

# Service Level Agreements in Virtualised Service Platforms

Georgina GALLIZO<sup>1</sup>, Roland KUEBERT<sup>1</sup>, Karsten OBERLE<sup>2</sup>,  
Andreas MENYCHTAS<sup>3</sup>, Kleopatra KONSTANTELI<sup>3</sup>

<sup>1</sup>HLRS - High Performance Computing Center Stuttgart,  
Nobelstraße 19, Stuttgart, 70569, Germany

Tel: +49 711 68564823, Fax: + 49 711 68565832, Email: [gallizo@hlrs.de](mailto:gallizo@hlrs.de), [kuebert@hlrs.de](mailto:kuebert@hlrs.de)

<sup>2</sup>Alcatel-Lucent, Bell Labs, Lorenzstraße 10, Stuttgart, 70435, Germany

Tel: +49 711 82132169, Fax: + 49 71182132185, Email: [karsten.oberle@alcatel-lucent.de](mailto:karsten.oberle@alcatel-lucent.de)

<sup>3</sup>National Technical University of Athens, 9 Heroon Polytechniou str.,  
157 73 Zografou, Athens, Greece

Tel: +302107722546, Fax: +302107722569, Email: [a\\_menychtas@telecom.ntua.gr](mailto:a_menychtas@telecom.ntua.gr), [kkonst@telecom.ntua.gr](mailto:kkonst@telecom.ntua.gr)

**Abstract:** Interactive real-time applications, either for soft or hard real-time, require strong QoS guarantees which, nowadays are only achievable through distinct hardware and software. The ICT FP7 project IRMOS (<http://www.irmosproject.eu>) will unite two distinct worlds - Service Oriented Infrastructures (SOIs) and real-time applications - in order to cope with soft real-time requirements. For that purpose, IRMOS is developing an SLA management framework, as part of the IRMOS virtualised service platform, considering all involved parties through the whole value chain. This includes the mapping of high-level application parameters to low-level resource provision attributes of the virtual environment, having a key role in all steps of the application lifecycle (requirements definition, SLA negotiation, monitoring application execution and SLA violation detection). The IRMOS SLA management framework is being validated, among others, by an application scenario which provides a distributed work environment for virtual and augmented reality, namely COVISE - COLlaborative VISualization and Simulation Environment.

## 1. Introduction

A Service Level Agreement (SLA) is a contract, between the provider and the customer of a service, specifying the function performed by the service, the agreed bounds of performance, the obligations on both contractual parties and how deviations are handled [1], [2], [3], [4]. An SLA is made in some business context and therefore it must include all (strictly necessary) aspects of the context related to the provided service that are relevant to all interested parties. When the service involves multiple stakeholders, independent bipartite interactions may not be sufficient to cover all requirements to guarantee the end-to-end service provision. An SLA management framework must therefore support an end-to-end SLA negotiation, considering requirements ranging from high-level business requirements to low-level resource requirements.

There are various approaches in the field of SLA management in distributed environments. In many cases, SLAs are modelled according to business objectives of both customers and service providers as discussed in [7], [8] and [9]. Authors in [10] and [11] present approaches that deal with SLA management by providing Quality of Service (QoS) [3],[5],[6] guarantees at the same time. What is regarded of major importance refers to the provision of such guarantees for real-time interactive applications that are deployed and executed in virtual environments. In that framework, the ICT FP7 project IRMOS [12] will unite two, at the moment, distinct worlds: Service Oriented Infrastructures (SOIs) and real-

time applications. SOI provides a system for describing infrastructure as a service, which includes all configurable infrastructure resources such as compute, networking and storage hardware and software. Consistent with the objectives for SOA (Service Oriented Architecture), SOI facilitates the reuse and dynamic allocation of necessary infrastructure resources. This dynamicity complicates the prediction of system behavior, which is necessary for giving real-time guarantees on system response. For the provision of real-time applications over SOIs and since multiple stakeholders are expected to be involved in the value chain, a number of QoS requirements needs to be taken into account and guaranteed through the whole chain. An SLA management framework should consider these factors in order to ensure the provision of interactive real-time applications in SOIs.

## 2. Objectives

Interactive real-time applications, either for soft or hard real-time, distinguish themselves from normal applications in that they require strong QoS guarantees. Real-time guarantees are, nowadays, only achievable through distinct hardware and software and real-time applications are, in general, not delivered over SOIs.

This paper presents how the IRMOS project copes with soft real-time requirements as expressed by interactive real-time applications and consequently how to guarantee those, going from high-level service requirements as expressed by the customer (e.g. end-user or application provider) down to low-level resource requirements. This is accompanied by presenting an application scenario which provides a distributed work environment for virtual and augmented reality, namely COVISE - Collaborative Visualization and Simulation Environment [13].

