This provides the benefits of being able to both reduce maintenance cost and minimise lost revenue due to downtime and also takes advantage of the improvements in computer efficiency facilitated by hardware vendors such as Intel and AMD.

Using the previously defined scope, the definition of a Cloud most relevant and appropriate to the research topic of QoS is:

> "Clouds are a large pool of easily usable and accessible virtualised resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to an optimum resource utilisation. This pool of resources is typically exploited by a pay-per-user model in which guarantees are offered by the Infrastructure Provider by means of customized SLA's."[66]

The definition refers to a pay-per-user economic model taken from the paradigm of Utility Computing. Utility computing is a:

> "service provisioning model, which provides adaptive, flexible and simple access to computing resources, enabling a pay-per-use model for computing similar to traditional utilities such as water or electricity."[53]

Research has already been carried out on the commercial benefits of Utility Computing within the Grid Economy[40, 33] and thus it is easy to envisage why such an economic model is important in Clouds and is being exploited within Cloud Computing, which is heavily oriented towards business applications and where revenue is a primary concern.

3 Quality of Service in Grids

By exploring the current state of QoS in Grid Computing, the lessons already learnt can be exploited and potentially utilised in Clouds. In the early 21st Century the dynamics of the Internet economy changed and the ratification of e-commerce as a new source of revenue growth within businesses increased. This led to the development of the Web Service, as businesses turned to Service Oriented Architectures to simplify their interactions in the digital world, through the loose coupling of the service providers and consumers.

The introduction of Web Services affected the development of Grid Computing as emphasis was placed on Grids providing services to reduce the complexity and cost that had been previously associated with them. The Service-Oriented economy also provided the mechanism to create virtual organisations where computation resources could be shared securely. Service oriented Grids created new problems concerning the management and availability of shared resources across organisational boundaries. Grids relied for many years on the provisioning of resources on a best effort guiding principle of operation and as interest in commercial utilisation of Grids surmounted, more stringent guarantees on the management of resources via QoS were realised as a necessity for the wide spread adoption of Grids to take place in industry[58].

The following two subsections of this paper will focus on the management of resources to guarantee performance and the technology in place to facilitate the reservation of resources, both of which are highly relevant to Cloud Computing.

3.1 Resource Management

Without the management of resources Grids would be unable to function. Resource management encompasses the dynamic allocation of tasks to computational resource and requires the use of a scheduler (or broker) to guarantee

performance. QoS is enabled in Grids by the efficient scheduling of tasks, this guarantees that resource requirements of an application are strictly supported but resources are not over provisioned and used in the most efficient manor possible. Sequences of tasks are represented as workflows, directed graphs comprised of precedent constrained nodes, which each represent the specific ordered invocation of a service on computation resources to process a given task. Several research projects have tackled the complexities of resource reservation and allocation in Grids such as the Phosphorus Project[19] and utilise schedulers such as DSRT[35] and PBS[20].

Monitoring tools are essential in ascertaining the availability of resources and providing feedback to schedulers within Grids. Monitoring tools enable guarantees to be made on the performance of any given resource by making sure that the computational resource in question is not over utilised and is online. Performance is characterised by the amount of useful work accomplished by a computer system in comparison to the time and resources used. Monitoring tools are also essential in providing fault tolerance and the migration of tasks in the event of a resource failure in the Grid. Fault tolerance involves the identification of a resource failure via monitoring tools, the rescheduling of the task to an alternative available resource and migration of the state of the task to the newly allotted resource, at which point the task continues execution. The state of a task in execution must be regularly saved for fault tolerance to function, this process is known as check pointing. Many monitoring tools have been developed for Grids[63, 54, 49, 61, 64].

An example of a Grid software stack enabling resource management is The Globus Toolkit[7]. It has become the academic and industry leading open source software solution for building Grids and provides the necessary middleware to manage and monitor resources.

3.2 Service Level Agreement Standardisation

As the importance of Service Level Agreements (SLAs) as facilitators for the widening commercial uptake of Grids has grown, substantial effort has been made in standardising their use. The Web Services Agreement Specification (WS-Agreement)[29] is one such standardisation effort by the Open Grid Forum[17]. WS-Agreement is a Web Services protocol for establishing an agreement between two parties, using an extensible XML language for specifying the content of an agreement, and agreement templates used to discover appropriate agreement parties. The specification consists of three parts

- A schema for specifying an agreement.
- A schema for specifying an agreement template.
- A set of operations for managing an agreements life-cycle.

