On the Assessment of the S-Sicilia Infrastructure: A Grid-Based Business System

Carmelo Ragusa[1], Francesco Longo[1], and Antonio Puliafito[2]

[1] Cometa Consortium research group at University of Messina,
98166, S. Agata, Messina
[2] Department of Mathematic, University of Messina,
98166, S. Agata, Messina
{cragusa,flongo,apuliafito}@unime.it

Abstract. The enablement of the Grid paradigm for commercial solutions is a fundamental issue for both research bodies and companies. Different technologies seem to be mature to serve the purpose, but still research and experimentation is needed. In fact, the introduction of the business aspect poses new challenges such as new business models, service composition, relationships management, etc. In this paper, we present the S-Sicilia project, a 2-year collaboration between Oracle and the COMETA consortium, aiming to setup a Grid-based business infrastructure to provide business services with guaranteed QoS for companies. The system does not aim to provide answers for all business related issues, but rather be a kind of benchmark where experimentation on real cases takes place.

Keywords: Grid, Business Grids, SLA, SOA, Web Services, B2B, B2C.

1 Introduction

Business Grid promises the wide adoption of economic valuable Grid services. Lot of effort is being spent by the research community as well as by companies that are interested in its adoption. The business component involves more stringent requirements in terms of security, confidentiality, trust, guarantees etc. Moreover, by its nature a business process requires most of the times interactions with other business processes and therefore the Business Grid has to provide service composition. Also, regulations of Business-to-Business (B2B) and Business-to-Consumer (B2C) interactions have to be performed through Service Level Agreements (SLAs) which need a management system that deals with those contracts. Researchers are now focusing on those business aspects, trying to address some of the new arising challenges namely business models, pricing models and market economies. Also, some of the current Grid middlewares (Globus [15], Gria [16], Unicore [17], gLite [9]) are including some of the mentioned aspects within their solutions.

In this paper we describe a real case study, the S-Sicilia project [2][3], which aims to create real services for SME companies with specific needs. The project does not pretend to address all aspects related to a Business Grid but, by solving particular requirements, be a sort of benchmark to reach a global business-based Grid adoption.

J. Altmann, D. Neumann, and T. Fahringer (Eds.): GECON 2008, LNCS 5206, pp. 113–124, 2008.

The Grid infrastructure we started with is primary used for scientific applications with no QoS guarantees (i.e. gLite-based). We have created a business layer on top of this Grid infrastructure in order to provide business services with guaranteed quality. Service demand being in most cases unpredictable, the ability to scale the system with it can be a winning factor for SME companies, which normally have limited budget to spend compared to large enterprises.

In the next section we set the background of our work. In section 3 we describe the S-Sicilia project, the system architecture and a reference scenario. The assessment of the scenario is also discussed. Section 4 shows further scenarios that will be developed in the future. Finally, in Section 5 we give the conclusions and the next steps of our work.

2 Background

The business element adds complexity to the Grid approach. Although, some of the requirements in this context might coincide with the ones in scientific Grids, in business scenarios they are more strict. Research is focusing on all those aspects that will bring the Grid paradigm to the next level. In particular, a business oriented Grid infrastructure needs to address the following issues:

- **Distributed data management:** a solution providing support for integrating and updating data residing at multiple transactional resources. For example a distributed query and transaction facilities approach can serve the purpose;
- **Service composition:** services will tend to provide specific features addressing particular needs. Composition of such services across multiple domains is fundamental. Web services and Service Oriented Architecture (SOA) enable this important aspect;
- **Security:** the composition of services through different administrative domains gives rise to new security issues. A business Grid system must guarantee that the execution of a program on behalf of the vendor does not violate security policies, especially when the program has been supplied by the vendor or some delegated third party. New security policy languages allowing the definition of security driven service composition mechanisms are required;
- **Privacy:** user's data has to be protected through an efficient access management system;
- **SLA management:** a Business Grid enables the delivery of services as utilities and provides means for meeting SLA commitments for such services. Tools for service design to handle the definition of QoS contracts in SLAs have to be developed;
- **Accounting and billing:** currently, accounting and billing arrangements for outsourced services are based on raw machine resource consumption (CPU-time, storage capacity etc.). A Business Grid has to define a framework that allows accounting and billing in terms of the services that were completed, taking into consideration the QoS provided;
- **Business models:** new business models have to be developed to make business Grid services effective. Different approaches to business relationships and new business roles need to be experimented. This will create new source of revenue;

