

Workshop on Service Level Agreements in the Grid

Grid computing allows virtual organizations to share resources across administrative domains. In its early days, Grid computing was inspired by the need for transparent access to supercomputing resources and by the idea to even couple the resources in a metacomputing environment to create even more powerful computational resources. Currently the focus is on service-oriented architectures (SOA) where a wide variety of services from multiple administrative domains can be accessed by service clients. One of the most important tasks of current Grid middleware centers on efficient resource management. Resource providers offer their resource to virtual organizations and publish detailed information about the resources. Recent efforts have also focused on exposing computational and data resources as “services” – thereby providing a single abstraction that could be applied at different levels of software deployment. Based on this information appropriate resources for Grid applications are selected, and jobs are finally submitted to these resources. Service Level Agreements (SLA) are attracting more and more attention in Grids as a means to guarantee quality of service terms for grid applications and to enable the establishment of novel business models. A wide range of research and development questions have to be addressed in this context. This covers the creation of languages for formulating SLAs that are powerful enough to express the relevant QoS terms, but can also be used to automatically manage the negotiation, execution, and monitoring of SLAs. Brokering systems are required that can select resources for job execution based on the SLA templates offered by the resource owners. Scheduling algorithms that can optimize for different goals in the context of multi-item, multi-attribute, and multi-unit optimization problems are also necessary. Flexible local resource management algorithms are required for provisioning the resources at the provider’s side to meet signed SLAs. The seminar brought together people working on SLAs in the context of grid computing mainly from computer science, but also from information systems and application areas. These researchers come from different areas and bring in a wide range of research work. The topics covered by people on the invitation list include but are not limited to the following areas:

- Languages and protocols for creation of SLA
- Business models
- Grid economy
- SLA management
- Resource management
- Job scheduling
- Application deployment mechanisms
- Negotiation strategies

Agenda

The seminar included the following sessions:

1. SLA application
2. SLA implementations, technologies and approaches
3. SLA negotiation approaches
4. SLA policies and legal issues
5. Interoperability: Standards for describing and creating SLAs
6. SLA applications, status, monitoring and billing

In addition to working group meeting on WS-Agreement Profiles and on WS-Agreement-Negotiation of the GRAAP working group of the Open Grid Forum were held in conjunction with this seminar.

Open Questions

- Dynamic environments
 - How to efficiently support highly dynamic environments where SLAs need to be cancelled or re-negotiated often?
- Failures & responsibilities
 - Who defines failure – impact depends on the target and granularity of the SLA?
 - How to detect responsibility if the SLA is violated, e.g. in case of incorrect workflow data?
 - What if the results are incorrect but the SLA wasn't violated?
 - Who has the burden of proof, who has the benefit of assumption?
 - Agree in the SLA already what is considered as violation
- Monitoring of SLAs
 - Monitoring & user feedback should be included in SLA frameworks
- Negotiation
 - Support for many-to-many negotiation not in existing standards
 - Need for more than one negotiation protocol
 - Need a negotiation protocol description language as part of the infrastructure
 - Bootstrapping between different SLA environments to automatically support different templates, different SLA implementations, e.g. WSLA, WS-Agreement
 - Should use "Invitation to treat" instead of "request for (non-binding) quote"
 - Need extensions to WS-Agreement to steer the negotiation.
 - Negotiation & Renegotiation: which is a special case of which?
- Interoperability
 - How interoperability issues coming from different hosting environments can be handled efficiently?

- * Is it possible to identify a taxonomy of SLA terms so terms found in SLAs from different hosting environments can be classified?
- Term languages
 - * WS-Agreement is agnostic of the application environment
 - * Currently, JSDL is used as term language to describe the service terms in a scheduling/resource management environment
 - * More term languages are needed (and partly under work) for using WS-Agreement in other environments, e.g. when dealing with co-allocation of resources or services, software licenses, network properties.
 - * Need an agreed term language like EJSDDL to express economic data.
 - * Micro-specifications needed in WS-Agreement for interoperability
 - * Need agreements with malleable constraints, e.g. having terms with ranges or functions instead of fixed values
- Legal issues
 - Recommendations
 - * Specify governing legal system (but be aware that there are constraints when trading with a consumer).
 - * Use an overall governing (paper) contract before doing (electronic) SLAs (good place to establish payment processes, locations of access services, SLA negotiator services, ...)
 - * Be specific!
 - To be clarified/decided beforehand: Which party carries the risk
 - “Legally Binding”
 - * Term doesn’t actually help; better to think in terms of whether there is agreement between parties that are legally considered to be able to enter into contracts.
 - * SLAs can be used to modify a contract if there is agreement within the contract to do so; this is the real foundation of SLAs.
 - “Quote” vs “Invite to Treat”
 - * In English law at least, quotes are binding (within reason).
 - * Formally, want customer to make offer and provider to accept it. This is core of WS-Agreement already.
 - B2B vs B2C
 - * There are additional protections that apply to consumers to prevent certain types of abuse. The details are complex!
 - Prior paper agreement between the parties that the computer-generated SLA is binding (Pre-contractual phase between customer and service provider).
 - However, prior agreements not generally possible in dynamic environments
 - Delegation to a broker for further negotiation is also legally binding if there is an SLA between the broker and the end-user exits
 - Is using advanced certificates instead of progressive ones for signing SLAs beneficial for legal binding of SLAs?

- Do we need electronic SLAs with legal capability comparable to the paper ones?
- SLA Infrastructure
 - Need support for template discovery
 - Support for resource management systems negotiating SLAs?
 - Need automatic SLA template generation depending on resource status, e.g. what the provider can offer actually.
- SLA Management
 - What do we need to compare two SLAs?
 - Are SLAs understandable for users?
- General questions
 - Should SLA-Based environments try to meet an overall (most likely) measurable objective?
 - Which are the use-cases that need SLAs (and there is a large number which probably won't need SLAs).

We would like to thank the people at Dagstuhl for their excellent support, the inspiring atmosphere, and the easy collaboration. We again enjoyed very much having the opportunity for staying at Dagstuhl.