Chapter 28

Enterprise Service Bus for Building Integrated Enterprises

M. Antonia Martínez-Carreras
Universidad de Murcia, Spain

Francisco J. García-Jiménez
Universidad de Murcia, Spain

Antonio F. Gómez-Skarmeta
Universidad de Murcia, Spain

ABSTRACT

Lately, the building of Enterprise Application Integration (EAI) where different legacy-applications may interoperate between them has gained the focus of business research. In this sense, the Service Oriented Architecture (SOA), and particularly the utilization of Web services standards, has attracted the attention of several researchers and practitioners for implementing the needs of EAI. More concretely, the emergence of Enterprise Service Bus (ESB) has brought a layer for favouring the mediation, transformation, and thus, the communication between these diverse applications, services, or business processes in a decoupled way. Indeed, the ESB technology integrates a wide range of current technologies and business standards. The aim of this chapter is to offer the design and necessities of Future Business Environments comparing how open ESBs fulfils these requirements. Furthermore, this chapter compares six of the most well-known open ESBs considering the characteristics provided in the design of Future Business Environments.

INTRODUCTION

Nowadays the utilization of Service Oriented Architecture (SOA) is an important trend in current Enterprises for increasing interoperability between different systems and legacy applications. The main purpose of this architecture is to allow the building of coarse-grained components and loosely coupled frameworks. Through this communication of services, interoperability can be achieved fulfilling some of the needs of Enterprise Application Integration (EAI). The main purpose of EAI is to “integrate applications systems that are designed for different business
functions” (Deng et al, 2008), providing in turn, new business processes and services and allowing the cooperation between them with the aim of fulfilling some objectives in the organization.

One of the leading technologies for implementing SOA are those based on WS-* standards such as SOAP, WSDL and WS-BPEL (Erl, 2005). Furthermore, the Web Service Business Process Execution Language (WS-BPEL), also known as BPEL, emerged with the aim of enabling the service composition.

According to Beimborn and Joachim (2010) the study of SOA has been carried out during several years, however researchers have started recently to investigate about the impact of SOA in the performance with regards to enterprises. Moreover, with the aim of obtaining the benefits of SOA effectively, organizations are currently using the approach of the Enterprise Service Bus (ESB) (Ortiz, 2007).

Considering the definition of Chappell (2004) “an ESB is a standards-based integration platform that combines messaging, web services, data transformation, and intelligent routing to reliably connect and coordinate the interaction of significant numbers of diverse applications across extended enterprises with transactional integrity”. Therefore, what ESB provides to SOA is the communication layer and integration logic between the client and server modules (Papazoglou and van den Heuvel, 2007). In this line, Fulton (2009) stated that “an ESB is usually the core infrastructure for SOA”. Indeed, ESB improves the ability to change between different implementations of the technology infrastructure without affecting to the business layer (Beimborn and Joachim, 2010). In fact, ESB provides some fundamental functions such as support of multiple protocols (HTTP; SOAP, REST, JMS) and data transformation or message routing.

With regards to the building of integrated business systems there exists several works (Bagheri et al., 2007; Beimborn and Joachim, 2010; Du et al., 2008) which indicate the benefits of using message broker architectures for communicating the different services, applications or resource in the enterprise, and additionally some of them point to the use of ESB as the core for the integration of the disparate applications and resources managed by the enterprises.

Such is the relevance of this technology that several ESB platforms have appeared concerning both commercial, such as IBM WebSphere Message Broker (WebSphere ESB), TIBCO Business Work (TIBCO Business Works) or Oracle ESB (Oracle ESB), and open-source implementations, such as Fuse ESB (FuseESB), Mule (MuleESB), Petals (Petals ESB), JBoss ESB (JBoss ESB), WSO2 ESB (WSO2 ESB) or OpenESB (OpenESB). Due to the fact that some business and research is based on open source systems, our chapter is centered on them. All the above-mentioned ESB products manage different kinds of technologies for transforming, integrating or delivering information on it. Thus, the choice of different ESB products influences in the developments or integrations carried out on it (García-Jimenez, Martinez-Carreras & Gomez-Skarmeta, 2010).

The aim of this chapter is twofold. In one hand we describe the design of Future Business Environments and how the ESB fulfills their needs. And on the other hand, we provide a comparison of six of the most well-known open source ESBs analyzing how they fulfill the requirements of Future Business Environment. More precisely, we compare the followings products: Fuse ESB, Mule ESB, Petals ESB, JBoss ESB, WSO2 ESB and OpenESB. The reason why we use open source products in our developments is due to the advantages they can offer to business perspective such as reducing costs.

This chapter is structured as follow. First we review the evolution of interoperability and the ESB concept. Section three indicates the state of the art in the research on ESB. After that, we indicate some of the functionalities that can be provided by ESB products. In the subsequent sec-