



From Service-oriented IT to Business Value

- [About Grid](#)
- [Technical](#)
- [Grid Business](#)
- [Download](#)
- [Contribute](#)

You are here: [Gridipedia](#) →

British Publishers 
media with influence

HPC 

Gridipedia Partners:

In the spotlight:



[Grid Voices: Grids and Clouds in a multi-core world](#)

Posted by Rosa M. Badia. While we already have had quite a bit of discussion about the Grid/Cloud buzz words, I would like to raise your attention on one more of these type of words: multicore. The forecast is to have thousands...

[\[...\] Read more](#)

[24-04-2009: Gridipedia Top 5 This Week!](#)

Cloud Computing Providers -

GridVoices: Virtual World Interoperability: Problem Solved? -

Security Token Service -

Virtualization and Cloud -

Application Virtualization

[\[...\] Read more](#)

Gridipedia launches Grid Voices Blog

A new blog, Grid Voices, is launched by Gridipedia

[\[...\] Read more](#)



Welcome to the Grid Voices Blog – A blog about Grid technologies, cloud computing, [service-oriented architecture](#) and distributed computing. We will analyse the technology area, providing the point of view of leading members of the research community and the industrial firms leading the commercial uptake – such as [BT](#) and [Atos Origin](#), and academic institutions such as the [Fraunhofer Institut](#), the [High Performance Computing Centre in Stuttgart](#) and the [National Technical University of Athens](#). You can find the full list of [Grid Voices contributors](#) here.

Grids and Clouds in a multi-core world

Posted by [Rosa M. Badia](#) 13 May 2009

While we already have had quite a bit of discussion about the Grid/Cloud buzz words, I would like to raise your attention on one more of these type of words: *multicore*. The forecast is to have thousands of cores in a chip. How will this impact the way we do computing? How will this impact the way we program? How will this impact grid and cloud computing?

With the objective of overcoming the three walls: the memory wall, the power wall and the ILP wall, the current trends in chip fabrication have led to placing more than one processor (from now on, core) in a chip. While manufacturers are now shipping chips with a few cores (at most 4-8 cores), the forecast is to include hundreds or thousands of them in a chip within a few years. What is more, while now only a few of these chips has an heterogeneous nature with a non traditional memory organization, like the controversial Cell chip, in the future the prediction is to have highly heterogeneous organizations in a chip, with complex memory hierarchies and different type of cores (like accelerators and GPUs).

The community is currently very excited with this new and performant chips, but also aware of the increasing complexity of software development in these platforms. While current programming methodologies can be used with up to 4-8 cores, new methods that enable the parallel execution of applications need to be devised. The pressure cannot be only on the programmers, but also on the programming models that should be able to abstract the underlying architecture, and even more enable, if possible, automatic parallelization and perform the required data transfers between different levels of the memory, all in a very dynamical fashion. Additionally, now more than ever we cannot expect programmers to tune and re-program their applications every time a new architecture appears, and therefore portability or the so called performance portability is a pre-requisite.

Having this in mind, there are several considerations to make with regard to Grid computing. The first is about programming: the principles that guided the research in programming the grid were very close, nor to say identical to the ones described above for multicore chips. The goals of programming models for the grid were: to be able to manage computing environments that are inherently parallel, distributed, heterogeneous and dynamic, both in terms of the resources involved and their performance. While it may be possible to build grid applications using established programming tools, they are not particularly well-suited to effectively manage flexible composition or deal with heterogeneous hierarchies of machines, data and networks with heterogeneous performance. Therefore, the experience of the research

performed in the recent years in programming the grid can be applied to multi-core programming. Successful examples of this are environments like [ProActive](#) or GRID superscalar ([GRIDSs](#), in the form of their siblings [CellSs](#) or [SMPSs](#)). Feedback from the groups that have done research on how to program the grid into standards such as OpenMP (with its current movement towards considering parallel tasks) and OpenCL (that considers the heterogeneity of the systems) can be key.

A second consideration is given that compute nodes will be much more powerful in the near future one can think that there would not be (or would be less) need for computational grids as we conceive them now, given that the fat local computing nodes will be enough to satisfy the needs of computing. There are even some voices that maintain that in fact there would be *too much computing*, and that the problem is to find applications that need it. However, my opinion is that we will continue to have a need for grids. These forthcoming grids will be grids of fatter nodes, but the community will be able to conceive applications that need this computing (we do not have to forget that, for example, some scientific communities that, like expanding gases fit all available space, are always able to consume all available cycles). An important consideration here that supports this idea is the fact that even considering a world with local computers, the data sources and communities will continue to be distributed. Therefore, the need for grids and grid software will remain.

