From Internet to Cross-Organisational Networking

Lutz Schubert[a,1], Alexander Kipp[a], Stefan Wesner[a]

[a]HLRS – University of Stuttgart, Nobelstr. 19, 70569 Stuttgart, Germany

Abstract. The Internet has become a powerful means of communication and interaction and various research projects have shown its potential to revolutionize business models and means of cooperation. Only recently, development has made significant progress in catching up with research and a series of products have been exposed to the market which may well represent the next step to realize this revolution.
This development will allow flexible resource and capability sharing across the net, as if the according capabilities would be locally available – even though this is already possible in principle, new models will allow maintenance of resources & capabilities on an operating system level, making it completely transparent to the average user.
This paper will show how the market is currently changing to host a new range of operating systems and collaboration support that will give rise to complete new capabilities, business models and communities, but at the same time will have us rethink classical approaches to problem solving. The paper will therefore examine recent research approaches to so-called Virtual Organisations and how they contribute to realizing new collaboration modes. It will show how major IT vendors are approaching this vision and where the current development may lead to, and how this will influence future business models.

Keywords. future internet, platform as a service, collaborative networks, service oriented architectures, virtualisation

1 Introduction

The Internet is no longer just a means of sharing data and information: with the increase in bandwidth and hosts, it has become a new form of resource itself. With the advent of the Grid, respectively more recently of Web Services, it has become possible to use and share application logic, code and local resources *programmatically* over the web. This shifts the need for resource availability away from the actual organization wanting to perform specific tasks to any host available on the web, i.e. a form of outsourcing over the Internet. Say for example that company Y needs to acquire more computational power in order to complete a specific calculation in time – since the advent of computational Grid (see e.g. EGEE [8]) it is possible for scientists to use distribute computational resources in order to get their results in time, without having to spend money on buying

[1] Corresponding Author Email : schubert@hlrs.de

additional computing systems which they may not need as part of their common, day-to-day business. We will explain in section 1 of this paper how Web Services and the concept of Virtual Organisations have realized new business models.

However, the concept of exposing resources and capabilities over web services / the net is taking up only very slowly – up to day, only a few thousand web services [15] exists that are openly available on the internet and only a few of them are actually of commercial interest, whilst most are provided by academic research or communities, similar to open source tools. Even Amazon and Google, ranking to the most popular commercial organizations providing web service, still offer their services cautiously and with a higher interest in research than in commercial exploitation.

As with any new development in the market, the problem behind this lack of commercial support is a mixture of both supply and demand: providers will not want to go through the effort of exposing capabilities via the web with no obvious demand for it and with ongoing problems in resource usage accounting, whilst potential customers do not yet see the benefit of troubling themselves with writing applications for such remote resources when there are still so few commercially interesting capabilities available which are furthermore difficult to retrieve. Whilst an experimental, research driven transition is certainly a valid approach to increase the interest in the consumers, the main problems are thereby not addressed: remote resource usage is still complicated, interoperability issues hinder simple integration, required capabilities / providers are difficult to find and security breaches, respectively resource misusage are difficult to prevent.

Efforts undergone e.g. by IBM and SUN, as well as by standardization bodies to reduce this problem by introducing standard means for interoperability, protocols for resource account, security strategies have not yet impacted upon the community as was originally hoped for. Framework support that intends to cover the full problem scope, as realized e.g. by Globus, Unicore or gLite, is still cumbersome to use and has hence not found the uptake necessary.

Only recently, a complete new set of base capabilities has been published, that opens up a complete new range of possibilities for future internet based interactions and cooperation. In section 2 of this paper we will examine these technologies, such as Google Apps, Force and in particular the programming foundation .NET3.5 by Microsoft, which, with its WCF, provides a means of realizing future platforms tightly integrated with and across the Internet.

Section 3 will show how such future platforms could be devised, what they will look like and how this will revolutionize the classical ways of enacting Virtual Organisations across the Internet, respectively to realize collaborative setups in a complete new, dynamic and community-like fashion.

Finally, in section 4 we will show how such new business models may find explicit usage in the domain of concurrent engineering, being one of the most resource demanding, and complexity driven domains that may hence even be considered a testing milestone for complex infrastructures. This will also show up restrictions of the models, as well as outstanding work to be performed in this area.

