Using WS-Agreement for Risk Management in the Grid*

Matthias Hovestadt, Odej Kao, Kerstin Voß, Georg Birkenheuer
Paderborn Center for Parallel Computing
University of Paderborn
Germany
{maho,okao,birke,kerstinv}@upb.de

Karim Djemame, Iain Gourlay, James Padgett
School of Computing
University of Leeds
United Kingdom
{karim,iain,jamesp}@comp.leeds.ac.uk

Abstract

Risk is defined as "Hazard, danger, exposure to mischance or peril" [1]. Risk management is a discipline that addresses the possibility that future events may cause adverse effects [4] and is important in a diverse range of fields such as statistics, biology, engineering, systems analysis and operations research. Modern risk management recognises that, in contrast to risk avoidance strategies, accepting certain risks can also be beneficial. Examples include day-traders on the stock market and professional poker players, who aim to make statistically profitable decisions but with high variance [7].
Current state-of-the-art Grid computing does not incorporate risk management and current Grid infrastructures still follow a best-effort approach. This is insufficient for attracting commercial end-users to use the Grid. To clarify this, consider the following scenario. An end-user wishes to make use of Grid resources to run an application. In a typical scenario a user is looking to pay a Grid resource provider (or providers) to execute his or her application. This application may consist of a single job or a workflow with a number of sub-jobs. The end-user wants transparent access to the resources in the sense that the complexities involved in interacting with Grid middleware are hidden from them. Consequently, users negotiate for resource usage through a Grid resource broker which queries resource providers on their behalf to find suitable resources. They require a job execution with a desired level of priority and quality. For example they may have a deadline for the completion of their application which, if not met, may lead to financial loss. Consequently, users may want to negotiate for Service Level Agreements (SLAs) to define all aspects of the business relationship between themselves and the Grid resource provider(s) and specify the Quality of Service (QoS) that can be expected such as performance of the resource provider as well as a penalty fee which the provider has to pay if it does not perform as contracted. Accordingly, contracted performance and penalty fees are of particular importance when the user is paying for the resource usage.
Clearly in this scenario, risk management is of great importance to the end-user the broker and the resource providers. A number of research projects address the issue of SLA negotiation in a Grid environment [6, 3]. However, providers are still cautious on adopting such a system, since the agreement to meet the objectives specified in an SLA is a business risk. SLA violation can be caused by many events like network or resource failure or even operator unavailability. Without a means of assessing the risk of agreeing an SLA, providers are only able to make uncertain decisions regarding suitable SLA offers. Similarly, end-users would like to know the risk of an SLA violation by a Grid resource provider so that they can make appropriate decisions in relation to acceptable costs and penalty fees. A broker which is acting on behalf of a user to search for suitable resource providers may also require risk assessment mechanisms. Risk assessment enables the broker to evaluate the overall risk involved in mapping a workflow consisting of a set of possibly inter-dependent sub-jobs onto a number of resource providers. Further, the capability to

*This work has been partially supported by the EU within the 6th Framework Programme under contract IST-031772 "Advanced Risk Assessment and Management for Trustable Grids" (AssessGrid).

evaluate the reliability of risk assessments presented by resource providers is important. This new functionality will be valuable for avoiding contracts with unreliable providers and significantly enhances the service of a broker.

The goal of the AssessGrid project is to address the key problem of risk by introducing a framework for supporting risk assessment and management for all three Grid actors discussed here (end-user, broker and resource provider) [5].

The AssessGrid system architecture takes these three actors into account, and new system components are introduced in the architecture to support risk management. Risk assessment and management are not integrated into any contemporary Grid solution. The idea of estimating the risk of violating an SLA is an obvious consequence of current research topics. The importance of SLAs to Grid commercialisation has led to a drive to standardise SLA negotiation in the Grid. Within the Global Grid Forum (GGF), the Grid Resource Allocation Agreement Protocol (GRAAP) working group has been leading this work, and this has resulted in two draft standards for SLA description (WS-Agreement [2]) and SLA negotiation (WS-Agreement Negotiation). The WS-Agreement specification is a promising approach to provisioning services in an interoperable manner. Its negotiation features make it an appropriate framework to use in AssessGrid for resource providers to advertise resource capabilities and for users/brokers to make job requests. WS-Agreement is designed to ensure that any domain specific or other standard condition expression language can be used to define Service Level Objectives [2]. Risk can be incorporated into SLAs as additional attributes.