Finally, how multi-cores will impact cloud computing? Similarly to grid computing, cloud computing will be enabled with fatter computing nodes. Also, and this applies also to grid computing, the cloud middleware will have to be adapted to consider the underlying multi-core hardware. Since most of cloud computing technology is based on virtualization, the key here is the enablement of this technology to multicore taking into account that these would be much more complex and heterogeneous. The new multicore platforms enable to host more than one instance of the operating system and the challenge now is how to perform the right dynamic resource management of the virtualized systems to meet the Service Level Agreements established with the end-users.

Thanks to Daniele Lezzi for discussion and comments.

About the author



[Rosa M. Badia](#) has a PhD in Computer Science (1994) from the Technical University of Catalonia (UPC). Since year 2008 she has been a Scientific Researcher from the Consejo Superior de Investigaciones Científicas (CSIC). And has been the manager of the Grid Computing and Cluster research group at the Barcelona Supercomputing Center (BSC) since 2005. She has been involved in teaching and research activities at the UPC since 1989, where she has been an Associated Professor since 1997. From 1999 to 2005 she has been involved in research and development activities at the European Center of Parallelism of Barcelona (CEPBA). Her current research interest are: performance prediction and modelling of MPI programs and programming models for complex platforms (from multicore to the Grid). She has participated in several European projects (like DAMIEN or CoreGRID) and currently she is participating in projects BeinGRID, Brein, XtremOS and OGF-Europe and she is a member of the HiPEAC2 NoE.

Virtual World Interoperability: Problem Solved?

Posted by [Theodora A. Varvarigou](#) 24 April 2009

Virtual World interoperability is currently a hot topic in the research world. There are at least two major initiatives trying to tackle the issue by setting up a standard for enabling interoperability: the joint effort from Linden Labs and IBM within the auspices of the Internet Engineering Task Force (IETF), and the MPEG-V, which is taking place within the auspices of the International

Standardisation Organisation (ISO) with Philips driving the effort. At the same time there are numerous virtual world platforms based on different implementations (and some widely used open source ones such as Project Wonderland and OpenSim) and a great number of users which is constantly increasing. This is only an indication of the virtual world dynamics and in turn, how enormous are the technical challenges, especially for an application that is being traditionally based on well-established software engineering technologies such as multi-tier (client-server) architectures.

But let's talk a little bit about these challenges. Apart from the presentation issues that arise when someone attempts to port an avatar from a virtual world platform to another which are more related to 3D modelling and 3D-engines in general, there are other more hardcore challenges such as data, identity and license schemes management, security, privacy, trust and common interfaces issues. And addressing these aspects is crucial if one wants to achieve end-to-end interoperability coupled with viable business models.

For one, data aggregation is required for dealing with the variable data formats and structures that each virtual world is using while secure data transmission is necessary to protect sensitive private and commercial information. Then, it is the common API for application development in several virtual world platforms. Moreover, even though actual user identity is not always reflected in virtual worlds -mainly because of current vague registration and authorization schemes-, the identity that is attached to the avatar must be carried throughout the virtual island hopping adventure. Finally, a trust framework must be established between virtual world providers which are often antagonistic, by allowing them to gain control over the type of information that they are giving out as well as the other quality terms that govern these relationships.

However, a more careful study, will lead the meticulous reader to the conclusion that all the above seem to be aspects of the more general problem of complex system heterogeneity, scalability and business relationship management. This, coupled with the fact that virtual worlds are by nature service-based applications, rings many bells to the researchers that work in distributed computing. Several of the abovementioned problems have been addressed to a large extend by Grid computing, i.e. data aggregation, end-to-end security, trust establishment and SLA management, functional interoperability through common infrastructure services and the list goes on. BEinGRID alone covers many of these issues. One has only to check some of the technical solutions in Gridipedia to find out that we can start building an interoperability middleware for virtual world interoperability using these as a baseline SOI.

The bottom line is that problems which have been addressed by Grids (even with functionalities exposed through Clouds) seem to remain all-the-rage and re-using such solutions even partially makes great sense. It is not necessary that it will worth the effort, but it is necessary to investigate it.