Although WS-Agreement can be effectively used to facilitate SLAs, the life-cycle model does not accommodate the dynamic nature of the Grid economy, providing facilities to negotiate and renegotiate an agreement. The current state of the art research in QoS within Grids is concentrating on this problem[38, 60]. Another cutting edge topic of research surrounding QoS in Grid Computing, is solving problems related to risk assessment and dependability of service providers and is being tackled by projects such as AssessGrid[39]. The AssessGrid

Consortium[1] have researched heavily into QoS but more specifically SLA's. Many of the objectives of the project are also relevant in the context of Cloud Computing, such as how to evaluate the reliability of Cloud service providers and how best to estimated the risks involved in accepting any given SLA but are not relevant to the scope of this paper.

4 Clouds Today

4.1 Commercial Cloud Adopters

An overview of commercial cloud vendors, the technology they have in place and the state of their QoS provisioning is essential for the priorities of academic research to be in sink with the needs of businesses and for research in to QoS to be of real world intrinsic value.

In this paper four main commercial adopters of Cloud technology will be discussed, which are providing services and software products guiding the direction of research in Cloud Computing. Amazon the first company to supply Cloud infrastructure services via its Amazon Web Service[2] products in early 2006, provides a PaaS architecture on a pay per use financial model. The architecture is marketed as two individual products the Amazon Elastic Compute Cloud (Amazon EC2) and the Amazon Simple Storage Service (Amazon S3) and a set of well defined API's that are becoming widely adopted as standards in many open source Cloud architectures such as Enomalism[4], Eucalyptus[5] and OpenNebula[18]. These projects are providing interface compatible with Amazon's services to enable on demand scale out of service workloads to supplement local resources to satisfy peak or fluctuating demands.

Another contender positioning themselves as a provider of Cloud services is Google. Google provides SaaS via its Google Apps[8] software and an PaaS via its Google App Engine[9]. The Google App Engine provides the architecture that Google Apps run on and promises transparent scalability on a pay per use financial model. The Google App Engine is limited to a set of Python API's that provide a proprietary data storage query language and other cloud related services.

IBM has released literature on its vision of cloud computing[32] and provides a PaaS based around the API's created by Amazon, know as IBM's Research Compute Cloud[14]. IBM also supplies enterprise Cloud Computing solutions in the form of Cloud Service know as IBM Computing on Demand[15].

All the aforementioned major commercial cloud vendors provide best effort QoS provisioning and provide only the most basic guarantees on the availability and performance of resources, primary motivation to research further and publicise the benefits of QoS in Cloud Computing.

Microsoft the final commercial organisation with an interest in Cloud Computing, is not providing Cloud services but is instead developing the Azure Services Platform[3], a PaaS operating system, which integrates many of Microsoft's current proprietary software packages into one package via a layer of middleware, that can be utilised by licensed cloud vendors and is being marketed as an all-in-one Cloud software solution.

Due to the closed source proprietary nature of these commercial Clouds, limitations are present concerning interoperability. Unlike Grids the nature

of Clouds are very much orientated towards providing services behind closed doors. This is resulting in an emergent topic of research investigating the development of Cloud standards to enable the sharing of Cloud resources outside administrative and organisational boundaries, standards that could also encompass QoS and provide the basis for Cloud brokering systems which are currently impossible without standard interfaces to communicate with Cloud services and descriptive languages to define Cloud services.

4.2 Open Source Cloud Implementations

Understanding the specific technical problems surrounding QoS is not possible in commercial Clouds as the services they provide are transparent, the End-User has no idea of the underlying implementation. Advanced understanding and knowledge of all aspects of Cloud Computing in detail is required to understand the limitations present in commercial QoS provisioning. This subsection discusses three popular distributions of open source Cloud architectures that can be used to shed light on how QoS would be best integrated into a Cloud and which architecture would be best suited as a testbed for use in QoS research. These packages fall into IaaS or PaaS cloud system previously discussed. SaaS systems have been omitted from the scope of this paper as there are currently no out of the box open source solutions available, most likely due to a lack of PaaS API standards, but there are commercial entities such as SalesForce[23] that provide SaaS packages that can run on Amazon Web Services.