In the AssessGrid architecture, the end-user (supported by a GUI) can describe prerequisites for the jobs, such as hardware architecture, operating system, amount of memory, and libraries, etc. The user interface modifies the broker's/provider's SLA template based on this input and sends it to the Grid broker or resource provider in order to gather SLA offers. This communication will be realised using the WS-Agreement protocol [2]. The offers are returned to the end-user who can select the different SLAs and read their content, e.g. fee, risk of failure, penalty in case of failure, options to reduce the computed risk, and the operation's costs, etc. WS-Agreement [2] will be used to represent SLA conditions. However, extending WS-Agreement in the risk assessment and management domain first necessitates the definition of risk specific agreement terms. Thus, what needs to be defined is a term language to describe not only job requirements and attributes that can be negotiated by users/brokers and resource providers to reach agreements, but risk as well.

The broker negotiates with the provider on behalf of the end-user. It is responsible for enabling risk-aware negotiation for resource usage and application execution. The broker is deployed as a Grid service and the WS-Agreement protocol (WS-Agreement) [2] is used to establish SLAs. We can envisage a commitment level for the term *risk* which can take on the value *Observed* to indicate that both parties (user/broker or broker/resource provider) have committed to a value of the term and will operate in an agreement based on it. The broker's role is supported by a risk assessment module which contains a risk assessor and a workflow assessor with the functionality to compute risk assessments for workflow orchestrations. In addition the broker has access to a confidence service which provides statistics to enable the reliability of providers' risk assessments to be determined. The confidence service also has access to a database which contains information describing previous SLA offerings and fulfilment.

In the Grid fabric in which the provider manages its resources, reservation and allocation of corresponding resources are important to achieve the desired QoS. Accordingly, reliability, availability, cost, and performance as well as an estimate of the risk of failure of an SLA must be considered. A consultant service supports the provider's risk assessment methods with statistical information. Communication between the components within the proposed architecture is performed thanks to an adaptation of the existing WS-Agreement protocol [2], within which the resource provider can advertise its capabilities as part of its dynamic interface using dynamic service data. Also job submission port types may need to be defined by extending the domain-independent port types exposed in the WS-Agreement specification to support risk management.

References

[1] Oxford English Dictionary. http://dictionary.oed.com.

[2] A. Andrieux, K. Czajkowski, A. Dan, and et al. Web Services Agreement Specifi cation (WS-Agreement). In *Global Grid Forum GRAAP Working Group*, Sep 2005.

[3] K. Czajkowski, I. Foster, C. Kesselman, V. Sander, and S. Tuecke. SNAP: A Protocol for Negotiating Service Level Agreements and Coordinating Resource Management in Distributed Systems. In *Proceedings of the 8th Workshop on Job Scheduling Strategies for Parallel Processing*, 2002.

[4] F. Kloman. Risk Management Standards. March 1995.

[5] K. Djemame, I. Gourlay, J. Padgett, G. Birkenheuer, M. Hovestadt, O. Kao, K. Voß. Introducing Risk Management into the Grid. Submitted to the 2nd IEEE International Conference on e-Science and Grid Computing, Amsterdam, 2006.

[6] A. Sahai, S. Graupner, V. Machiraju, and A. van Moorsel. Specifying and Monitoring Guarantees in Commercial Grids through SLA. In *Proceedings of the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID'03)*, Tokyo, Japan., May 2003.

[7] D. Sklansky. *The Theory of Poker*. Gamblers Book Club, 1999.