About the Author



Prof. Theodora A. Varvarigou received the B. Tech degree from the National Technical University of Athens, Athens, Greece in 1988, the MS degrees in Electrical Engineering (1989) and in Computer Science (1991) from Stanford University, Stanford, California in 1989 and the Ph.D. degree from Stanford University as well in 1991. She worked at AT&T Bell Labs, Holmdel, New Jersey between 1991 and 1995. Between 1995 and 1997 she worked as an Assistant

Professor at the Technical University of Crete, Chania, Greece. In 1997 she was elected as an Assistant Professor while since 2007 she is a Professor at the National Technical University of Athens, and Director of the Postgraduate Course "Engineering Economics Systems". Prof. Varvarigou has great experience in the area of semantic web technologies, scheduling over distributed platforms, embedded systems and grid computing. In this area, she has published more than 150 papers in leading journals and conferences. She has participated and coordinated several EU funded projects, including IRMOS, SCOVIS, ANSWER, BEinGRID, AKOGRIMO, POLYMNIA, NextGRID, FIDIS, MARIDES.

Grid Standards and the Global IT Industry

Posted by [Mark Parsons](#), 14 April 2009

Grid standards were supposed to make it simple to create interoperable service-oriented Grids. However, very few global Grid standards have been agreed and even fewer have been widely adopted. Why is this and what does it tell us about the global IT industry today?

The early part of this decade saw an explosion of Grid research projects trying to understand how to build a new generation of distributed computing applications which would revolutionise science and business. I was involved in this process from the beginning – helping to write CERN’s DataGRID proposal in 2000 and writing and leading a series of UK and European Grid research projects and spending many millions of Euros in the process.

From the beginning the funding bodies insisted we devote a considerable amount of our time to “standards setting”. Ever keen to ensure our proposals were funded we dutifully undertook to focus on standardisation as a key exploitation outcome. I now wonder if this was the right choice.

Many of the people I’ve worked with over the past decade have spent countless hours pursuing standards in face-to-face meetings around the world or teleconferences that have either never come to fruition or, when they have done, have been so diluted as to be pointless.

It was clear that early implementations of Grid middleware were poorly written lash-ups; written to prove an idea rather than deliver product quality software. By proposing the Open Grid Services Infrastructure (OGSI) in 2003, the Global Grid Forum (largely led in this case by the Globus project and IBM) tried to move from a non-standardised distributed computing framework to a web service based platform which could be extended over time. This of course upset everyone. I now believe that this was because people wanted to experiment more before they were constrained into the OGSI standards.

It’s very interesting to look at how HTML came about. People had been playing with hypertext for many years before Tim Berners-Lee used the ideas to create HTML 1.0. At the outset HTML wasn’t a standard but it was rapidly picked up and used by so many people that it became one by default. Given the economic value that has ensued from HTML, I don’t believe it could be created today by a standards process. None of our large IT companies would sit down and allow any of their competitors to gain such advantage and the discussions would be endless.

In many ways this is what has happened to Grid standards since OGSI. Many in the IT industry saw OGSI as an IBM-inspired land-grab. All of the major vendors then marched their troops into the standards arena – largely sidelining the researchers who had conceived the underlying ideas in the first place – and halted all appreciable progress in Grid services standards thereafter. WS-RF and WS-DM tried to bring everyone together, but Microsoft and Sun countered with WS-Management. A grand plan was hatched to bring WS-DM and WS-Management together but this has still to see the light of day.

So what does this tell us about the Grid? It tells us that the distributed computing community hit upon an interesting research area at the turn of the century. So interesting in fact that the major IT vendors felt sufficiently threatened to spend large amounts of their money debating standards that now seem pointless. The world has moved on. Those of us who deliver real-world Grid computing solutions use what’s available – and we learn more about how to use these pieces of the Grid jigsaw puzzle every day. We also know that standards are important but only after the technologies on which they are based have been proven – not before – and certainly not because one IT vendor or another says so.

About the author



Dr Mark Parsons is Commercial Director of EPCC, the supercomputing centre at The University of Edinburgh. Mark joined EPCC in 1994 as a software developer working on various industrial contracts before being appointed the Centre's Commercial Director in 1996. He is a leading figure in the European Grid and HPC community and has led several of the major EC funded Grid projects. With degrees in Physics and Digital Microelectronics, Computer Science and Particle Physics, Dr Parsons has a very broad spread of knowledge in the area of advanced IT.

Where the Cloud meets the Grid

Posted by [Adrian Mouat](#), 01 April 2009

The term "Cloud computing" has seen an enormous rise in popularity since its inception in 2007. This [article](#) highlights the slow retreat of the terms Utility computing and Grid computing against the sudden surge of Cloud computing.

But what exactly is the difference between Cloud computing and Grid computing? A lot of people have written about this, but few have come up with a definitive answer. Of course, this is largely due to the irritating vagueness surrounding the definition of both terms. However, through the rest of this post, I will try to highlight what I consider to be some of the main differences.

