

FAST-SE: An ESB Based Framework for SLA Trading

Jose Antonio Parejo, Antonio Manuel Gutiérrez,
Pablo Fernandez, and Antonio Ruiz-Cortes

University of Sevilla

Abstract. SLA driven service transaction has been identified as a key challenge to take advantage of a SOA. FAST System provides a software framework for the automated creation of SLAs. In particular it have been developed as an extension to the ESB (Enterprise Service Bus) paradigm to create a transparent SLA management layer that drives any service invocation. Our framework has been successfully applied in two different scenarios and provides an extensible architecture to address new domains.

1 Introduction

As SOC has evolved into a mature paradigm, new challenges appear in the horizon. In particular, the automatic provision of services is a promising field that could lead to a new generation of organizations that adapt "on demand" to rapid changes in its business environment.

FAST system addresses a core element of the automatic provision of services: the creation of SLAs (Service Level Agreements) that will describe the rights and obligations of the service consumer and the service provider during the transaction. The terms of the agreement could refer to either functional (such as the type of service – i.e. the interface-) or non-functional (such as the availability of the service) features. In doing so, our approach provides an architecture based on components to extend a JBI Enterprise Service Bus with a new element: the FAST-SE (FAST Service Engine). This element provides an extensible software framework to deal with different service domains: On the one hand, the system defines standard data models (information, proposals, counterparties, agreements, etc.) that can be refined with a specific vocabulary of the domain. On the other hand a flexible orchestration system is provided to allow different SLA creation transactions.

A short video demonstration of the system can be found in¹

2 FAST Extension Capabilities

The FAST (Framework for Automatic Service Trading) system provides an architecture based on components that are interconnected with a set of linking choreographies and generic data models. The implementation of linking choreographies is independent of the vocabulary of the information so they can be adapted to different domains.

¹ <http://www.isa.us.es/fast>

Also, the components develop a set of generic roles executed as autonomous process so their behavior can be orchestrated.

Adapting to a specific domain only requires defining the domain vocabulary: i.e. the specific (functional and non-functional) properties. Once the framework is instantiated the parties (consumers and providers) specify their SLA creation preferences in the trading process using the domain vocabulary. Finally, a domain can also specify the component orchestration to model the trading process that will create the SLA. Following, a short description of two domains already implemented is described.

Computing Marketplace Domain. In this case, a set of providers trade with a computation service according to different properties as cost, computation time or delay. The interaction choreographies are implemented in a distributed deployment scenario and the agreement conditions are obtained optimizing domain constraints between consumer requirements and providers features. Additionally, it is important to remark that the application of the framework in this scenario provides the feasibility of adapting the trading process stages in each provider.

Federated bus government. The framework is applied to govern services in a federated bus inside a corporation. The system creates agreements to plan service invocations inside a federated bus with different service providers according to certain conditions as reliability, performance or priority. The linking choreographies are implemented to adapt a high performance environment. Moreover, the framework provides an extensible environment to adapt new corporate services or constraints with minimal effort. Currently, the system is in a pre-production test stage and it is deployed to federate corporative services implemented by several departments in a wide-size organization for a regional government.

3 Conclusions

The system has been deployed successfully in two different scenarios proving their adaptability to their vocabulary and interaction needs. Currently, the system is being extended to enrich the expression language in properties and preference. A set of generic components is developed to deal with complex negotiations and CSP-based selection.