QoS Specification for Component Infrastructures and Mobile Web Services

Miguel A. de Miguel, Juan C. Yelmo
Department of Telematics Engineering, Technical University of Madrid.
Ciudad Universitaria, 28040 Madrid Spain
mmiguel,jcyelmo@dit.upm.es

1. Introduction

Currently, the interface description languages in component and web service infrastructures provide support for the description of functional properties, and structures such as deployment descriptors configure some specific non-functional aspects (e.g. persistence, security, transactions). But the infrastructures do not provide support for the description and management of general QoS (e.g. latencies, reliability, availability). In this poster we propose some solutions to integrate the QoS specification of model driven approaches and infrastructures of components and web services. The component reuses this information for QoS management purposes. In our approach, QoS-aware models and infrastructures use compatible QoS specification languages. The QoS specification provides basic information for the QoS management tasks. In component infrastructures, basic structures can integrate the QoS management tasks. Management tasks access to QoS repository to get information, evaluate expressions, and test monitored values. The *QoS Specification Repository* handles the QoS information and provides some basic operations (e.g. evaluation of constraints and introspection for adaptation management).

UML is a set of notations that provides support for the construction of general objects-oriented systems. In January of 2002 Object Management Group published a Request For Proposals searching for a *UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms* [3]. The RFP includes three main mandatory requirements: i) *General quality of service framework* that ensure the consistency of models that support different types quality characteristics and categories. ii) *Definition of individual characteristics*. iii) *Generic fault-tolerance framework*. The *QoS framework* included in our initial submission [3] defines a metamodel that is a basic concept for the construction of a QoS profile and a repository of QoS specifications.

1.1. QoS-aware Web Services and Components

Web Services as a whole can be defined as a loosely coupled middleware based on standardized XML, allowing a piece of service logic located somewhere on the internet to be accessible using standard-based Internet protocols such as HTTP. Even though, some concerns regarding non-functional requirements (QoS aspects) remain to be solved.

WSDL maybe enriched with QoS specification [2] expressing the QoS level supported by a service provider at a specific end-point. Service clients request establishment of the binding by specifying the reference to a Web Service interface. This request also contains the required QoS. A QoS broker of the QoS-aware platform searches a UDDI registry for the service providers offering and starts a negotiation by comparing offered and required QoS. In case of successful negotiation, both client and service provider are informed and binding successfully built.

The model elements of components that we use are based on a general reference architecture [1], and our implementations are based on EJB. In our architecture the facets and receptacles interconnect the group of components, which collaborate to provide support of qualified functionalities. The *qualified functionalities* are the external QoS system operations, which have associated a *quality utility* that express the degree of satisfaction of the operation from the user or external system point of view. In this model, the container support the QoS management based on adaptation and negotiation processes. The container is generated automatically based on the information at the QoS model.

2. References

[1] International Organization for Standardization, CD15935 Information Technology: *Open Distributed Processing - Reference Model - Quality of Service,* ISO document ISO/IEC JTC1/SC7 N1996 (October 1998).

[2] A. Mani, A. Nagarajan, "Understanding Quality of Service for Web Services", IBM's developersWork, (Jan. 2002)

[3] Object Management Group, UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms RFP, OMG document number ad/02-01-07 (January 2002).

[4] Object Management Group, UML Profile for Modeling Quality of Service and Fault Tolerance Characteristics and Mechanisms Initial Submission realtime/2002-09-01 http://www.omg.org/cgi-bin/doc?realtime/2002-09-01