- **Trust:** relationships in economic environments might be long term based, but also very short and dynamic. Trust is definitely more solid and easier to achieve in the former than in the latter. Qualification information on suppliers to select business partners, clear and transparent contracts, security, monitoring of business operations can enable trust at all levels;
- **Risk assessment:** companies relying on business Grid services need to have tools to support their choices. Assess the risk of such choices can make a difference. The provision of specific tools is very important. Research of this aspect is therefore required;

Projects such as EGEE [18], NextGrid [28], Globus [15], Akogrimo [30], GRIA [16], GRASP [29], TrustCom [27] and AssessGrid [35] have all focused to address specific Grid business concepts, resulting in the development of new components, architectures or more generically guidelines. On the other hand the GridEcon project [25][26][36] aims to provide a generic framework addressing all aspects.

Gridipedia [19] defines itself as a Grid meeting point for individuals and organizations on the use of the Grid technology to address their business needs. The case studies section gives an overview of real cases that have successfully adopted the Grid technologies in different areas. Improvements in terms of resilience, performance, scalability and flexibility have been achieved within the financial, pharmaceutical and engineering sectors. Examples of commercial providers of business Grid are Digipede [20], DataSynapse [21], Sun Grid [22], Platform [23], EC2 [31]. The difference with research ones is that their solutions usually do not involve several providers and are based on single domains. Also, being commercial solutions they do not provide the inside and mechanisms of how services are implemented.

In the next section we introduce the S-Sicilia project. As said, our work is based on the gLite Grid middleware, which does not provide most of the functionalities mentioned. On the other hand, the aim of our system is not to address all Business related issues, but on delivering services with guaranteed quality. Aspects such as SLA management, accounting and billing, service composition and security are relevant.

3 Experimenting with Business Grids: The S-Sicilia Project

The S-Sicilia project, a 2-year collaboration between Oracle and the COMETA consortium [1], intends to setup a Grid-based business infrastructure able to provide services for industry. SMEs can benefit from Grid-based economy by reducing their costs to setup and run new services and hence increasing their competitiveness. On the other hand, large enterprises can take advantage from this new marketplace too. In fact, while normally large enterprises possess in-house data centers to serve normal work load, they could upload extra load to business Grids in case of peak demands.

An e-service is usually defined and managed through SLAs (Service Level Agreement), service contracts between a service provider and a service consumer. Those contracts specify parties' commitments, obligations, violations, performance levels and price.

In our system SLAs are based on the WSLA schema [4]. Its main characteristic is the ease of creation of SLAs via an XML schema that can be modified accordingly to the necessity. The schema structure also matches the SLA management tasks implemented within the system. At the time of the project start, this was the most appropriate choice. However, WS-Agreement [32] has gained consensus over time resulting in a recommendation from the OGF and therefore a future step will be to migrate the framework from WSLA to WS-Agreement.

The system manages services by monitoring the relative contracts and taking any due action such as service re-configuration, service re-location or resources re-allocation. Decisions are taken according to the customer's SLA and compensations are given in case of contracts violations. This known mechanism should give the right level of guarantees for business customers.

Other aspects are also addressed by the system such as accounting and billing. In this business environment, customers are billed not for raw resources as in common scientific Grids but for the service used. In order to provide such level of abstraction, contracts are specified in business terms, using terminology closer to customers.