For that purpose, as mentioned above, IRMOS is developing an SLA management framework covering all necessary parameters for the guaranteed provision of QoS in interactive real-time applications, considering all involved parties through the whole value chain. The latter includes the mapping of high-level application parameters to low-level resource provision attributes that will allow the expression of real-time requirements as a set of QoS requirements. This framework has a key role in all steps of the application lifecycle, from the requirements definition and the SLA negotiation to the application monitoring and SLA violation detection during the execution.

## 3. Methodology

Many actors are taking part in collaborative interactive applications when these are provided through SOIs. IRMOS has identified the following actors as part of the “generic” IRMOS value chain (the roles identified could also be split in more concrete actors based on the business model that is followed):

- **Client** is the final user of the service.
- **Application Provider:** Provides application as a service over the IRMOS platform. These applications are composition of more basic components: Service Components and Client Components.
- **IRMOS Framework Services Software Provider:** It is a software vendor that provides the IRMOS Framework Services layer to IRMOS Providers.
- **ISONI Provider:** ISONI (Intelligent Service Oriented Network Infrastructure) is slated to create entirely new market segments. The ISONI Provider virtualizes the infrastructure/resources offered by one or multiple Operators/Providers and allows an optimal resource sharing of computing, storage and networking resources and offers those to several customers in parallel, supporting real-time applications.

- **IRMOS Provider:** This role is the frontend to the IRMOS platform; it provides all services required by Application Providers to use the IRMOS platform. It embraces the ISONI Provider and the IRMOS Framework Services Software Provider.

Interactions between the adjacent providers deal with very different data which can hardly be encapsulated in one single contract. Therefore, it becomes necessary to split the value chain into different parts, including distinct contracts between them, which are modelled in IRMOS in the following SLAs: **Application SLAs** (between Clients and Application Providers), **“Static” SLAs** (between Application Providers and IRMOS Providers) and **Technical SLAs** (between IRMOS Providers and ISONI Providers).

The **“Static” SLAs** are baseline and long-term contracts which are rarely changed, allowing the hosting of the Application Provider’s application via the IRMOS Provider. This static long-term relationship is a prerequisite of the automated SLA negotiation process and does not have to be negotiated every time a new client uses an application over the IRMOS platform. During the SLA negotiation phase this SLA is considered as fixed.

The **Application SLA (A-SLA)** is a dynamic SLA between Clients and Application Providers. Its process is initiated by the client when a concrete service request is occurring. This SLA contains the high level QoS requirements of applications as defined by the client.

The **Technical SLA (T-SLA)** is also a dynamic SLA. It is negotiated between the IRMOS Provider and the ISONI Provider and contains low-level QoS parameters associated with the (virtualized) infrastructure, including the complete topology of the virtual environment (virtual machine units, storage elements, network links) in which the application will be deployed and executed.

The mechanisms mapping high level QoS requirements (part of the A-SLA) towards low level QoS requirements (part of the T-SLA) are provided by the “IRMOS Framework Services Software Provider”. These mechanisms are used by the Application Provider to collect information for the behavior of the application in order to optimize the parameter mapping in the SLA negotiation process.

The overall SLA framework needs to cover the following aspects:

- Different lifecycles for SLAs and services.
- Automatic SLA management through the complete value chain. This means that the part of the IRMOS architecture involved in the process needs - to some extent - the capacity to decide without human intervention about the SLA negotiation.
- Dynamic SLAs (i.e. SLA re-negotiation), allowing modification of reserved resources.
- Modeling, within the different SLAs, the requirements corresponding to different technological levels between value chain actors.
- Real-time constraints in the whole SLA Management process (including e.g. discovery, negotiation and monitoring).
- Detection of SLA violations in both at application- and resource-level and real-time management of the resources to avoid failures.
- Specific QoS parameters for interactive real-time applications should be included in the Technical SLA, covering different types of (virtualized) physical resources (e.g. for the network resources QoS parameters may be bandwidth, jitter, delay; for the computing resources the parameters may be CPU, RAM; etc).

The challenges in realizing the complete SLA framework are manifold: firstly, the framework exists on two levels (A-SLA and T-SLA level) and each one must provide the abovementioned functionality. Secondly, the complex functionalities and interactions, distributed among different project partners, lead to major integration work. Thirdly, real-time aspects need to be taken into account during all developments. IRMOS addresses these challenges by continuous integration of produced results and different demonstrators at clear intervals which will implement an ascending level of functionality.

