Airport Enterprise Service Bus with Three Levels Self-Healing Architecture (AESB-3LSH)

Suha Afaneh, Department of Computer Science, Faculty of Information Technology, Isra University, Amman, Jordan

Issam Al Hadid, Department of Computer Information Systems, Faculty of Information Technology, Isra University, Amman, Jordan

ABSTRACT

This paper introduces the different aviation and airport information technology systems. Also, this paper provides Airport Enterprise Service Bus with Three Levels Self-Healing Architecture based on the Service Oriented Architecture (SOA) that improves the information accessibility and sharing across the different Airport's departments, integrate the existing legacy systems with other applications, and improve and maximize the system's reliability, adaptability, robustness and availability using the Self-Healing Agent, Virtual Web Service Self-Healing Connector and Extended Execution Engine with Process Execution Self-Healing Manager to guarantee the Quality of Service (QoS) or Service provided and business process execution.

Keywords: *Airport Enterprise Service Bus, Aviation and Airport Information Technology Systems, Quality of Service (QoS), Self-Healing, Service Oriented Architecture (SOA)*

1. INTRODUCTION

Airports need to adapt new technologies to react effectively and quickly to customers' needs and to provide a better service such as the electronic ticket. In addition to the challenges of the ability to respond to the growing requirements of the automatic information interchange between the different systems to ensure safe and efficient airport operations. Most of the challenges in the Airport systems today lie in the ability to respond to the growing requirements of the automatic information interchange between the different departments including operational, statistical, aviation and financial information. In addition, the integration with the existing legacy systems to ensure safe and efficient airport operations. All the operations in the airport are driven by the exchanged information; Airport business units create information, transform information, distribute information, and take action on received information. Airports' systems are developed by different vendors and were not designed to be interoperable, which makes systems integration a very complicated and not easy to be implemented. Service Oriented Architecture (SOA) provides the ability to address the distributed computing requirements; protocol independent, loosely coupled, reusability and standard based (Papazoglou & Heuvel, 2007; Keen et al., 2004;

DOI: 10.4018/ijstmi.2013070101

Chappell, 2005). It is based on the Web Services; distributed, loosely coupled, reusable software components that encapsulate a discrete functionality and can be accessed using standard internet and eXtensible Mark-up Language (XML)-based protocols (Sommerville, 2007; Colan, 2004; Schlimmer, 2002). SOA encourages a lot of businesses to move toward the adapting the SOA architecture to enable the response to change faster and to cut the cost of replacing the legacy systems that they have and integrate with the new systems so all the information can be accessed and shared by all the systems (Keen et al., 2004; Minoli, 2008), accordingly; SOA will provide a guideline for airport Information systems architecture design, development and integration. The functionality provided by the integration platform Enterprise Service Bus (ESB) is based on the SOA that utilizes Web service standards to supports a variety of communication patterns over multiple transport protocols to connect different applications and technologies (Papazoglou & Heuvel, 2007). In addition, the features of loosely coupling and breaking up the integration logic into separate parts can be easily managed (Keen et al., 2004). ESB provides architecture based on the SOA that improves the information accessibility and sharing across the different Airport's departments. Furthermore, it provides a component interface to existing legacy system so it can be integrated with other applications, accessed over the web, and support the reusability of the legacy systems. Also, the Self-Healing Agents, Virtual Web Service Self-Healing Connector and Extended Execution Engine with Process Execution Self-Healing Manager which will improve and maximize the system's reliability, adaptability, robustness and availability.

2. RELATED WORK

There are many different information technology systems that are used in the aviation industry (Abu-Taieh, 2009), shown in Table 1. Many Airports and Air Traffic Control (ATC) Units have moved toward the adapting of new open systems (Goold, n.d.). The information can be accessed, shared and flow across the different hardware systems, operating systems, networks and airport management systems, because there must be an integration framework for defining the information integration requirements, and designing the systems integration architecture to address the loosely coupled systems, standards-based interfaces, and protocol independent distributed computing. this manages information elements with a defined process and provides the ability to upgrade, replace or move systems or components without having to modify code and disrupt execution of the existing applications (Cheng, 2001). Service Oriented Architecture (SOA) provides the ability to address the distributed computing requirements; protocol independent, loosely coupled, reusability and standard based (Papazoglou & Heuvel, 2007). It is based on Web services; distributed, loosely coupled, reusable software components that encapsulate a discrete functionality and can be accessed using standard internet and XML- based protocols (Sommerville, 2007). SOA provides flexible architecture that unifies the business process by modularizing applications into services which satisfy the addressed requirements for the Airport and aviation systems integration architecture. The functionality of the integration architecture must also support a variety of communication patterns over multiple transport protocols, this requirement is addressed by the integration platform Enterprise Service Bus (ESB) which is based on the SOA, It utilizes Web service standards to support a variety of communications patterns over multiple transport protocols to connect different applications and technologies (Papazoglou & Heuvel, 2007). It also, provides the integration logic between the service consumer and provider which is used to transform messages, route requests and convert transport protocols between the two parties (Keen et al., 2004; Chappell, 2005). The configuration and orchestration of the services in unified and clearly defined processes are provided by the Service Orchestration using the Business Process Execution Language (BPEL) (Juric, 2006; Baresi & Guinea, 2008; Gustavo et al., 2003), this allows the business