Chapter 2

Enterprise Integration: Challenges and Solution Architecture

Leo Shuster
Nationwide Insurance, USA

ABSTRACT

The purpose of this chapter is to discuss the current state and associated challenges of Enterprise Integration (EI). The chapter will explore EI’s past, present and future, examine its path towards the current practices, and contemplate its future evolution. Synergies between Service-driven and other modern architectural approaches will be investigated. Major challenges associated with EI strategy and execution will be explored in depth. This will include organizational, technology, process, methodology, and governance challenges. In addition to the current state concerns, future trends and directions will be investigated and specific challenges outlined. A major part of this chapter will be devoted to defining and discussing modern solution architectures associated with EI. This will include current architectural best practices, technology constructs employed, design patterns, governance mechanisms, and implementation considerations.

INTRODUCTION

When Charles Darwin published his Origin of Species in 1859, he was not aware that his work would be applied to so many topics outside of biology. Yet, the evolutionary paradigm, survival of the fittest, and many other postulates from Darwin’s work can be directly applied to a variety of other topics. Just like the Earth’s species, Enterprise Integration has evolved over the years – from humble beginnings in data replication and messaging to ETL (Extract-Transform-Load) and Service-driven mechanisms. The current integration approaches have evolved over the years and are all built on the deep foundation of integration knowledge and experience. Today, Service Oriented Architecture (SOA) plays a major part in Enterprise Integration.

SOA has significantly influenced the direction of Enterprise Integration. Many enterprise technologies such as BPEL, BPM, Business Rules, etc, enable Service-driven methodologies for enterprise integration. This chapter identifies SOA’s influences and usage and primarily concentrates on the integration architectures.

DOI: 10.4018/978-1-4666-4193-8.ch002
What is Enterprise Integration?

Many definitions of Enterprise Integration exist. According to Gartner, it is the “unrestricted sharing of data and business processes among any connected applications or data sources in the enterprise.” This is the simplest and the best definition of Enterprise Integration. It succinctly defines the core function of EI and draws connections between its past, present, and future.

The modern form of Enterprise Integration, Service-driven Application Integration is based on SOA. First described by Gartner (1996), Service Oriented Architecture (SOA) is a style of software architecture that is modular, distributed and loosely coupled. SOA-style applications use business components that are designed to be reusable across applications and enterprise boundaries. These components are invoked through Services that are based on well-defined interface definitions, and are independent of the underlying hardware and software platforms, as well as development language (Gartner, 1996). Service-driven Application Integration uses SOA approaches to enable integration across the enterprise.

Considering the key differences between the Enterprise Integration and SOA definitions above, one can see the emphasis on integration with EI and reuse with SOA. Therein lays the core development in the maturation process of integration techniques – from basic application and data integration, to better componentized, reusable, and specialized code fragments.

Charles Darwin described human evolution from single cell organisms to the complex, intricate life forms we are today. Integration evolution can be described in similar terms. It started out as simple point-to-point communication and evolved into the complex Service-driven architecture style as shown in Figure 1.

Enterprise Application Integration (EAI), the first structured integration technique, became popular in the late 1980s, early 1990s. IT environments had matured to the point where multiple systems performed various critical functions and needed to be integrated with each other. The days

Figure 1. Evolution of enterprise integration
Related Content

[www.igi-global.com/article/service-oriented-software-engineering/126635?camid=4v1a](www.igi-global.com/article/service-oriented-software-engineering/126635?camid=4v1a)

A Proposal to Elicit Usability Requirements within a Model-Driven Development Environment
[www.igi-global.com/article/a-proposal-to-elicit-usability-requirements-within-a-model-driven-development-environment/120171?camid=4v1a](www.igi-global.com/article/a-proposal-to-elicit-usability-requirements-within-a-model-driven-development-environment/120171?camid=4v1a)

Temporal Join with Hilbert Curve Mapping and Adaptive Buffer Management

Open Source Software Systems: Understanding Bug Prediction and Software Developer Roles
[www.igi-global.com/chapter/open-source-software-systems/37047?camid=4v1a](www.igi-global.com/chapter/open-source-software-systems/37047?camid=4v1a)