Although, as stated in the background section, other Grid middlewares are focusing on business aspects, the use of the gLite middleware is compulsory to us. Therefore, the S-Sicilia project extends gLite basic functionalities by adding a business management layer to offer business services. However, the system has been designed to do not rely on a particular middleware, but to be rather generic.

In the next sections we present the details about our solution, describing the system architecture and illustrating the first scenario we have setup. An initial assessment of this scenario is also discussed.

3.1 System Architecture

Our system is based on different technologies. In particular, we have used a SOA approach to create a set of services dealing with high level requirements, and a Grid infrastructure for the low level resources.

The system is logically divided into two layers: the first and higher layer deals with customers, processing their requests, providing possible offers and managing SLAs; the second and lower layer interfaces with the Grid infrastructure. A set of QoS mechanisms have been implemented at both layers, in order to provide functionalities that the Grid does not offer natively.

The higher layer is called SLA engine and handles the service lifecycle. In particular, it manages service definition, deployment, monitoring, compliance and termination. Customers' requests are received in the forms of SLAs and are processed accordingly. A check for availability is performed and consequently an offer is sent back to the customer who can accept or reject it. Accepted SLAs are monitored to check their compliance and in case of needs actions are taken. As mentioned, the SOA approach was used and in particular the SLA engine is composed of a set of interconnected services providing the contracts' lifecyle management functionalities as well as billing and payment processes. The tools used to build this layer are the ORACLE SOA suite [5] and the ORACLE DB 10G [6]. The Oracle SOA Suite is a complete set of service infrastructure components for building, deploying, and managing SOA applications. Services can be created, managed, and orchestrated into composite applications and business processes.

The SOA suite contains the following packages:

1. The Oracle BPM (BPEL Process Manager), which offers a comprehensive and easy-to-use infrastructure for creating, deploying and managing BPEL business processes [7];
2. The Oracle ESB (Enterprise Service Bus), which provides everything for seamless integration of data and enterprise applications within an organization and with trading partners [8];

Figure 1 shows the system that has been implemented. We have used the BPM to create a BPEL process that combines together the small services represented in the figure:

1. **Accounting Service**: this service registers the user and grants the access to the infrastructure;
2. **Negotiation Service**: this process is started by the user that requests an offer for a service. The *negotiation service* gets the SLA from the user and submits it to the *performance prediction service*. Based on the result from the *performance prediction service*, the strategy and the pricing model in use this service fills the SLA and sends the completed offer back to the user;
3. **Performance Prediction Service**: it returns an estimate of the raw resources that can potentially satisfy the SLA, based on the history of similar SLAs stored within the DB;
4. **Process Scheduler Service**: this service receives the user agreed SLA and stores it in the DB;
5. **SE Uploader Service**: this service allows the user to upload its application files and stores the files paths, within the SE (the gLite Storage Element [9]), in the DB;
6. **Monitoring Service**: this service monitors the SLA lifecycle. It checks if there are SLAs in a new state and sends them to the *QoS service*. It also detects if an SLA needs attention and informs the *QoS service*. Finally, it determines when an SLA finished and contacts the *billing service*. The operations performed by this service are also implemented by some procedures within the *Service Repository*;
7. **QoS service**: this service performs a match operation of the user requests with the Grid resources. It also deals with SLAs that need attentions by taking due actions;
8. **Billing Service**: this service gets the finished SLA, invokes the credit service and informs the user that his/her service has been completed;
9. **Credit service**: this service invokes an external service, usually offered by a bank, which debits the user credit card or account;
10. **JDL maker service**: this service invokes a script that creates a JDL (i.e. gLite Job Description Language) file for the SLA and sends it to the gLite User Interface (UI);