## 4. Scenario Description

Real-time requirements, both for resources like processing power and storage as well as for network resources, are often found in the area of collaborative working and virtual environments. COVISE [13] was developed at the High Performance Computing Center Stuttgart and consists in a prime example of a highly demanding collaborative application. COVISE enables the collaboration of distributed people in the visualization of simulation results. The augmentation of real-world input with the simulation data is also possible. Naturally, this poses extreme requirements to the infrastructure. The solutions to either bring all persons into one central place for collaboration or for providing dedicated hardware are what is normally done today. This is expensive, either in time – potentially long travels – or, additionally, in money – expensive hardware or travel expenses. By ensuring that high demand on the infrastructure and timely constraints for collaborative working can be satisfied even between distributed users, IRMOS facilitates collaboration and, at the same time, the reduction of expenses. The approach presented will be a huge step forward for collaborative working and will provide application providers as well as end-users with dynamic access to guaranteed resources.

## 5. Developments

The Virtual and Augmented Reality scenario, through the COVISE application, has been chosen as one scenario to validate the concepts developed within the IRMOS project.

The first step for the validation of the proposed SLA management framework has been the analysis and identification of the functional and technical requirements of COVISE application, from a high-level perspective, as expressed by the customer (e.g. end-user or application provider). To find out these application requirements first the different phases of the design process, the instantiation and execution, as well as the roles of the involved stakeholders had to be identified. For this purpose, the application has been analysed in detail with selected use cases, as described in [14].

Several application-agnostic use cases have been identified, as shown in Figure 1, which must be performed for all applications intended to run on IRMOS. These use cases describe tasks to be performed by an actor to authenticate towards the IRMOS platform, specify requirements of the application affecting resource selection and interconnection of resources in IRMOS and finally negotiation and acceptance of an SLA based on the given requirements. A final use case refers to the monitoring of the parameters during execution of the application, that are essential for real-time aspects to the application, as well as monitoring of application performance as far as this is supported by the components of the application, which could trigger the re-negotiation of SLAs to adjust resource parameters to the customer and consumer QoS requirements.

Application specific use cases have been also identified, as illustrated in Figure 2.

Each of these use cases imposes a set of requirements over the environment in which it is executed. The following aspects have been analysed to identify those requirements:

- Actors involved.
- Pre-conditions which must be met before its execution, such as available hardware, installed software, etc.
- Post-conditions given after the execution of the use case, such as hardware and software running, etc.
- Basic flow detailing the executed steps as part of the use case.
- Alternative flows to the basic flow.
- Special requirements and real-time requirements imposed by the application.
- Relationship with other use cases.

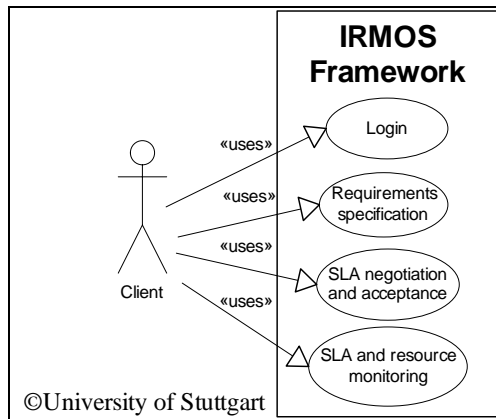


Figure 1: Overview Use cases for actor interaction with the IRMOS framework [14]

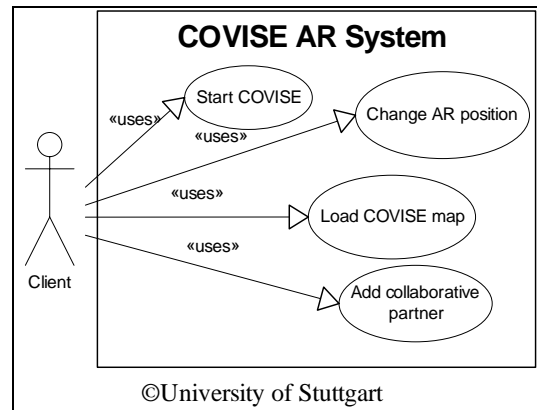


Figure 2: Overview Use Cases for actor interaction with the COVISE System [14]

One of the most relevant aspects for the research described in this paper is the one related to real-time requirements imposed by the application. These are requirements that must be met in order to guarantee the real-time behaviour of the application. A preliminary analysis on how these high-level (application) requirements are translated into low-level (resources) has been performed in [14]. For this purpose, the affected key parameters (as described in [15]) by the application are identified. These key parameters are related to resources, such as:

- Volatile Memory: includes parameters such as size, bandwidth, random access-time, and jitter.
- Processing: includes parameters such as CPU performance and GPU performance.
- Network: includes parameters such as latency, bandwidth, jitter and data stream synchronization.
- Other: includes parameters such as concurrency and uptime.