The first Cloud Architecture, Eucalyptus[5], is an IaaS system with the aim of creating:

> "an open-source infrastructure architected specifically to support cloud computing research and infrastructure development."[57]

The system combines a Cloud Controller responsible for processing incoming user requests, manipulating the Cloud fabric and processing SLAs in company with a Client Interface that utilises Internet standard protocols for instance HTTP XML and SOAP. The second Cloud architecture evaluated named Enomalism[4] is another IaaS system that presents an organisation with the capability to manage virtual infrastructures (including networks), virtual machine images and fine grained security and group management, in addition to the creation of virtual machine images. The third Cloud architecture evaluated, OpenNebula[18], based on the research being performed by the Reservoir Project[22], the European research initiative in virtualised infrastructure and cloud computing, combines both features of IaaS and PaaS in one architecture. The Reservoir Project primary deliverable is a complete definition of a reference architecture built on open standards to provide a framework for the delivery of scalable, flexible and dependable services. The project intends to develop key technologies enabling the migration of virtual machines across network and storage boundaries, algorithms for the effective allocation of resources conformant to SLA requirements and a test bed to benchmark the performance of the architecture in industrial and commercial uses cases.

Comparing these open source Cloud architectures from their present state and pace of development, the most promising is OpenNebula, due to its support from an academic research group actively publishing research and is the most feature complete system with concrete direction for future development.

The OpenNebula architecture has also been designed with modularity in mind making extensions easier to develop on scales applicable to academic research by outsiders. An example of this is Haizea[13], which can be used to replace the existing resource scheduler in OpenNebula. Haizea is an open source virtual machine lease manager that provides a resource management model[30] for virtual Cloud infrastructures, combining batch execution of applications such as scientific workflows on leased virtual resources[62].

4.3 Resource Virtualisation

Understanding Cloud architectures from the bottom up, starting with the technology that supports the provisioning of resources, both physical and virtual, in Cloud infrastructures is key to understanding the importance of QoS in Cloud Computing and how its implementation will differ from that of Grid Computing. The current state of the art technology in Cloud Computing centres on the virtualisation of resources at the lowest level, a characteristic that distinguishes Clouds from Grids. The main technology enabling virtualisation is the Hypervisor, a Virtual Machine Manager (VMM) that partitions a physical host server transparently via emulation or hardware-assisted virtualisation. This provides a complete simulated hardware environment; know as a virtual machine, in which a guest operating system can execute in complete isolation. There are several benefits of utilising virtual machines. Hardware can be consolidated when several servers are underutilised and provisioned as needed endowing a organisation with reductions in the up-front cost of hardware purchases and virtual machines can be migrated from one physical location to another with ease as the need arises. Academics can also benefit from utilising virtual machines. There are often limitation imposed on Grid users to what software they can use to develop a computer based simulation experiment. There are no such limitations on the availability of software that can be installed into virtual machine images.

There are five types of virtualisation:

- Full Virtualisation
- Hardware Assisted Virtualisation
- Partial Virtualisation
- Paravirtualisation
- Hybrid Virtualisation
- Operation System-Level Virtualisation

Full Virtualisation involves simulating enough hardware to allow an unmodified guest operating system to run in isolation, at a considerable performance penalty due to the overhead associated with emulating hardware. Hardware Assisted Virtualisation utilises the additional hardware capabilities, in the form of additional virtual machine extensions (VMX) within the host processor instruction set, to accelerate and isolate context switching between processes running in different virtual machines. This increases the computational performance of a virtual machine as instructions can be directly passed to the host processor without having to be interpreted and isolated at the expense of limiting guest operating systems to using the same instruction set as the host machine.

Partial Virtualisation involves the simulation of most but not all the underlying hardware of host and supports resource sharing but does not isolate guest operating system instances. This basic approach is utilised in Paravirtualisation, Hybrid Virtualisation and Operating System-Level Virtualisation. Paravirtualisation simulates all or most hardware by providing software interface or API's that are similar to that provide for the underlying hardware of the host. These can be utilised to create hardware device drivers for guest operating systems that achieve near native performance to that on the host. The downside of this approach is that the operating system must be modified to run on Paravirtualised VMM's.