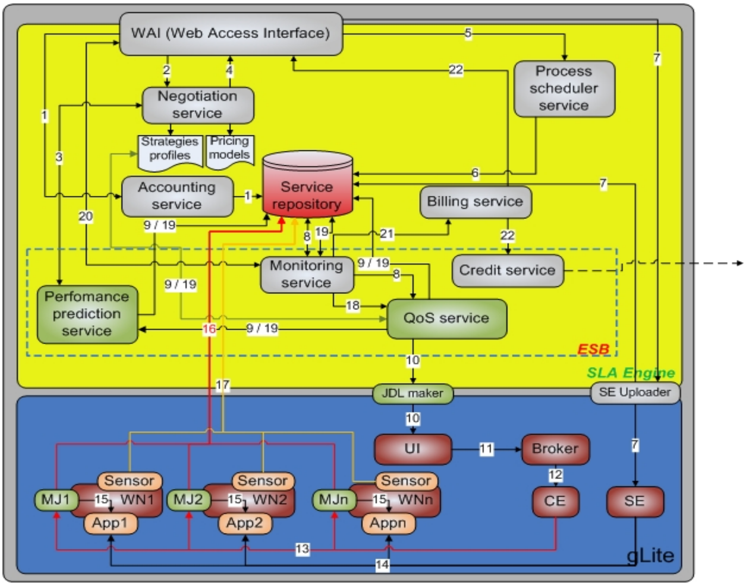

Fig. 1. S-Sicilia system architecture

One of the key element of the system is the *Service repository*, which is used by the SOA suite internal components, the SLA engine processes and the resource *sensors*. Users credentials, data and SLAs are stored in the *Service repository*. A native XML DataBase (DB) was needed, due to its higher performance in managing/storing XML data. Oracle DB 10g provides such feature. The *Service repository* structure was designed to accommodate generic applications, so that each time a new application has to be supported it can be easily integrated into the system. An application is composed by a main process which is supplied by the user, a set of services needed to run the application and a DB. Services have dependencies in terms of other services. Those dependencies are stored in the *Service Repository* as well. The main process as well as its services have to maintain some information needed to run such as addresses, ports, files, etc., which are also kept within the *Service repository*. Resources runtime consumption data is stored in specific tables to provide the current and past state of the system. Finally, an archive of the past SLAs is maintained. The *Service repository* contains also a number of stored procedures implementing the logic to instantiate and manage applications and services. Some of those procedures are exposed to external entities that need to interact with the *Service repository*.

Finally, gLite is the Grid middleware used and it can be seen as an integrated set of components allowing resource sharing. It provides a framework for building grid applications exploiting the power of distributed computing and storage resources across the Internet.

In order to integrate the SLA engine with the gLite infrastructure, a set of resource *sensors* has been created. These have been implemented and deployed in the gLite

Worker Nodes[1] (WNs). Their task is to timely measure resource instant consumption and send those measurements to the *Service repository*. *Sensors* have been implemented as linux cron jobs and scripts measuring computing (CPU and RAM), storage and network resources.

3.2 Scenario: Web Applications Hosting Service

Web applications have gained popularity over the past years, due to benefits they bring such as rapid development, scalability, user mobility. Also the possibility to have an initial low investment has given the chance to small businesses to get online and therefore extend their range of actions. However, with low investment only small infrastructure can be built which somehow limits the company chances to grow. On the other hand if a higher investment is considered it may be too risky if the success is not fully accomplished. Giving the companies the possibility to ease their activities from managing their IT-system and potentially lowering their investment will be indeed welcomed.

This scenario concerns a hosting solution for a hardware vendor's web application. Customers connect to the vendor's website and make their orders. Normally, those orders go to the vendor's system, running on an application server, that processes them. We have created an environment to host such application by installing application servers and DBs on the Business Grid nodes (see Figure 2). In this way all