A facetious definition could be "Cloud is what the cool Web 2.0 kids use, whilst Grid is for the old academics with their pipes and tweed coats". However, there is a grain of truth to this – there is an irrefutable overlap between Cloud and Grid computing but a stark difference between people who know what Grid computing is and people who have only heard of Cloud computing.

Both terms centre on the idea of viewing computing power as a service (Grid computing takes its name from an analogy with the electricity grid) supplied by a typically external provider. In both cases the end-user does not want or need to concern themselves with the actual hardware used by the provider. Grid computing aims slightly further than the Cloud by also pursuing the sharing of resources (computational, data or storage) between entities (often across organisational boundaries), whilst hiding the hardware and protocols used from the user.

One of the main drivers to the birth of Cloud computing was the need to scale Web applications up in response to sudden changes in demand – for example to cope with sudden news exposure. These upturns in demand can often be very short-lived, making it uneconomical for companies to purchase enough dedicated hardware to cope with peak demand. The solution provided by Cloud computing vendors such as [Amazon EC2](#) allows on-demand spawning of new servers to almost instantaneously deal with such surges. Grid computing is also designed to deal with the problem of peak demand, but in a slightly different way – it views processing requests as individual tasks to be dealt with on a large computing cluster (or clusters) with a batch job scheduler (for more discussion on this see the [RightScale blog](#)). This view stems from the traditional (and largely academic) HPC world where users submit long-running jobs and receive the results hours or even days later.

Another important difference is in terms of implementation: Cloud computing uses virtualization* to achieve a standard on which users can run their applications, whilst Grid attempts to bring heterogeneous platforms (both in terms of OS and hardware) together to solve problems. The use of virtualization allows Cloud computing to sidestep a whole host of issues that Grid computing has to contend with, such as the availability of software libraries on different platforms.

If we accept these as the main differences between Grids and Clouds, what does this mean for the future? Some analysts have argued that [Grids are dead](#) and that “Clouds are Grids done properly” or Cloud computing is “[the user-friendly version of Grid computing](#)”, but things are not as clear-cut as any of these statements suggest. This is an argument for another post, but consider the following: Is it possible to use Clouds within Grids? What about vice-versa? What about the issues that Grid developers have been grappling for years with (e.g. security, trust, SLAs) – how are they solved in a Cloud computing context?

Thanks to Craig Thomson, Kostas Kavoussanakis and Mike Jackson for discussion and comments.

* See also: [Blogs and Discussion: The Open Source](#) and IBM: [The stuff of clouds](#).

About the Author



Adrian is an Applications Consultant at [EPCC](#). He graduated from Heriot Watt University with a BSc Hons in Computer Science, before completing an MSc in High Performance Computing at EPCC. He is currently working on the BEinGRID project. He was a technical member of [FilmGrid](#), one of the first wave of Business Experiments. He also founded the open source project [diffxml](#), based on his undergrad dissertation.

Business Experiments in the Cloud ?

[Posted by Stefan Wesner](#) 12 March 2009.

After reading and hearing everywhere that “Cloud Computing/Storage/...” is to be the [successor of Grid](#)*, I was wondering if one could do “Business Experiments in the Cloud”, similar to the ones we did (and still do) for the EC project [BEinGRID – Business Experiments in Grid](#).

As well as requiring a clear business case for each pilot implementation, the Business Experiments also have to show why a distributed, cross-organizational solution is beneficial and why several service providers are needed. The offered services range from a thin abstraction layer provider (e.g. computing cycles or data storage) up to more complex and sophisticated services (e.g. licensing provider, product database provider) which are more in line with the coarse grained concept expected from a [Service Oriented Architecture](#) (SOA) solution.

For many of the delivered services that are not tightly coupled to a specific resource (such as a radio telescope) or internal data stores that cannot be exported due to legal constraints, cloud services could be used to realize these services. After all, most services do not care if they are provided using a physical infrastructure or a virtualized one. So from this viewpoint, yes, we could have: **Business Experiments in Clouds!**

Another key element of many of the experiments is that the collaboration with other companies in a [Virtual Organisation](#) (VO) is always a compromise between the potential benefit (e.g. cost savings, integration of additional expertise and information) and the associated risks (e.g. dependency on an externally controlled service, ...). The proposed solutions for increasing the certainty of service delivery by: binding providers to [Service Level Agreements](#) (SLAs); using semantically enriched service

descriptions; and implementing commercial quality security do not really fix the problem. An SLA can be violated (including intentionally); securing the transmission and controlling access does not prevent data being revealed through other channels and semantically described services may still be misleading. So the decision to collaborate is still based on an analysis of the risks versus trust in the service provider.