Hybrid Virtualisation combines the principles of both Hardware Assisted Virtualisation and Paravirtualisation[56] to obtain near native performance from guest operating systems but with disadvantages of both. Although these disadvantages prevent the consolidation of an organisation's current hardware they do provide an excellent foundation for the creation of new clouds based systems, reducing the number of physical machines needed at peak demand and thus hardware running and setup costs. Most VMM's support multiple types of virtualisation so the disadvantages can be some what mitigated. Operating System-Level Virtualisation is achieved through multiple isolated user space instances. A disadvantage of this virtualisation technique is that the guest operating system of the virtual machine must be the same as the host, but the guests run at native performance.

There are three main VMM's that have been widely adopted in Cloud Computing architectural development. VMware ESX[27] a successful commercial VMM provides Full and Hybrid Virtualisation. The open source alternatives are:

- Xen[31]
- KVM[16]

Xen is Hybrid Virtualisation VMM and is utilised in Amazon's EC2, while KVM is a Hardware Assisted Virtualisation VMM. Xen and KVM both contain built in Full Virtualisation support, via their integration with QEMU[21], for operating systems that cannot be altered. QEMU is another open source Full Virtualisation VMM providing emulation of both IO devices, such as network interfaces cards and CPU architectures through binary translation. All the open source Cloud architectures discussed in this paper support these two VMM's.

Distributed storage like in Grid Computing and other distrusted paradigms plays a large role in the scalability of Cloud Computing. Hadoop[12] based around the map/reduce functional programming principles, utilised in the backend of Google App Engine API's, is a software service that provides access to large amounts of virtualised storage. Further research is still required in Cloud storage services as the efficiency and performance of data storage and management can become a bottleneck in distributed systems and thus effecting QoS. Research into storage provision has also played a role in the development of the Grid Paradigm and new standards have arisen, such as the GridFTP[28] data transfer protocol.

It is more than likely that research in Cloud Computing will place emphasis on the efficient distribution and replication of data geographically as check pointing, fault tolerance and migration of large virtual machine images is improved.

5 Some Research Directions

The following list summarises the material discussed in this paper and highlights various additional challenges that are present in QoS in Cloud Computing. This list of issues is by no means exhaustive but provides a good foundation for tackling some interesting research questions.

- As with past and present Grid Computing projects[25, 26, 10, 24], SLAs will play a major role in the development of the Cloud Computing paradigm. Within the research topic of QoS in Clouds, emphasis will have to be placed on the performance of virtualisation technology and the tools necessary to monitor virtualised hardware.

- Simulation and modelling has furthered the understanding of Grid Architectures and will do so in Cloud Computing. Advancements in simulation and modelling techniques will aid in the better understanding, usability and streamlining of Cloud environments, as was seen in the development of Grids. Progress is already being made towards the development of a simulation tool. Currently the Grid Computing and Distributed Systems Laboratory from The University of Melbourne has identified the need for such a tool to support the performance evaluation of Cloud environments and is in the early stages of development, named CloudSim[34]. The simulator is based around the programming framework they previously created to model Grids, in the Grid Simulator, GridSim[11].

- Due to the increased commercial interest in Cloud Computing and the perceived lack of commercially viable Grid Systems within industry and the similar nature of technical problems that have been solved in Grid Computing and need to be solved in Cloud Computing, interest has been amplified in the interoperability between Grid and Cloud technology and how the two paradigms can complement each other. Nimbus is a prime example, created by The Globus Alliance[6], a community of organisations that have developed fundamental technologies in Grid Computing. Nimbus is a IaaS Cloud that enables the use of virtualised resource within Grids. It can be utilised with familiar Grid technologies, for instance the Web Service Resource Framework, Portable Batch System and Sun Grid Engine schedulers.

- An interesting extension and relevant topic of research related to interoperability, could be to investigate the possibility of Clouds of Grids and Grids of Clouds, how best to integrate entire Cloud and Grid systems into each other across organisational boundaries.