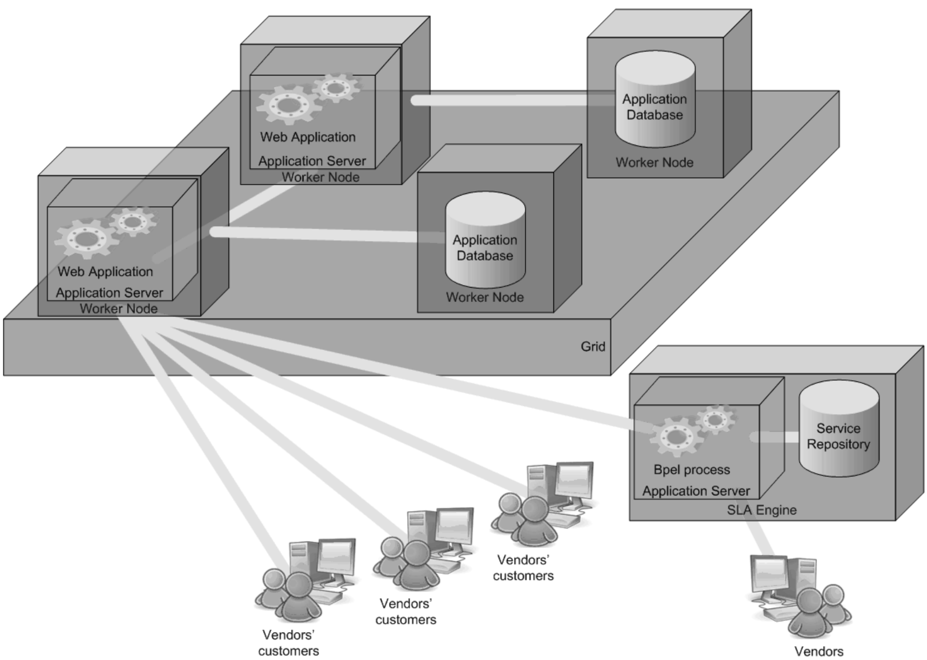

Fig. 2. Web hosting scenario representation

[1] A gLite Worker Node (WN) is a single unit of computation, containing a set of clients needed to run jobs.

customers' orders are processed by the vendor's system running on the application servers hosted on our Business Grid solution.

This scenario allows experimenting with such type of businesses to show the ability of the proposed system to cope with them.

The ORACLE SOA suite comes with a demo application, which is a commercial application for a hardware vendor webssite implemented through a BPEL process. This demo was chosen for our scenario.

A potential online vendor accesses the infrastructure through a web interface by filling a form with his/her requirements. The system verifies that the service can be performed and sends an offer. After accepting the offer, the vendor sends his/her application in a WAR/EAR format along with the database scripts that are needed to create the necessary DB tables and procedures. Those files are then stored on the gLite SE. The *monitoring service* detects the new SLA and informs the *QoS service*. The *QoS service* chooses the resources (i.e. gLite WNs) to run the service based on the indications of the *performance prediction service*. The deployment phase then starts by performing an installation of the vendor's application, a DB creation and relative configurations in a fully automated way. After the deployment phase, *sensors* are attached to the application and its services. Those *sensors* then send data read to the *service repository*, which are used by the *monitoring service* to assess the SLA compliance during its lifecycle. If current WNs are not enough to guarantee the SLA requirements a dynamic workload balancing is performed by the *monitoring* and *QoS services* adding more worker nodes as needed to fulfill the SLA constraints. Vice-versa, if an SLA, considering its actual load, has assigned too many WNs an action to release the WNs that are not necessary is performed.

3.3 Scenario Assessment

The first test we carried out aimed to assess basic functionalities of the system. The vendor's requirement was to support a maximum of 300 transactions per second. Initially, the vendor's application runs on a single worker node (WN1). We setup the network threshold on WN1 of 150 KB/s, after which the system detects that the SLA does not have enough resources and an action has to be performed. We have simulated accesses to the vendor's website using a tool called JMeter [10].

In the first phase we simulate traffic for 10 concurrent users performing an average of 7 operations each. WN1 is able to take on the all load, according to the threshold we setup.