2 Service Oriented Collaborations: Virtual Organisations

The principle of the internet allows consumption of resources across the web, in particular of simple data sets, such as text, multimedia etc. As Grid, ASP and Web Services have shown, however, the internet can go further than that and provide actual application logic over the web: server hosted code can be executed upon requests, just like JavaScript upon opening the website.

With the increasing bandwidth and processor speed it was thus principally possible to generate cross organisational business processes over the Internet where each task is represented by individual parties exposing the according capabilities (e.g. in the EU project GRASP [1]). This led to the concept of electronic Virtual Organisations, where business entities sell their resources, ranging from individual business logics to devices and human capabilities with the according interfaces exposed to the web.

This concept allows business entities to enhance their individual capabilities with resources they do not own themselves but can access and integrate over the web, whilst other entities can make better use of free resources by selling them over the internet. We leave security & contractual issues aside here, as they would go beyond the scope of this paper – please refer e.g. to [20] [21] [24] for more details on business requirements in Virtual Organisations.

The particular advantages of this approach consists in (a) increased control through additional supervision mechanisms and (b) the capability to principally set up, adapt and destroy such collaborations on demand. In particular from the customer perspective, this allows making use of resources upon time of actual need and thus, which is more, making the customer more independent from individual providers, as they may be principally replaced dynamically during runtime so as to maintain constant business logic execution. It is hence in the interest of providers to allow for simple and stable integration, in order to maintain competitiveness.

At the latest the integrated project TrustCoM [11] sponsored by the European Commission has shown how these capabilities can be used to build up complex collaborative networks that respect the business requirements per participant and thus allow business entities to extend their capabilities in a stable and contract managed manner. These collaborations undergo a lifecycle from finding the appropriate collaboration partners over operation of the VO down to the termination.

2.1 Lifecycles of Virtual Organisations

Virtual Organisations and thus most of the current Grid research projects aim at provisioning of resources, capabilities and services across the *full lifecycle* of a collaboration. This means, that a VO middleware will completely replace the necessity to (1: Identification) identify interaction partners according to the collaborative goal, (2: Formation) configure them in order to grant secure access etc. (3: Operation) execute distributed business processes across these participants, potentially requiring reconfiguration of the collaboration in order to address resource failures etc (Evolution), and (4: Dissolution) finally to shut down the VO again, thus ensuring that the resources are no longer accessible outside the collaboration (cf. Figure 1).

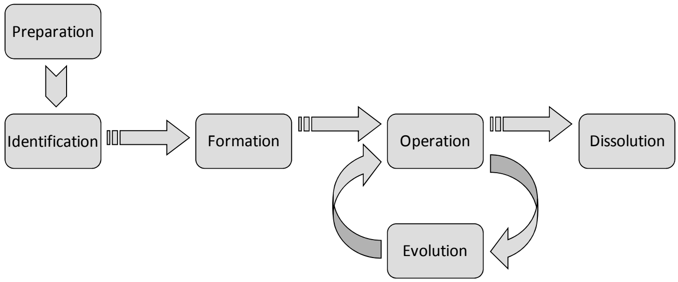

Figure 1. Lifecycle of Virtual Organisations

Addressing the *full* scope of real business collaborations obviously exceeds the (current) capabilities of the internet infrastructures, even considering legal restrictions (see e.g. [25]). However, whilst it is improbable that collaborations in the near future will use *only* web based resources, the principles are still relevant just for individual interaction partners. This means, that VO middlewares can take over the full support of managing *stable* web based resource access – we will come back to that specific point in later sections.

Readers interested in Virtual Organisation specifically, should refer to the TrustCoM framework [2].

2.2 Drawbacks

Non-regarding the big advantages of Virtual Organisations, the concept did not catch on to the degree originally hoped for – a particular obstacle consisting in the complexity of the steps to be undertaken in order to prepare and use the resources across the web. This applies in particular to the additional requirements towards security and control mechanisms to maintain the dynamicity and privacy aspects of such collaborations.

VO supporting middlewares so far do not cater for the fact that each service provider will have their own way of describing and exposing their capabilities, thus leading to confusions when trying to integrate the respective resource and thus when writing the overall collaboration description. This obstacle will not be overcome by standardisation effort, but instead must be addressed through semantic interpretation of “intention” (i.e. capability) vs. “expression” (interface-description).