The association of the aforementioned high-level user requirements with the low-level resource parameters takes place in two phases: the application adaptation phase (offline phase) and the application usage phase (online phase). During the offline phase, the monolithic application is split into individual components; each one of them is adapted to the IRMOS environment and modelled following the IRMOS specifications. The models are produced through extensive benchmarks and describe the behaviour of the component for each use case. Additionally, templates for the A-SLAs and workflows are produced in which the application clients specify their high-level QoS requirements. In the online phase, the mapping mechanisms analyze - using the application models - these requirements, associating them with the low-level resource requirements. The output of the mapping process is the creation of a complete description for the virtual environment that includes the computational and storage resources for the application components, the network links between them and the low-level QoS characteristics and operational parameters of each required resource. Based on the mapping results a T-SLA request is created and sent to one or more ISONI Providers. As soon as an agreement is reached at technical level and QoS requirements can be met, both SLAs are signed and the execution phase is initiated.

The role of the SLA management framework is also important during the execution phase where the related monitoring service of the Framework Services collect data delivered from the infrastructure (ISONI) as well from the application itself in order to:

- check the application requirements against the real resource usage; this might result in application reconfiguration or an SLA re-negotiation if more or less resources are required than previously agreed in the SLAs during the lifetime of a service,

- check the agreed terms against the provided QoS to detect any A-SLA or T-SLA violation and perform possible SLA re-negotiations in order to keep the agreed QoS level and therefore to guarantee smooth operation for the application.

## 6. Results

As a result of the analysis of the COVISE use cases, the following results have been found out, as described in [14], affecting the different SLAs that will be involved in the application design, instantiation and execution:

*Table 1 COVISE Real-time requirements analysis*

| COVISE Use Case                                | Application real-time requirements   | Resource-related key parameters affected  |
|--|--|---|
| Start COVISE                                   | Timing for the start-up of COVISE is dependent to local system capabilities.   | not applicable (n/a)  |
| Change Position of Augmented Reality spectator | COVISE visualised 3D geometry is updated in real-time (and synchronised with the related frames of the AR video stream) to correct position and orientation on the screen or projection space as well as updating the data for RemoteAR collaborative endpoints. | <ul style="list-style-type: none"> <li>• Network Latency</li> <li>• Data stream synchronization</li> <li>• Number of concurrent access to the server</li> </ul> |
| Load COVISE map                                | Performance of loading a map is dependent on I/O - operations on the local system as well as to response times of hard- and software connected through network.  | <ul style="list-style-type: none"> <li>• Network Latency</li> </ul>   |
| Add collaborative partner                      | Connection should be established in a human-perceived reasonable time period.  | <ul style="list-style-type: none"> <li>• Network Latency</li> <li>• Bandwidth</li> <li>• Number of concurrent access to the server</li> </ul>                   |

From this scenario analysis, it is clear that not all value chain actors can be aware of the aforementioned parameters. On one hand, the ISONI Providers are application unaware, hosting, simultaneously, in their virtual environments, several applications (of different type and imposing different QoS requirements). In that way, they cannot agree on SLAs that include application-specific parameters. On the other hand, the application clients, who only “speak the application language”, require an environment that fulfils the application requirements and cannot understand the low-level resource parameters to set up the virtual environment. The IRMOS SLA management framework tackles this with two-level SLAs and mechanisms for end-to-end SLA negotiation and QoS provisioning. The T-SLAs include terms from a finite set of low-level parameters that affect the real-time application execution based on the modelling and mapping suggestions which also IRMOS Providers can understand. The A-SLAs terms are high-level and application-specific, and, as a result, dynamic. However, benchmark tests are required for all of them in order to achieve accuracy on the results of the mapping process. There is always a possibility of wrong estimations on the required resources but this is addressed “on the fly” by the SLA Management framework re-negotiating automatically the low-level resources to prevent SLA violations and malfunctions.

## 7. Business Benefits

The application of the IRMOS SLA management framework to the COVISE application, as well as its integration within the overall IRMOS framework will provide significant advantages over the current deployment.

It is widely accepted that revenues in the telecommunication as well as in the IT market can only be achieved and improved by providing an ever-increasing number of new attractive added value services to the customers with minimum time-to-revenue.