- Another open research question surrounding QoS in Cloud Computing, which could be investigated, are the problems surrounding the transparent management of data resources, used by virtual machines, to perform job and service migration, fault tolerance and check pointing of tasks. Currently these problems are overlooked by many commercial cloud vendors but could possibly be remedied in the future but are inherently difficult to solve due to the large quantities of volatile data associated with virtual machines that need to be transferred, stored or backed up for such tasks to complete with a Cloud. This is not as much of a problem in traditional

Grid architectures as migration, fault tolerance and check pointing are often dwelt with by the applications running on the Grid, allowing the developer to optimise the amount of volatile data needed to be worked with. This has the disadvantage of complicating the development and maintenance of the applications created, where the advantages of a Cloud platform that could perform these tasks transparently can obviously be seen. The interest in Virtual Machine technology in Cloud Computing has seen a recent resurgence of research into the problems surrounding virtual machine image migration[47, 48, 65, 46, 45, 44] and would provide a sensible starting point for understanding the research problems surrounding fault tolerance and check pointing as they are closely related subjects.

6 Conclusion

In this paper the confusion surrounding the term "Cloud", the current consensus of what Cloud Computing is and the relevance of QoS in Clouds have been discussed. The importance of Grid Computing heritage in Clouds has been explained and the relevance of past QoS research in Grids discussed. The motivation behind, concepts, technology, projects and the state of QoS in Cloud Computing have been reviewed. A vision of some of the problems surrounding QoS in Cloud Computing is constructed through the proposition of open research questions.

7 References

References

[1] Assessgrid - advanced risk assessment and management for trustable grids. Website, December 2008. http://www.assessgrid.eu/.

[2] Amazon web services. Website, January 2009. http://aws.amazon.com.

[3] Azure services platform. Website, March 2009. http://www.microsoft.com/azure.

[4] Enomalism - elastic computing platform. Website, February 2009. http://www.enomaly.com.

[5] Eucalyptus - elastic utility computing architecture. Website, February 2009. http://eucalyptus.cs.ucsb.edu/.

[6] Globus alliance. Website, November 2009. http://www.globus.org.

[7] The globus toolkit - an open source software toolkit for building grids. Website, May 2009. http://www.globus.org/toolkit/.

[8] Google apps. Website, February 2009. http://www.google.com/apps/business.

[9] Google apps engine. Website, February 2009. http://code.google.com/appengine.

[10] Gria - service oriented collaborations for industry and commerce. Website, March 2009. http://www.gria.org/.

[11] Gridsim - grid simulation toolkit. Website, March 2009. http://www.gridbus.org/gridsim.

[12] Hadoop - distributed file system. Website, January 2009. http://hadoop.apache.org/core.

[13] Haizea - an open-source virtual machine-based lease management architecture. Website, April 2009. http://haizea.cs.uchicago.edu/index.html.

[14] Ibm - cloud computing. Website, March 2009. http://www.ibm.com/ibm/cloud/ibm_cloud/.

[15] Ibm - computing on demand. Website, March 2009. http://www-03.ibm.com/systems/deepcomputing/cod/.

[16] Kvm - kernel based virtual machine. Website, January 2009. http://www.linux-kvm.org.

[17] Open grid forum. Website, May 2009. http://www.ogf.org/.

[18] Opennebula - the engine for data center virtualization and cloud solutions. Website, February 2009. http://www.opennebula.org.

[19] Phosphorus - qos in scientific communities. Website, March 2009. http://www.ist-phosphorus.eu.

[20] Portable batch system (pbs). Website, May 2009. http://www.nas.nasa.gov/Software/PBS/pbsnashome.html.

[21] Qemu - open source processor emulation. Website, January 2009. www.qemu.org.

[22] Reservoir - resources and services virtualization without barriers. Website, February 2009. http://www.reservoir-fp7.eu.

[23] Salesforce - the crm software as a service (saas) leader. Website, March 2009. http://www.salesforce.com/.

[24] Sla4dgrid - service-level agreements for the national d-grid initiative. Website, March 2009. http://www-ds.e-technik.uni-dortmund.de/~yahya/.

[25] Sla@soi - empowering the service industry with sla-aware infrastructures. Website, March 2009. http://sla-at-soi.eu/.

[26] Sorma - self-organizing ict resource management. Website, March 2009. http://sorma-project.org/.