In the second phase JMeter increases the number of concurrent users to 40, in order to overload WN1. As a consequence, the *monitoring service* spots this situation, understand that the involved SLA has rights for more resources and informs the *QoS service*. The latter chooses another WN to install the vendor's application and re-start the deployment phase. Following the steps previously described, when this phase completes there are 2 WNs to serve all users' requests. The traffic towards the vendor's website is then divided between WN1 and WN2.

Finally in the third phase JMeter reduces the traffic by simulating again 10 concurrent users. In this situation there is an overallocation of resources. The *monitoring service* captures this new state and again informs the *QoS service*. The *QoS service* then starts an undeployment phase where the vendor's application is uninstalled from

WN2 along the associated *sensors*. WN1 is able to address all users' requests to the vendor's website. Table 1 shows the data of the total network traffic sent and that relative to WN1 and WN2. Those data are also graphically reported on Figure 3.

Table 1. Network traffic data of Web application hosting scenario

Time (mins)	Total Network Traffic sent (KB/s)	WN1 Network Traffic sent (KB/s)	WN2 Network Traffic sent (KB/s)
1	0	0	0
2	0	0	0
3	0	0	0
4	51.04	51.04	0
5	52.46	52.46	0
6	223.46	223.46	0
7	273.1	219.65	53.46
8	251.06	145.63	105.43
9	239.99	113.54	126.45
10	246.3	118.76	127.54
11	245.34	109.87	135.46
12	180.94	86.57	94.37
13	115.28	76.53	38.75
14	56.37	56.37	0
15	59.88	59.88	0
16	51.24	51.24	0

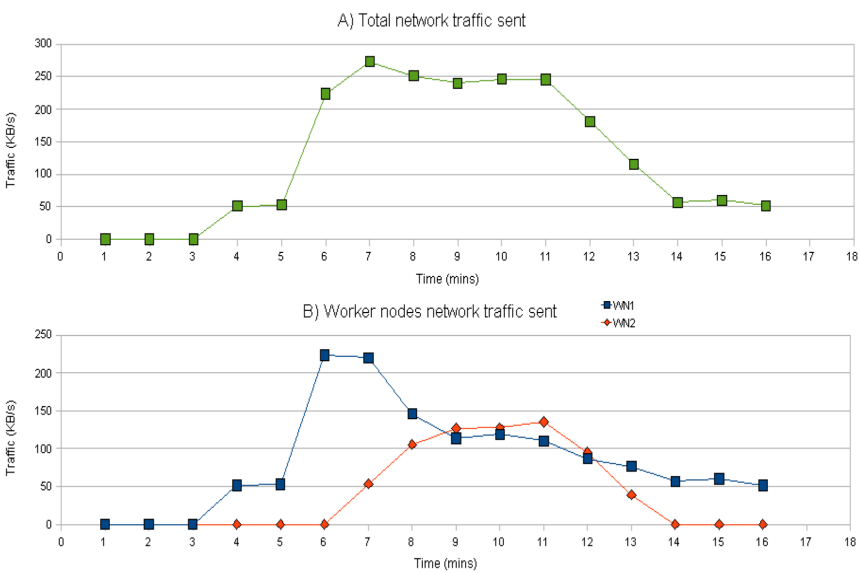

Fig. 3. A) Total network traffic sent in reply to vendor's website customers' requests. B) Traffic sent by WN1 and WN2.

As we can see, initially until minute 5 (phase 1) WN1 serves all requests to the vendor's website. After minute 5 starts phase 2 until minute 11. Here the traffic grows more than the threshold (150 KB/s) and more resources are added to serve users' requests, according to the vendor's SLA. From minute 6 to minute 12 WN2 is active and serves users' requests along WN1. In fact, the total traffic is split between the two nodes. At minute 12 starts phase 3 where the users' request decreases drastically and the vendor's application is removed from WN2. The situation then returns to its initial state.

Although only a simple scenario was used we can say that the system behaves correctly and delivers QoS services. Obviously, more tests will be needed for which more complex scenarios will be setup. For example, if traffic is very bursty, deployment and undeployment phases will be quite frequent. Tests to understand those situations will be needed.

