Chapter 2
Flexible and Dynamic SLAs
Management in Service Oriented Architectures

Giuseppe Di Modica  
Università di Catania, Italy

Orazio Tomarchio  
Università di Catania, Italy

ABSTRACT
The increasing adoption of service oriented architectures across different administrative domains forces service providers to use effective mechanisms and strategies of resource management in order to guarantee the quality levels their customers demand during service provisioning. Service level agreements (SLA) are the most common mechanism used to establish agreements on the quality of a service (QoS) between a service provider and a service consumer. However, the proposed solutions have not been taken up by business stakeholders due to the low flexibility and usability together with the lack of interoperability. Any framework for SLA management should address several issues, such as SLA modeling and representation, SLA publication and discovery, protocols for establishing and negotiating SLAs, SLA monitoring and enforcement. This chapter addresses the issues related to the SLA management in service composition scenarios, which impose stronger requirements about flexibility of SLAs, and presents a framework for the management of dynamic SLAs.

INTRODUCTION
The extensive usage of information and communication technology in modern organizations affects B2B interactions too. While the interaction has traditionally mainly been based on manual processes, the use of digital means allows organizations to increase both the effectiveness and efficiency of their mutual interaction and consequently improves their business goals.
Flexible and Dynamic SLAs Management in Service Oriented Architectures

Such interactions must be formalized to clearly define rights and obligations of all the involved parties. The formalization usually takes the form of a contract that defines the parties engaging in the collaboration, the goods, the services or funds exchanged, and details about the way this exchange takes place. The use of electronic contracts with automated support for their management allows an increase of effectiveness and efficiency in contract processing, opening new possibilities for interaction among the parties (Angelov & Grefen, 2001, 2004).

This issue becomes more important in Service Oriented Architecture (SOA) (Papazoglou, 2003; Papazoglou & van den Heuvel, 2007), which has imposed as the reference paradigm for automated business integration. The high level of interoperability offered by the SOA enables scenarios of world-wide and cross-domains service composition. In a service-oriented scenario a service may be the result of the composition of several services deployed in many administrative domains, each of which autonomously manages resources that are quantitatively and qualitatively different. The quality of the composite service delivered to the customer is strongly affected by all the services involved in the provision. In such a scenario, where the resource availability is highly dynamic, promising and guaranteeing specific QoS levels to the customer is a real challenge.

A service composition system that can leverage, aggregate and make use of individual services’ QoS information to derive the optimal QoS of the composite service is still an ongoing research problem. This is partly due to the lack of an extensible QoS model and a reliable mechanism to compute and police QoS that is fair and transparent to both service customers and providers (Liu et al., 2004). Currently, most of the approaches dealing with QoS of web services only address some generic dimensions such as price, execution duration, availability and reliability. In some domains, such generic criteria might not be sufficient. A QoS model should also include domain specific criteria and be extensible. Most of the current approaches rely on service providers to advertise their negotiable QoS information and/or provide an interface to access the QoS values, which are subject to manipulation by the providers. Customers can then search among the service providers, negotiate for the QoS that best suits their needs and eventually sign an agreement for the service provision.

Service level agreements (SLAs) (Overton, 2002) are the most common mechanism used to establish agreements between two or more parties involved in the provision of a service. An SLA is a formal, negotiated agreement between a service provider and its customer (the service requester) involving parameters, both functional and not functional, related to the service to be provided.

The establishment of an SLA is just one of the tasks that should be fulfilled by an SLA management architecture. First, there is a need for a mechanism which enables, on the one hand, the Service Providers to advertise its capabilities under the form of SLA templates (SLA publishing) and, on the other one, the customers to discover them (SLA discovering). Before signing an SLA, the parties enter a negotiation step during which the parameters advertised in the SLA can still be modified according to the customer’s needs and the provider’s capabilities (SLA negotiation). After that, the SLA is signed by the parties and, from that moment on, is considered established (SLA formation). To verify that the SLA is actually honored the parties can monitor the quality level of the provided service against the one promised in the SLA (SLA monitoring). Finally, the SLA is either naturally terminated (its lifetime expires) or is voluntarily terminated by one of the parties (SLA termination).

This is a very rigid vision of the SLA: a contract is established, enforced and then terminated (either voluntarily or not). As we are going to show, this static vision does not meet the requirements of highly dynamic and unpredictable scenarios. Actually, while monitoring the provided QoS, the