2.3 Next-Generation VO Support

Recent VO related projects, amongst them BREIN [12] try to address these drawbacks by providing more intuitive management of provider issues, in particular with respect to their individual business goals and means to realise them: by enhancing both the resources, as well as the overall service provider with agent-capabilities, that enable them to align themselves and take cooperative decisions without requiring human interaction – this aims mostly at minor decisions with in particular no legal implications.

BREIN deals in particular with resource management problems in distributed environments, i.e. where jobs can be dynamically hosted on different machines and resources – be that restricted to local management (e.g. for scheduling bus resources in the airport) or more globally, where service provider expose in particular computing resources (in particular for distributed engineering tasks). In either case, failures in schedule execution – be it due to resource shortage, delays etc. – are difficult to deal with in a dynamic fashion. With the dimensions of internet based collaborations constantly increasing, and hence the scope gaining in complexity, such management will become too complicated for the average resource provider.

In such environments, the agent enhancements allow for self-monitoring of the resources so that failures can be easily recognised and potentially compensated (cf. Figure 2). Currently, BREIN can demonstrate such self-managing resources in the context of bus scheduling at the airport: buses are treated as independent agents that can communicate their capabilities and availabilities – jobs (schedules) are no longer painstakingly assigned for each resource, but published to all resources, that can then negotiate who takes over the respective assignment, depending on (a) the relevance of the respective job (here in particular costs due to non-fulfilment) and (b) the impact of execution on other jobs of the respective resource.

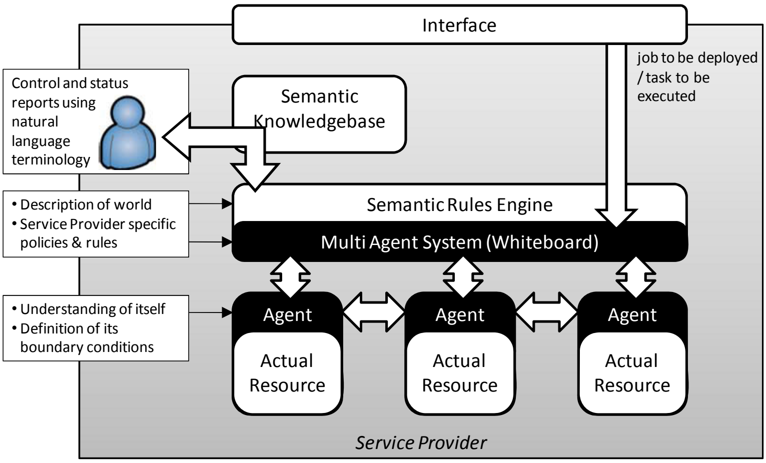

Figure 2. The BREIN enhanced service provider offering more abstract

In such a simple example, similar resource management take place across different providers (across two types of bus managers) in a similar fashion, though typically on a higher level, i.e. not per individual bus, but for a set of tasks.

3 New Approaches

Most approaches to realising Virtual Organisation suffer from the same drawback: supporting different messaging formats through standardised interfaces. Unless commercially supported, these solutions will not be able to cope with the changes and advances in research and on the market, and thus not find the uptake necessary to boost the product in the first instance.

Only recently, new approaches in Service Oriented Architectures and Infrastructures have been devised, which could well introduce such a change in approach and in particular commercial awareness towards this problem: Cloud computing, Platform as a Service and WCF.

3.1 Cloud computing

Amazon was one of the first companies to expose computational resources in a Grid like manner over Web Service interfaces (Amazon EC2 [1]) – this allows deployment and execution of various jobs on remote computers, just as foreseen in particular in computational grids, such as EGEE [8] and which commonly find application in particle physics (see e.g. [10]) and related areas. As opposed to the VO approaches, however, Amazon does not control any cross-resource interactions, as necessary for distributed applications, nor does it itself compose complex services into a single interface, i.e. Amazon does not provide abstract "products" [9],but simple computational resources in a restricted manner.