Service and application providers also need to control costs: from an operational expenditure (OPEX) perspective, this means being much more efficient in how they develop, test and launch new services; from a capital expenditure (CAPEX) perspective, it means reducing their dependence on the large Network Equipment Vendors that have historically locked them into service "silos" that can prove to be extremely expensive to onward develop new services.

The Framework being developed in IRMOS enables 'real-time' interaction between people and applications over a Service Oriented Infrastructure (SOI), where processing, storage and networking need to be combined and delivered with guaranteed levels of QoS. It is essential to provide an efficient overall SLA management framework allowing all involved parties ranging from infrastructure providers of all kind up to application providers to participate in the business chain. Especially the dynamic behavior of the SLA management solution is a key enabler of the IRMOS framework and gives added value to the developed SOI. Dynamic behavior allows the customer (e.g. application provider) a flexible expansion of deployed services with given guarantees and at the same time the infrastructure provider is enabled to optimize utilization of its own resources and, therefore, to create added value as well.

Other potential scenarios which could benefit from the proposed framework include those where:

- massive collaboration with real-time constraints is expected, such as e-learning applications with high numbers of users interacting with the application and with other users;
- high volumes of data need to be processed, supporting real-time interaction with end users, such as film post-production applications.

## **8. Conclusions**

The IRMOS project will develop a novel way of enhancing SOIs with real-time capabilities. IRMOS software services providers, which are located in the middle layer between applications and resource (network, computation, storage) providers, bridge the gap between high-level requirements and low-level resources. In order to support QoS throughout the whole value chain, distinct contracts between the value chain actors have been indentified and will be modelled through different types of SLAs. The above will allow the adoption of the proposed IRMOS framework in any heterogeneous and distributed environment (such as SOIs) that seeks to bring QoS knowledge with regard to real-time aspects within the SLA management process.

Initial steps have been carried out in order to implement and validate the IRMOS SLA Framework. The analysis of COVISE application specific use cases and the identification of an initial set of requirements, including affected resource-related parameters, have been performed. Further work, within the project next phase, will consider that set of requirements for the modelling of the concrete application components, the execution of the complete SLA negotiation process - in order to initialize a virtual environment with the appropriate resources - and the monitoring of the application execution, resulting in possible SLA re-negotiations to guarantee the required QoS levels. Software developments are under way and will result in a first demonstrator being realised at the end of the project's second year (February 2010).

## References

- [1] Kleopatra Konstanteli et al., IRMOS D2.3.1 State of the Art on IRMOS technologies - <http://www.irmosproject.eu/Deliverables/>
- [2] OGF GFD.107: "Web Services Agreement Specification (WS-Agreement)", A.Andrieux et al, March 2007
- [3] OGF GFD.120: "Open Grid Services Architecture - Glossary of Terms", J.Treadwell, December 2007.
- [4] "SLA Management Handbook: Volume 2 Concepts and Principles", Release 2.5, TeleManagement Forum, GB 917-2, July 2005.
- [5] ETSI TR 102 479: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Review of available material on QoS requirements of Multimedia Services".
- [6] ITU-T Recommendation E.800: "Terms and definition related to quality of service and network performance including dependability".
- [7] B. Mitchell and P. McKee, "SLAs A Key Commercial Tool", Exploiting the Knowledge Economy - Issues, Applications, Case Studies, eChallenges 2006.
- [8] Buo, M. J. Chang, R. N. Luan, L. Z. Ward, C. Wolf, J. L. Yu, P. S., Utility computing SLA management based upon business objectives, IBM Systems Journal, 2004,
- [9] A. L. M. Ching, Dr L. Sacks and P. McKee, "SLA Management and Resource Modelling for Grid Computing", Whitepaper, UCL, 2003
- [10] H. Chen, H. Jin, F. Mao, H. Wu, "Q-GSM: QoS Oriented Grid Service Management", Web Technologies Research and Development - APWeb 2005, Lecture Notes in Computer Science, 2005
- [11] Padgett, J., K. Djemame, and P. Dew, "Grid-based SLA Management", Lecture Notes in Computer Science, pp. 1282-1291, 2005
- [12] IRMOS Project, <http://www.irmosproject.eu/>
- [13] High Performance Computing Centre Stuttgart, COVISE <http://www.hlrs.de/organization/av/vis/covise/>
- [14] Wolfgang Huther et al., IRMOS D4.1.1 Definition and implementation of the three scenarios and its real time requirements - <http://www.irmosproject.eu/Deliverables/>
- [15] Eduardo Oliveros et al., IRMOS D2.1.1 Initial version of Requirements Analysis Report - <http://www.irmosproject.eu/Deliverables/>