[27] Vmware esx datasheet. Electronic Brochure, January 2009. http://www.vmware.com/.

[28] W. Allcock, J. Bresnahan, R. Kettimuthu, and M. Link. The globus striped gridftp framework and server. pages 54 – 54, November 2005.

[29] A. Andrieux, K. Czajkowski, A. Dan, K. Keahey, H. Ludwig, T. Kakata, J. Pruyne, J. Rofrano, S. Tuecke, and M. Xu. Web services agreement specification (ws-agreement). Technical report, Open Grid Forum, May 2007. http://www.ogf.org/documents/GFD.107.pdf.

[30] Borja Sotomayor Basilio. A resource management model for vm-based virtual workspaces. Master's thesis, The University of Chicago, Department of Computer Science, February 2007.

[31] Nikhil Bhatia and Jeffrey S. Vetter. *Virtual Cluster Management with Xen*, volume 4854 of *Lecture Notes in Computer Science*, chapter 25, pages 185 – 194. Springer Berlin / Heidelberg, March 2008.

[32] Greg Boss, Padma Malladi, Dennis Quan, Linda Legregni, and Harold Hall. Cloud computing. White paper, IBM High Performance On Demand Solutions (HiPODS), October 2007.

[33] James Broberg, Srikumar Venugopal, and Rajkumar Buyya. Market-oriented grids and utility computing: The state-of-the-art and future directions. *Journal of Grid Computing*, 6(3):255 – 276, 2008.

[34] Rodrigo N. Calheiros, Rajiv Ranjan, Cesar A. F. De Rose, and Rajkumar Buyya. Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services. Technical report, The Grid Computing and Distributed Systems (GRIDS) Laboratory, University of Melbourne, March 2009.

[35] Hao-Hua Chu and K. Nahrstedt. Cpu service classes for multimedia applications. volume 1, pages 296 – 301, Jul 1999.

[36] L. Chunlin and L. Layuan. Cross-layer optimization policy for qos scheduling in computational grid. *Journal of Network and Computer Applications*, 31(3):258 – 84, August 2008.

[37] David Colling, T Ferrari, Youssef Hassoun, Chenxi Huang, C Kotsokalis, Stephen McGough, Yash Patel, E Ronchieri, and P Tsanakas. On quality of service support for grid computing. April 2007.

[38] Giuseppe Di Modica, Orazio Tomarchio, and Lorenzo Vita. Dynamic slas management in service oriented environments. *Journal of Systems and Software*, 82(5):759 – 771, 2009.

[39] Karim Djemame, Iain Gourlay, James Padgett, Georg Birkenheuer, Matthias Hovestadt, Odej Kao, and Kerstin Voss. Introducing risk management into the grid. page 28, Washington, DC, USA, 2006. IEEE Computer Society.

[40] N. Dube and M. Parizeau. Utility computing and market-based scheduling: shortcomings for grid resources sharing and the next steps. pages 59 – 68, Piscataway, NJ, USA, 2008.

[41] I. Foster, Yong Zhao, I. Raicu, and S. Lu. Cloud computing and grid computing 360-degree compared. page 10, Piscataway, NJ, USA, 2008.

[42] Ian Foster. What is the grid? a three point checklist. GridToday, July 2002. http://www-fp.mcs.anl.gov/~foster/Articles/WhatIsTheGrid.pdf.

[43] Jeremy Geelan. Twenty-one experts define cloud computing. Electronic Journal, January 2009. http://cloudcomputing.sys-con.com/node/612375.

[44] Sriram Govindan, Arjun R. Nath, Amitayu Das, Bhuvan Urgaonkar, and Anand Sivasubramaniam. Xen and co.: Communication-aware cpu scheduling for consolidated xen-based hosting platforms. pages 126 – 136, San Diego, CA, United states, 2007.

[45] Eric Harney, Sebastien Goasguen, Jim Martin, Mike Murphy, and Mike Westall. The efficacy of live virtual machine migrations over the internet. Reno, NV, United states, 2007.

[46] Michael R. Hines and Kartik Gopalan. Post-copy based live virtual machine migration using adaptive pre-paging and dynamic self-ballooning. pages 51 – 60, New York, NY, USA, 2009. ACM.

