A Pluggable Framework for Tracking and Managing Faults in Service-Oriented Systems

Daniel Robinson and Gerald Kotonya

Computing Department, InfoLab21, South Drive,
Lancaster University, Lancaster, United Kingdom
{robinsdb,gerald}@comp.lancs.ac.uk

Abstract. Practical fault management in service-oriented systems requires dynamic monitoring of services for SLA violations, failures and undesirable changes in the system runtime environment. It should also include effective fault recovery strategies, and be transparent and lightweight to enhance trust and to minimise the load on the consumer and providers. This paper describes a technology-independent fault management approach that uses a pluggable brokerage model to track and resolve service changes and faults. A case study is used to illustrate the efficacy of the approach.

Keywords: Service-oriented systems, Fault tracking, Change management.

1 Introduction

Failures in service provision, Service Level Agreement (SLA) violations and changes in the system runtime environment can impact adversely on the quality of a service-oriented system. There are several initiatives based on monitoring that are designed to track changes and detect SLA violations. However, these generally support static rather than dynamic analysis and provide poor support for resolving undesirable changes and violations [1]. In addition, most are designed to support service providers avoid SLA violations, rather than help the service consumer to detect and respond to problematic QoS [3]. Effective fault management in service-oriented systems requires a consumer-centred approach that actively monitors services for SLA violations, failures and undesirable changes; and provides strategies for minimising their adverse effects.

Our solution has been to develop a failure management approach that uses a consumer-centred, pluggable brokerage model to track and renegotiate service faults and changes. The brokerage model is reported in [2]. Our approach is service-technology independent and incorporates pluggable support for different monitoring and negotiation models in addition to assessing provider reputation. To help with the automation of negotiation and monitoring processes, and to ensure a shared set of terms for describing services, our approach also incorporates pluggable support for a service ontology. We will show using a service-oriented case study and different fault and change scenarios how our framework tracks and manages faults and changes.

L. Baresi, C.-H. Chi, and J. Suzuki (Eds.): ICSOC-ServiceWave 2009, LNCS 5900, pp. 637–638, 2009.

2 An Overview of the Approach

Figure 1 shows the framework on which the approach is based. Service consumers and providers supply the brokerage system with templates that specify strategies for the services they require or provide. For consumers, the strategy describes the ideal QoS requirements of the functional services they wish to use. The brokerage incorporates an engine builder component, which uses the templates to assemble a custom service broker engine for processing negotiation messages and service proposals. The proposal engine creates and evaluates service proposals. The broker engine contains a separate negotiation engine for each negotiation protocol it supports. The negotiation engine concurrently negotiates with multiple parties. The engine maintains a separate negotiation session for each negotiation.

The framework provides a service monitoring system, which actively monitors the quality of negotiated services for emergent changes, SLA violations and failure. The primary monitoring approach adopted by the framework is a passive model, which transparently intercepts service requests and responses between service consumers and providers.

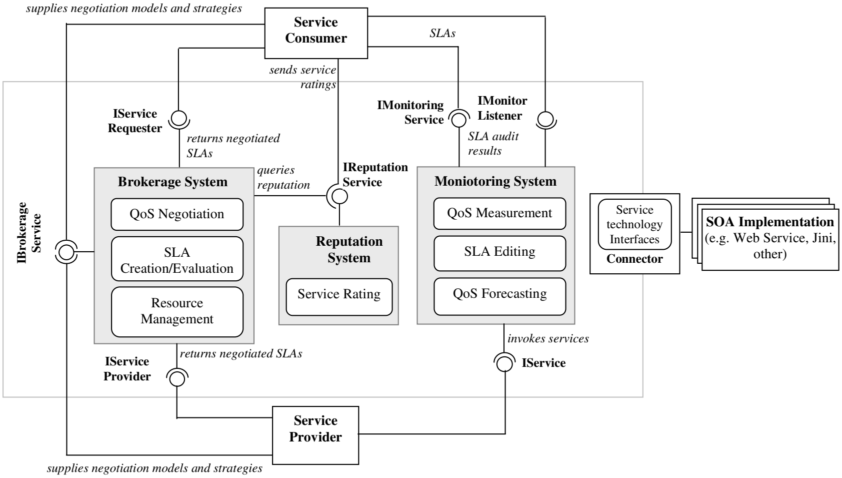

Fig. 1. Pluggable Service brokerage framework

References

1. Saunders, S., Ross, M., Staples, G., Wellington, S.: The Software Quality Challenges of Service-Oriented Architectures in E-Commerce. Software Quality Control 14(1), 65–75 (2006)
2. Robinson, D., Kotonya, G.: A Runtime Quality Architecture for Service-Oriented Systems. In: Bouguettaya, A., Krueger, I., Margaria, T. (eds.) ICSOC 2008. LNCS, vol. 5364, pp. 468–482. Springer, Heidelberg (2008)
3. Hoffman, R.: Monitoring, at your service. ACM Queue 3(10), 34–43 (2005)