As such, it is e.g. not possible to make use of these resources in a similar fashion local resources could be used, as the classical Grid vision foresees it. However, with the Amazon programming interfaces, these resources could be used as a means to execute code that is generated for the particular purpose of remote hosting – Code Providers (rather than service providers) thus may generate sets of applications / tools that could be hosted *on demand* on Amazon like cloud computing networks, thus reducing the costs for providing and administrating such machines.

3.2 Platform as a Service

One step further than Amazon EC2 go Salesforce' "Platform as a Service" [4]: it allows not only hosting of prepared machine images complying to the Amazon specifications, Salesforce provides a platform for developing and hosting complete applications. This allows complete web server hosting capabilities, as opposed to pure computational power. Whilst this is principally comparable to classical Web Server hosting, Service Platforms significantly reduce the development time and in particular the management overhead, in particular regarding scaling, load balancing, security etc.

Google has shortly followed on this approach with its "Google App Engine" [5] and there are rumours that Microsoft will soon join in with a Cloud Service called "Red Dog" [6].

3.3 .NET 3.5 Framework

One of the most interesting developments in recent years, however, has been the release of Microsoft's .NET Framework, and in particular its introduction of the following enhancements:

- The Windows Communication Framework (short WCF [7]): WCF provides a new breed of communications infrastructure built around the Web services architecture, providing secure, reliable, and transacted messaging along with interoperability.
- The Windows Presentation Foundation (short WPF [16]): WPF provides a new presentation system for building visual client applications. The Extensible Application Markup Language (XAML) being part of the WPF is a declarative language with flow control support allowing the creation of visible UI elements in the declarative XAML markup, and then separate the UI definition from the run-time logic. This allows the transmission of visual user interfaces e.g. via web services without the need to consider the underlying service infrastructure.

4 Putting it All Together: Tomorrow's Internet

When looking at the technologies that have become available over recent years, one can note in particular the following main developments:

- web servers (hosts) have become more accessible and available
- computational resources (and thus machines) are offered over the web
- communication frameworks maintain a stronger Service Oriented Architecture approach
- full platforms become available as part of the internet
- applications and web services merge
- semantic enhancements
- the internet bandwidth increases constantly

If this development is pursued further and will finally be merged in future technologies, we face the brink of a new, internet wide infrastructure that allows data, information and code exchange in a complete new way, exceeding the aims of Grid Services, though not reaching the envisaged stability and reliability. This development has also been noted by the European Research Community by initiating the FIRE initiative [23] to support advanced networking research coupled with large-scale experimentation in order to find solutions to overcome the shortcomings of the current Internet architecture.

In the following sections we will sketch this new platform in more detail:

4.1 A Vision of Cross-Web Infrastructures

In principle, local computers have become obsolete: in most cases, a simple browser-like interface of a thin client is sufficient to fulfil most daily tasks, given that the according services are hosted on a remote machine. However, such interaction is still tiresome, due to the lack of connection speed and due to "incomplete" interfaces that do not provide the look & feel of local, complex applications.

This may change soon: computational power and in particular storage are already subject to outsourcing across the web (cf. above). Web Services and the new .NET framework allow easy transaction not only of simple commands, but also of full interfaces, as well as the generation and execution of complex code on the fly. Semantic searches enable interface abstractions and standardisation efforts ensure common messaging, even for complex data structures.

Thus, it is principally possible to host rich applications on remote machines with just deploying interfaces locally. With standard formats for application descriptions (XAML) and the convergence of messaging protocols between different layers of usage (namely between applications, web forms and web services), any application can principally be hosted remotely with the form being represented locally – the only obstacle: speed, even though the actual code execution may increase through stronger (remote) processor power, the actual interaction speed decreases due to bandwidth limitations (cf. Figure 3).