[47] Wei Huang, Qi Gao, Jiuxing Liu, and Dhabaleswar K. Panda. High performance virtual machine migration with rdma over modern interconnects. pages 11 – 20, Austin, TX, United states, 2007.

[48] Wei Huang, Jiuxing Liu, Matthew Koop, Bulent Abali, and Dhabaleswar Panda. Nomad: migrating os-bypass networks in virtual machines. pages 158 – 168, New York, NY, USA, 2007. ACM.

[49] Emir Imamagic and Dobrisa Dobrenic. Grid infrastructure monitoring system based on nagios. pages 23 – 28, Monterey, CA, United states, 2007.

[50] S. Jarvis, N. Thomas, and A. van Moorsel. Open issues in grid performability. *International Journal of Simulation: Systems, Science and Technology*, 5(5):3 – 12, December 2004.

[51] A. Keller, K. Voss, D. Battre, M. Hovestadt, and O. Kao. Quality assurance of grid service provisioning by risk aware managing of resource failures. pages 149 – 57, Piscataway, NJ, USA, 2008.

[52] Huan Liu and D. Orban. Gridbatch: Cloud computing for large-scale data-intensive batch applications. pages 295 – 305, Piscataway, NJ, USA, 2008.

[53] Ignacio M. Llorente, Ruben S. Montero, Eduardo Huedo, and Katia Leal. A grid infrastructure for utility computing. pages 163 – 168, Manchester, United kingdom, 2006.

[54] M.L. Massie, B.N. Chun, and D.E. Culler. The ganglia distributed monitoring system: design, implementation, and experience. *Parallel Computing*, 30(7):817 – 40, July 2004.

[55] Jarek Nabrzyski, Jennifer M. Schopf, and Jan Weglarz, editors. *Grid resource management: state of the art and future trends*. Kluwer Academic Publishers, Norwell, MA, USA, 2004.

[56] Jun Nakajima and Mallick Asit K. Hybrid virtualization - enhanced virtualization for linux. 2007.

[57] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, and Dmitrii Zagorodnov. Eucalyptus: A technical report on an elastic utility computing architecture linking your programs to useful systems. Technical report, Computer Science Department University of California, Santa Barbara Santa Barbara, California 93106, October 2008.

[58] A. Sahai, S. Graupner, V. Machiraju, and A. van Moorsel. Specifying and monitoring guarantees in commercial grids through sla. pages 292 – 299, May 2003.

[59] Jennifer M. Schopf and Bill Nitzberg. Grids: The top ten questions. *Scientific Programming*, 10(2):103 – 111, 2002.

[60] S. Sharaf and K. Djemame. An application of dynamic service level agreements in a risk-aware grid environment. To appear in the Proceedings of the 2009 International Conference on Grid Computing and Applications (GCA'2009), Las Vegas, Nevada, July 2009.

[61] Shava Smallen, Kate Ericson, Jim Hayes, and Catherine Olschanowsky. User-level grid monitoring with inca 2. pages 29 – 38, Monterey, CA, United states, 2007.

[62] Borja Sotomayor, Kate Keahey, and Ian Foster. Combining batch execution and leasing using virtual machines. pages 87 – 96, Boston, MA, United states, 2008.

[63] M. Swany and R. Wolski. Representing dynamic performance information in grid environments with the network weather service. pages 48–48, May 2002.

[64] B. Tierney, W. Johnston, B. Crowley, G. Hoo, C. Brooks, and D. Gunter. The netlogger methodology for high performance distributed systems performance analysis. pages 260 – 267, July 1998.

[65] Anand Tikotekar, Geoffroy Vallee, Thomas Naughton, Hong Ong, Christian Engelmann, Stephen L. Scott, and Anthony M. Filippi. Effects of virtualization on a scientific application running a hyperspectral radiative transfer code on virtual machines. pages 16 – 23, Glasgow, United kingdom, 2008.

[66] Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres, and Maik Lindner. A break in the clouds: towards a cloud definition. *SIGCOMM Comput. Commun. Rev.*, 39(1):50 – 55, 2009.

[67] M.A. Vouk. Cloud computing - issues, research and implementations. pages 31 – 40, Piscataway, NJ, USA, 2008.