Chapter 15
Enterprise Integration With the Structural Services Architectural Style

José Carlos Martins Delgado
Instituto Superior Técnico, Universidade de Lisboa, Portugal

ABSTRACT

The Service-Oriented Architecture (SOA) and Representational State Transfer (REST) architectural styles are the most used for the integration of enterprise applications. Each is more adequate to a different class of applications and exhibits advantages and disadvantages. This chapter performs a comparative study between them. It is shown that SOA and REST are dual architectural styles, one oriented towards behavior and the other towards state. This raises the question of whether it is possible to combine them to maximize the advantages and to minimize the disadvantages. A new architectural style, Structural Services, is proposed to obtain the best characteristics from SOA and REST. As in SOA, services are able to offer a variable set of operations and, as in REST, resources are allowed to have structure. This style uses structural interoperability, based on structural compliance and conformance. A service-oriented programming language is also introduced to instantiate this architectural style.

INTRODUCTION

After three industrial revolutions, focusing on mechanization, mass production and digitization, respectively, the industry has set its goals on a fourth revolution, commonly known as Industry 4.0 (Liao, Deschamps, Loures, & Ramos, 2017), entailing a vision of an intelligent factory where people, machines, processes, customers and suppliers are streamlined to produce and maintain smart products and services, thereby contributing to an improved society.

One of the main challenges that need to be overcome to turn this vision into a reality is *integration* (Panetto & Molina, 2008), the ability to meaningfully and efficiently cooperate with other subsystems in order to pursue the goals of the system as a whole. Integration can be seen at all levels of abstraction and complexity, from low-level cyber-physical systems (Zanero, 2017) to high-level enterprise value chains

DOI: 10.4018/978-1-5225-4936-9.ch015

targeting the capabilities required by Industry 4.0 (Schumacher, Erol, & Sihn, 2016). This chapter focus on the integration of enterprise information systems (or simply enterprise integration), a major goal to support the requirements of Industry 4.0. The EN/ISO 19439 standard (ISO, 2006) refers to enterprise integration as the process of ensuring the interaction between enterprise entities necessary to achieve domain objectives.

Enterprise integration is a very complex issue, spanning from lower levels, such as data and service interoperability, to the highest levels, including strategy and governance alignment. The former typically resort to technologies based on architectural styles such as SOA (Erl, 2016) and REST (Pautasso, 2014). The latter are mostly dealt with in a tacit way or at the documentation level, in coordination with the enterprise architectures.

The ISO/IEC/IEEE 24765 standard (ISO, 2010) provides the seemingly most cited definition of *interoperability*, as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged".

Interoperability asserts the ability of two systems to understand each other's messages, whereas integration requires collaboration to achieve common goals. Interoperability is thus necessary but not sufficient to achieve enterprise integration (Chen, Doumeingts, & Vernadat, 2008), which usually entails cooperation and coordination at higher abstraction levels.

Just ensuring interoperability is already a daunting task, given the complex, heterogeneous and highly variable enterprise collaboration networks that characterize today's fast-paced enterprise landscape, namely Industry 4.0 scenarios. This is particularly true when we consider the most recent, game-changing developments, such as cloud computing (Mezgár, & Rauschecker, 2014), big data (Marz, & Warren, 2015; Reed, & Dongarra, 2015) and applications in the context of the Internet of Things (Want, Schilit, & Jenson, 2015).

However, this is not the full picture. In general, the integration problem revolves around two conflicting goals:

- **Interoperability:** Enterprises need to interact to accomplish collaboration, either designed or emergent. This necessarily entails some form of mutual knowledge and understanding, but this creates dependencies that may hamper the evolution (changes) of enterprise information systems;
- **Coupling:** Enterprises should not have dependencies on others, in order to evolve freely and dynamically. Unfortunately, independent enterprises do not understand each other and are not able to interact, which means that some form of coupling is unavoidable.

Therefore, the *fundamental problem of integration* is to provide (at most) the *minimum coupling* possible while ensuring (at least) the *minimum interoperability* requirements. In other words, the main goal is to ensure that each interacting party knows just enough about the others to be able to interoperate with them but no more than that, to avoid unnecessary dependencies and constraints. This is an instance of the *principle of least knowledge* (Hendricksen, 2014).

Enterprises are complex systems and suffer exactly from the same problem, with the additional concern that agility (which translates to fast evolution) is a critical requirement for the survival of even the largest enterprises. These are in fact quite vulnerable, because their complex architecture has typically woven a large web of dependencies. Current integration technologies, based on SOA and REST, do not necessarily comply with the principle of least knowledge.