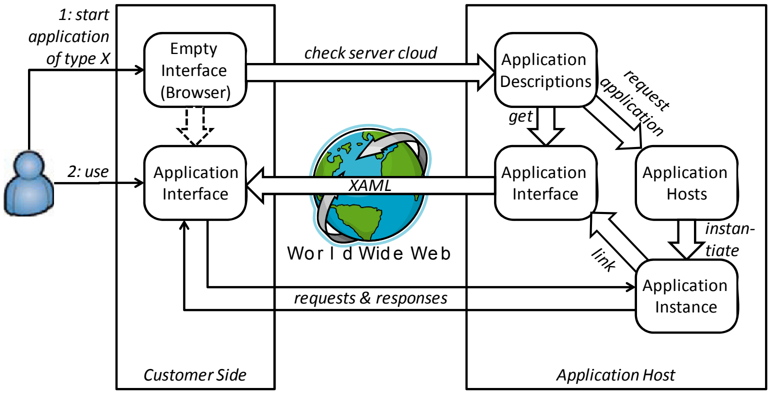

Figure 3. remote access through dynamic local interfaces

In order to overcome this problem, *partial* deployment is the keyword: we need to distinguish not only between application types that are either computing or interaction intensive, but also between code parts with the same characteristics. As such, interaction intensive code should be hosted locally whilst computational code could run remotely or even in a distributed fashion not only to increase performance speed, but also to increase stability (cf. below).

The attentive reader will notice the similarity to the concept of Virtual Organisations discussed above: by splitting up the main functionality into individual components and distributing them across the web, strengths (and expertise) of the individual providers can be exploited optimally, thus leading to better performance, stability and reliability of the overall system.

Microsoft again has undertaken a first step in this direction: with the new .NET framework following strongly the service / component oriented approach comes a new model of software interpretation: the “Just in Time” (JIT) compiler does no longer compile all code ahead of time and stores it as a self-running executable, but compiles the *text-based* code at the time of need and in a partial fashion: only currently relevant parts are compiled and stored in memory, whilst rarely used features remain uncompiled until time of need – unexpectedly, the result is highly performing.

A similar step is taken by Microsoft’s “SoftGrid” [19]: a server farm with preinstalled and preconfigured applications – independent of the application type – allows domain users to access and use these applications on the fly. Instead of just an interface, SoftGrid transfers the whole application to the user – however, in a time- and work-efficient manner, i.e. only the parts of the application are transferred that are currently of need. The commonality is obvious: segmentation of code bits into “relevant” and “less relevant” features that could be executed locally and remotely.

Let us take that one step further and distinguish between the sensitivity of data, how critical correct execution is etc. and we will come to Distributed Managed Platforms.

4.2 Distributed Managed Platforms

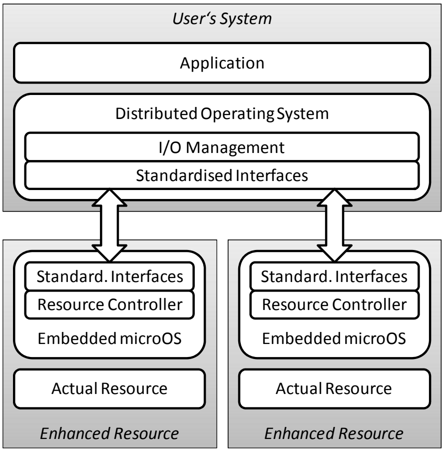

Figure 4. enhanced hardware resources to build up a distributed OS

Additional aspects, such as privacy, reliability and quality determine the platform of choice for individual tasks – managed platforms, such as Salesforce [4] and the Google App Engine [5] allow for replicated, secure execution of code on a remote server farm. Whilst this does not yet meet all requirements, it shows that the concept of Virtual Organisations and its according business requirements (see e.g. TrustCoM [11]) still hold true, but will have to move to a new level, closer to actual execution layer of operating systems.

Let us again go one step further and apply this concept on the operating system, and thus the most relevant platform from the user perspective. By linking parts of the execution environment on a low level, remote platforms could be used as part of the local infrastructure and thus all code execution managed in a similar fashion to current multi-core processors (cf. Figure 4) – a similar approach towards enhancing operating systems was already foreseen by Andrew Tanenbaum [26]. With the current development on both the hardware and software market, this vision is on the verge of becoming a technical, commercial solution.

4.3 A New Community Model

In this environment, the "Prosumer" [13]is taken to a complete new level, introducing new ways of doing business, but also of general interaction across the web: not only server farms will become more relevant in the future again, but also hosting of data, code, as well as provisioning of storage and computing resources will become more easy than ever before.