4 Future Scenarios

Further scenarios will be implemented in the future. One of the goals of the S-Sicilia project is to show the flexibility of the system to support very different applications. A part from the Web hosting scenario which will be further developed to add more complexity, other applications will be integrated within the system.

We have already implemented a virtualization solution that runs on our gLite based infrastructure. Virtualization has a lot of advantages, because allows setting up customized environments tailored to the users' needs. For example, a user might have specific requirements for its application in terms of operating system, or his/her application might be a legacy system with specific requirements. Virtualization is the perfect choice in those situations. Also new range of applicability of the virtualization approach are being studied. Different companies showed interest in this approach and have provided some application to experiment.

INTERGRAPH [11] is providing SmartPlant Foundation SPF (Engineering Database), SmartPlant P&ID (CAD 2D), SmartPlant Instrumentation and Marian (Material management and procurement). ST Microelectronics [14] will provide its software suite for microchip design and simulation. Other agreements for experimentation have been reached with INSIRIO [12] and INQUADRO [13] to supply further applications. In the future, the virtualization solution will be integrated within the S-Sicilia infrastructure.

5 Conclusions and Future Work

This paper presented the S-Sicilia project. Issues related to Business Grids were discussed. New business models, service composition, SLA management, security, trust, privacy are some of those issues. The relative projects conducting research on those aspects were also mentioned. Our system aims to deliver solutions targeting SME companies with specific needs and can be a point of reference for the development of the business mechanisms that will lead to the global adoption of the Grid paradigm.

We have presented the system infrastructure. The system sits on top of a Grid infrastructure based on the gLite middleware. The Grid infrastructure is mainly used for scientific applications, but we have extended it to provide business solutions. The business layer is based on a SOA approach. Basic services composing our system were introduced. Addition of *sensors* at gLite level were also introduced. Users access the system through web interface, establishing a contract containing the service requirements, parties commitments, service performances, terms of service violation and service price. A web hosting scenario was also shown. In this scenario we offer a service that allows a web vendor to have his/her web application hosted on our infrastructure. An initial assessment of the basic functionalities to deliver the user's contracted quality was discussed.

The future steps of our work will be to implement some of the logic behind the scenes. For example, we are working on a mapping mechanism that will allow to get the raw resources based on high level requirements. Tests are being carried out for the purpose. Moreover, we want to increase the complexity of our web hosting scenario. We want to start experimenting with an increasing number of concurrent SLAs and see when and where the system breaks. Improvements will be definitely needed.

Another important aspect that we are working on is the use of a different approach for the definition of SLAs. Currently we are using WSLA of IBM, but the adoption of WS-Agreement is under study. WS-Agreement [32] is getting more consensus from the research community. Recent works [33][34] have shown how it can be extended to allow service runtime re-negotiation.

Finally, we have discussed further scenarios through which we will be able to demonstrate the flexibility of our solution. Companies interested in those services have already agreed to provide their applications for experimentation. The implementation and assessment of those scenarios is also a priority for us.

Acknowledgments. This work makes use of results produced by the PI2S2 Project managed by the Consorzio COMETA, a project co-funded by the Italian Ministry of University and Research (MIUR) within the Piano Operativo Nazionale ”Ricerca Scientifica, Sviluppo Tecnologico, Alta Formazione” (PON 2000-2006). More information is available at http://www.pi2s2.it and http://www.consorzio-cometa.it.