In some scenarios the trust is based on the obvious common interest of all participating parties in the VO, in others only penalties described in the SLAs provide the confidence that a provider will do its best not to violate the agreements. The SLA might also place constraints on how data must be treated.

This problem has nothing to do with the virtualization of resources, as it should not be seen on this level. The fact that one or more of the service providers may build on top of a cloud infrastructure does not matter. But are current SaaS models ready to support such scenarios?

A frequently mentioned problem of clouds is the lack of appropriate SLAs providing confidence in the provider. I do not think that this is a problem in general. Depending on what you want to be done externally, one needs to make an analysis of risks versus benefits, and for many cases – in particular if the providers are easily replaceable or if no real-time constraints leads to an overall failure in case of a delay – the implied SLA (Quality: Best effort, Penalty: No payment on failure) is sufficient. A typical approach used in Grid solutions is to have local resources and remote resources with standardized interfaces, thus making providers replaceable. This approach can also be applied to clouds (e.g. [OpenNebula](#) + [Amazon EC2](#)). In my view, the major aspects not addressed by cloud solutions are real collaboration scenarios beyond a consumer and provider relationship. Cloud approaches either provide a virtualized infrastructure or deliver Software (or “Everything”) as a Service (SaaS). The scenarios considered in Business Experiments in Grid always have more than two clear stakeholders in the scenario. In order to realize such scenarios the Cloud/SaaS provider might be part of such a Virtual Organisation, but this would need infrastructure for VO Set-up, potential quick replacement of providers in case of failures and so on, very similar to that found in current Grid technology. So, for this aspect, our hypothetical cloud version of BEinGRID is now more like: **Business Experiments in Grids partially delivered by Clouds!**

The problems found in many discussions about clouds (e.g. look at [Above the clouds](#)) apply in a similar way to collaborative business Grids as realized in many past, or ongoing, service-oriented Grid projects such as [BEinGRID](#), [BREIN](#), [IRMOS](#), [NextGrid](#) or [Akogrimo](#). I think combining both worlds using virtualization approaches for the provision of services and aiming for the delivery of complete solutions in a SaaS/XaaS model using clouds with a standardized interface could contribute to the reliability of provided services. The cloud could provide parts of the VO Management elements, such as Creation of Instances on Distributed Hosting Environments as identified on [Gridipedia](#). However, in this case, cloud is not a replacement of Grids, but a replacement of how certain services within a Virtual Organization are provided. The [Grid is dead](#)? No! But as Virtual Organizations delivering a collaborative business scenario are far more complex compared to outsourcing scenarios, they are – as of now – mostly used in business to business scenarios and need to justify the increased effort.

So in conclusion, our Business Experiments in Clouds is fine if it is more a client-server relationship we are considering, but in truth we need a Business Experiments in Grids (maybe partially delivered using the cloud) if the target is a collaborative scenario involving many business partners in a non-hierarchical relationship. So, the question is not “Clouds or Grids?” but “How to integrate Grids **and** Clouds?”.

Notes

*See for example : <http://cloudcomputing.sys-con.com/node/771947> and <http://www.faz.net/s/RubB3C76B7D0C6444579C67AF441E2ECB6D/Doc~EB01457881F4B422CA2857E8817E57936~ATpl~Ecommon~Scontent.html>

About the Author



Stefan is the deputy director of the [High Performance Computing Centre in Stuttgart](#) and has worked in the area of distributed systems for more than ten years, starting with CORBA based systems and early Grid solutions such as UNICORE 1.0. He contributed to early efforts to move Grids towards a Service Oriented Architecture as well as business aspects of Grids whilst a member of the [GRASP](#) and [TrustCoM](#) projects. These projects developed the concept of the Virtual Hosting Environment and Abstract Entity, which are quite similar to Cloud approaches. He technically coordinated the Integrated Project Akogrimo, which looked into Mobile IPv6 Network, Network- and Grid middleware integration using virtualisation approaches. He is currently involved in this area as technical coordinator of BREIN and the principal investigator for the High Performance Computing Centre in BEinGRID and IRMOS.

Regular Contributors

Stefan Wesner - High Performance Computing Centre in Stuttgart, Germany

Santi Ristol - Atos Origin, Spain

Theo Dimitrakos - BT, UK

Dora Varvarigou - National Technical University of Athens, Greece

Mark Parsons - EPCC, UK

Rosa M Badia - Barcelona Super Computing Centre, Spain

Horst Schwichtenberg - Fraunhofer SCAI, Germany

If you would like to get involved with the blog please contact [Daniel Field](#).



Copyright © 2008 BEinGRID, ALL RIGHTS RESERVED. | [Disclaimer](#) | [Sitemap](#)