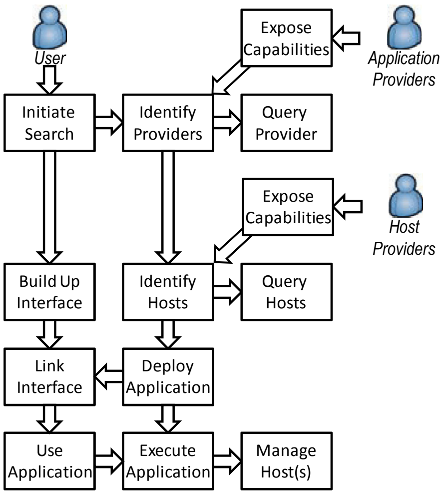

Figure 5. the process of distributed managed application use

With the semantic enhancements of Web 3.0 [14], the actual usage may look as follows: a user initiates a local search engine to look for all applications that provide a specific feature, rather than having a specific name – similar to current Desktop Search engines, but with support for semantic enhancements. The actual search is executed across the web and retrieves all remote applications satisfying the user constraints and definitions. Upon execution, interaction intensive code bits are transferred to the local machine, whilst in particular computational intensive and data critical code is distributed across a server farm, which may actually consist in community members offering parts of their local infrastructure. Depending on relevance of data and related aspects, code and data may be replicated, maintained on a single platform, load balanced across different servers etc.

Notably, such a web oriented platform community will realise more of the initial ideas associated with "the Grid" [22] than any other approach so far. And the technological basis is already there:

4.4 Realisation

Due to lack of space, this section can only outline the basic realisation approach: the architecture of the WCF framework (see [17]) already provides most of the details relevant for cross-organisational communication on a peer-2-peer basis and across different operational layers on the operating system. However, it is mainly restricted to Windows platform machines and as such not performing well in particular on a low device near level, as would ideally be granted in order to realise the framework. Almost all current operating systems however come at the high cost of providing a huge infrastructure that significantly reduces computational power, available storage etc.

Key to the new technology will be the development of embedded micro kernels as a basis for more complex operating systems. Current micro kernels (such as [18]) do not fulfil this purpose and are in fact replacements for full-fledged operating systems with no higher-level instance to integrate them.

5 The Future of CE

Now where is the impact of that for future concurrent engineering tasks? After all, the technologies described above aim particularly at realising a *low level* means of integration that seems at first sight to be of no interest for the coordination of high level, complex tasks as addressed by CE.

As already mentioned in the introduction, the area of concurrent engineering is one of the most demanding for distributed task management – one specific aspect thereby being stable, reliable and secure execution of multiple complex computational tasks. Particularly for design and analysis, a lot of effort is vested in realising and executing HPC machine code that then needs to be carefully maintained so as to ensure correct execution.

The future of the internet will not make HPC obsolete, even though followers of the SETI and BOINC movement may claim so: as HPC relies on much faster data transfer then ever possible over the web, distributed p2p approaches are only valid for optimally parallelisable codes, i.e. with weak data exchange between nodes. However, with a micro kernel per node and for the overall cluster, management of HPC tasks becomes ever more simple: as the micro kernel combination will be unaware of the actual distribution of nodes, but only of their requirements and availabilities, any changes in the infrastructure – be it due to failing nodes or other circumstances – will go unnoticed by the overall execution, thus allowing smooth transitions.

On a further, higher level, this applies similarly to all code-based application across Virtual Organisations which transform to a mixture of EGEE like community model and TrustCoM like business collaboration. With further enhancements, such as the ones envisaged by BREIN, a collaborative network could be realised, in which node management, security and reliability come implicit with the system.

Whilst the future internet will therefore mainly contribute to stronger interaction models, distributed management tasks on top of it, such as BREIN's VO concept become more simple and therefore realistic: current VO approaches spend more time on coping with interoperation and low level resource management issues than on actually realising management strategies, as the low level platform has not yet been realised.

6 References

[1] Wesner, S.; Serhan, B.; Dimitrakos, T.; Randal, D.M.; Ritrovato, P. & Laria, G. Overview of an Architecture Enabling Grid Based Application Service Provision, 2nd Across Grid Conference 2004

[2] Wesner, S.; Schubert, L. & Dimitrakos, T. (2005), 'Dynamic Virtual Organisations in Engineering', Notes on Numerical Fluid Mechanics and Multidisciplinary Design

[3] Amazon Elastic Compute Cloud (Amazon EC2) - Beta. Available at: <http://www.amazon.com/gp/browse.html?node=201590011>. Accessed on: April 8th 2008.