References

1. The COMETA Consortium, Consorzio Multi Ente per la promozione e l’adozione di Tecnologie di calcolo Avanzato, http://www.consorzio-cometa.it
2. The S-Sicilia project, http://www.consorzio-cometa.it/s-sicilia, s-sicilia.unime.it
3. Oracle and COMETA Consortium Combine Approach to Grid Computing for Sicilian Pilot Project, http://www.oracle.com/corporate/press/2006_dec/cometa.html
4. Ludwig, H. et al.: Web Service Level Agreement (WSLA) Language Specification, http://www.research.ibm.com/wsla/WSLASpecV1-20030128.pdf
5. Oracle SOA suite, An SOA Suite of Oracle’s Most Popular Best-of-Breed Technologies, http://www.oracle.com/technologies/soa/soa-suite.html
6. ORACLE DB 10G, http://www.oracle.com/database/index.html
7. Oracle BPM, Business Process Management, http://www.oracle.com/technology/products/ias/bpel/index.html

8. Oracle ESB, Enterprise Service Bus, http://www.oracle.com/appserver/esb.html
9. gLite, User Guide, https://edms.cern.ch/file/722398/gLite-3-UserGuide.html
10. Apache JMeter, http://jakarta.apache.org/jmeter
11. INTERGRAPH, Leading global provider of spatial information management (SIM) software, http://www.intergraph.com
12. Insirio, Business Solution Integrator, http://www.insirio.it
13. InQuadro, Insirio Innovazioni, http://www.inquadro.it
14. ST Microelectronics, http://www.st.com
15. Globus, Globus Toolkit, http://www.globus.org
16. Gria, Service Oriented Collaborations for Industry and Commerce, http://www.gria.org
17. Unicore, Uniform Interface to Computing Resources , http://www.unicore.eu
18. EGEE, Enabling Grids for E-sciencE, http://public.eu-egee.org
19. Gridpedia, The European Grid Marketplace, http://www.gridipedia.eu
20. Digipede, Distributed computing solutions for real-word business problems at any scale, http://www.digipede.net
21. DataSynapse, The leader in Real-Time Infrastructure Software, http://www.datasynapse.com
22. Sun Grid, Sun Utility Computing, http://www.sun.com/service/sungrid/index.jsp
23. Platform, HPC Management Software, http://www.platform.com
24. Altmann, J., Ion, M., Mohammed, A.A.B.: Taxonomy of Grid Business Models. In: 4th International Workshop, GECON 2007, Rennes, France, August 28 (2007)
25. Altmann, J., Routzounis, S.: Economic Modeling of Grid Services. e-Challenges 2006, Barcelona, Spain (October 2006)
26. Altmann, J., Courcoubetis, C., Darlington, J., Cohen, J.: GridEcon – The Economic-Enhanced Next-Generation Internet. In: Veit, D.J., Altmann, J. (eds.) GECON 2007. LNCS, vol. 4685, pp. 188–193. Springer, Heidelberg (2007)
27. TrustCom, An environment for trust, security and contract management in B2B collaborations, http://www.eu-trustcom.com
28. NextGrid, Architecture for Next Generation Grids, http://www.nextgrid.org
29. GRASP, Grid Based Application Service Provision, http://www.eu-grasp.net
30. Akogrimo, Access to Knowledge through the Grid in a mobile World, http://www.akogrimo.org
31. Amazon EC2, Amazon Elastic Compute Cloude, http://www.amazon.com/gp/browse.html?node=201590011
32. Web Services Agreement Specification (WS-Agreement), http://www.ogf.org/documents/GFD.107.pdf
33. Di Modica, G., Tomarchio, O., Vita, L.: A framework for the management of dynamic SLAs in composite service scenarios. In: Workshop on (NFPSLA-SOC2007) in conjunction with ICSOC2007, Vienna (Austria) (September 2007)
34. Di Modica, G., Regalbuto, V., Tomarchio, O., Vita, L.: Enabling re-negotiations of SLA by extending the WS-Agreement specification. In: IEEE Int. Conf. on Service Computing (SCC 2007), Salt Lake City, Utah (USA) (July 2007)
35. Djemame, K., Gourlay, I., Padgett, J., Voss, K., Kao, O.: In: Buyya, R., Bubendorfer, K. (eds.) Market-Oriented Grid Computing. Wiley, Chichester (to appear, 2008)
36. GridEcon, Grid Economics Business Models, http://www.gridecon.eu