WSQoSX – A QoS Architecture
for Web Service Workflows

Rainer Berbner, Michael Spahn, Nicolas Repp, Oliver Heckmann, and Ralf Steinmetz

Dept. of Computer Science, Darmstadt University of Technology, Germany
{berbner,spahn,repp,heckmann,steinmetz}@kom.tu-darmstadt.de

Web Services as a technology to enable distributed business processes gain in importance, especially in the area of Enterprise Application Integration (EAI) and Business Process Outsourcing (BPO). However, the support of Quality of Service (QoS) is crucial in this context. Without any guarantee regarding QoS, no enterprise is willing to rely on external Web Services within critical business processes. Thus, we designed and implemented the *Web Service Quality of Service Architectural Extension (WSQoSX)* as an integrated Web Service system with comprehensive QoS support [2, 4]. WSQoSX supports the assessment of Web Services to assure that only Web Services will be used in critical business processes that satisfy the requirements defined by the user. The selection and execution of a certain Web Service depends on its QoS-properties described by a Service Level Agreements (SLAs) document. The compliance with given SLAs is monitored by WSQoSX as well. In case of a Web Service not being able to fulfil the requirements, it can be replaced during runtime by selecting an alternative Web Service out of a pool of similar Web Services. Additionally, providers can register their Web Service offerings using the same Web-based interface, making WSQoSX a marketplace for Web Services.

If a workflow managed by WSQoSX is started, the workflow engine does not invoke a Web Service directly. Web Service invocation is managed by a Proxy Component instead. This Proxy Component can determine which category (e.g. shipping) has been triggered for invocation and hands this information over to the Selection Component. The Rating Component calculates a score for each Web Service according to specific user preferences. Based on these calculations the Selection Component chooses and invokes the best suitable Web Service. The Accounting Component tracks detailed information about which Web Services have been invoked and their runtime behaviour. This data is used by the QoS Monitoring Component to detect SLA violations during the execution of Web Services [1]. The management components (Figure 1) of WSQoSX described above are implemented in Java.

The QoS-aware selection of Web Services is based on a QoS-model on which selection algorithms are applied [1]. The algorithms used mainly emanated from the operations research discipline, adapted to the special needs of Web Service selection. For this, a utility function maximizing the overall QoS subject to particular QoS constraints is introduced. This leads to an optimization problem that is NP-hard. Thus, we propose a heuristic based approach to solve the QoS-aware Web Service composition problem. For this, we design a heuristic H1_RELAX_IP that uses a backtracking algorithm on the results computed by a relaxed integer program. The evaluation of H1_RELAX_IP reveals that this heuristic is extremely fast and leads to

results that are very close to the optimal solution. H1_IP_RELAX outperforms the linear integer programming based solution of a solver with regard to the computation time, especially with increasing number of candidate Web Services and process tasks.

Due to the volatile nature of the Web Service environment the actual runtime behaviour of Web Services may deviate from the one estimated in the planning phase. Thus, we introduce a heuristic based replanning mechanism for adapting a workflow to the real behaviour ensuring that its execution remains feasible, valid and optimal subject to the preferences and constraints defined by the user [3].

Using WSQoSX enterprises are enabled to build flexible and agile business processes, generating the foundation for future cost savings.
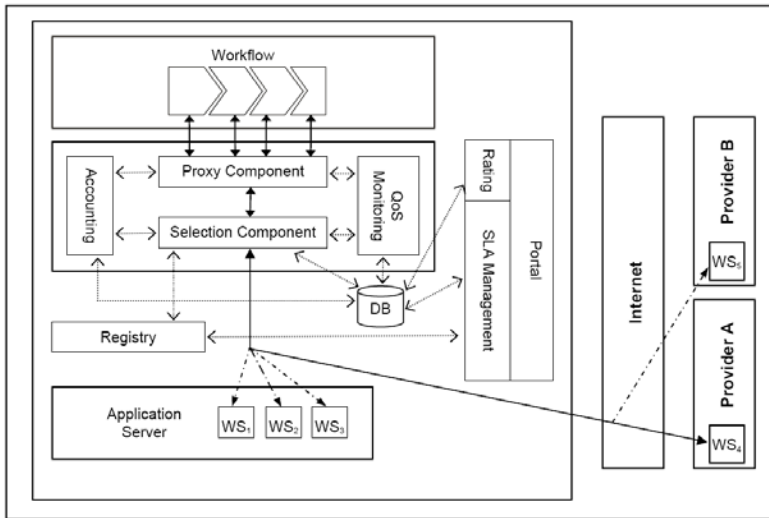

Fig. 1. WSQoSX – Architectural blueprint

References

1. Berbner, R., Spahn, M., Repp, N., Heckmann, O., Steinmetz, R.: Heuristics for QoS-aware Web Service Composition. In: ICWS 2006. Proc. 4th IEEE International Conference Web Services, Chicago, IL, USA, pp. 72–82. IEEE Computer Society Press, Los Alamitos (2006)
2. Berbner, R., Grollius, T., Repp, N., Heckmann, O., Ortner, E., Steinmetz, R.: An approach for the Management of Service-oriented Architecture (SoA) based Application Systems. In: EMISA 2005. Proc. Enterprise Modelling and Information Systems Architectures, Klagenfurt, Austria, pp. 208–221 (2005)
3. Berbner, R., Spahn, M., Repp, N., Heckmann, O., Steinmetz, R.: An Approach for Replanning of Web Service Workflows. In: AMCIS 2006. 12th Americas Conference on Information Systems, Acapulco, Mexico (2006)
4. Berbner, R., Heckmann, O., Steinmetz, R.: An Architecture for a QoS driven composition of Web Service based Workflows. In: NAEC 2005. Networking and Electronic Commerce Research Conference, Riva Del Garda, Italy (2005)