[4] On-Demand Business Application Platform and Programming Language. Available at: <http://www.salesforce.com/platform/>. Accessed on: April, 11th 2008

[5] Google App Engine. Available at: <http://code.google.com/appengine/>. Accessed on: April, 4th 2008.

[6] Mary Jo Foley. Red Dog: Yet another unannounced Microsoft cloud service. Available at: <http://blogs.zdnet.com/microsoft/?p=1324>. Accessed on: April, 9th 2008

[7] Windows Communication Foundation. Available at: <http://www.microsoft.com/net/wcf.aspx>. Accessed on: March, 3rd 2008.

[8] Enabling Grids for E-sciencE. Available at: <http://public.eu-egee.org/>. Accessed on: March, 17th 2008

[9] Haller, J.; Schubert, L. & Wesner, S. Private Business Infrastructures in a VO Environment. In: Paul Cunningham & Miriam Cunningham, ed. Exploiting the Knowledge Economy - Issues, Applications, Case Studies. 2006, 1064-1071.

[10] LHC Computing. Available at: <http://public.web.cern.ch/public/en/LHC/Computing-en.html>. Accessed on: March, 3rd 2008.

[11] TrustCoM - EU IST Project (IST-2003-01945). http://www.eu-trustcom.com.

[12] BREIN - EU IST Project (IST- 034556). http://www.gridsforbusiness.eu.

[13] WikiPedia - http://en.wikipedia.org/wiki/Prosumer. Accessed on April, 15th 2008

[14] Times Online – Web 3.0. Available at - <http://technology.timesonline.co.uk/tol/news/tech_and_web/the_web/article2726190.ece>, Accessed on April, 15th 2008

[15] SOA4All. Available at: <http://www.soa4all.org/>. Accessed on April, 15th 2008

[16] Windows Presentation Foundation. Available at: <http://msdn2.microsoft.com/en-us/library/ms754130.aspx>. Accessed on April, 15th 2008

[17] WCF Architecture. Available at: <http://msdn2.microsoft.com/en-us/library/ms733128.aspx>. Accessed on April, 15th 2008

[18] L4 Micro-Kernel Family. Available at <http://os.inf.tu-dresden.de/L4/>. Accessed on April, 15th 2008

[19] SoftGrid. Available at <http://www.microsoft.com/systemcenter/softgrid/default.mspx>. Accessed on April, 15th 2008
[20] L. Schubert, S. Wesner, T. Dimitrakos. Secure and Dynamic Virtual Organizations for Business. Paul Cunningham & Miriam Cunningham, ed., Innovation and the Knowledge Economy: Issues, Applications, Case Studies, IOS Press Amsterdam, 2005, pp. 1201 - 1208.
[21] S. Wesner, L. Schubert, T. Dimitrakos. Dynamic Virtual Organisations in Engineering. 2nd Russian-German Advanced Research Workshop on Computational Science and High Performance Computing, March 14 - 16, 2005.
[22] OGSA, OGSI and GT3. Available at <http://gdp.globus.org/gt3-tutorial/multiplehtml/ch01s01.html>. Accessed on April, 15th 2008
[23] FIRE Initiative. Available at <www.cordis.europa.eu/fp7/ict/fire>. Accessed on April, 15th 2008
[24] K. Stanoevska-Slabeva, C. Figà Talamanca, G. Thanos and C. Zsigri: Development of a Generic Value Chain for the Grid Industry. Lecture Notes in Computer Science, Volume 4685/2007. Springer, Berlin 2007, pp. 44-57
[25] Arenas, A. E.; Wilson, M. D.; Crompton, S.; Cojocarasu, D.; Mahler, T. & Schubert, L. (2008), 'Bridging the Gap between Legal and Technical Contracts', *IEEE Special on Virtual Organisations – in print*.
[26] Tanenbaum, A. S. (2001), *Modern Operating Systems*, Prentice Hall PTR, Upper Saddle River, NJ, USA.