Chapter 11
Service Oriented Architecture
Conceptual Landscape:
Part I

Ed Young
Victoria University, Australia

ABSTRACT

Contemporary architectural approach is for an orchestrated, agnostic, federated enterprise through the adoption of loosely-coupled open Service interfaces. The Service-Oriented Architecture (SOA) paradigm unifies disparate, heterogeneous technologies. It resurrects legacy technology silos with a Service ‘face-lift’ while maintaining their autonomy. Somewhat in its infancy as standards and methodologies are evaluated and adopted, the differences between theory and praxis of SOA remain to be fully determined, predominately due to the size and complexity of the conundrum it addresses.

INTRODUCTION

Service-Oriented Architecture (SOA) attempts to deliver a potentially Panglossian promise of an IT infrastructure agile enough to cater for rapidly changing Business demands. It offers a panoptic vantage point for enterprise Business state and empowers the Business to define and map IT infrastructure to process.

This article draws extensively on published research in the past two years and supporting sources germane to current SOA issues in collation, to describe a conceptual landscape of current, prominent SOA concerns.

Part 1 addresses how SOA is defined, its characteristics, evolution, motivation and approach.

Final conclusions are presented based on the literary review and in relation to the OASIS SOA Reference Architecture (OASIS, 2006).

SERVICE-ORIENTED ARCHITECTURE (SOA)

Service Oriented Architecture (SOA) is a paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains (OASIS, 2006).
Service-oriented architecture (SOA) is defined as a paradigm for organizing and using distributed capabilities that might be under the control of different ownership domains. SOA is also known as a methodology for achieving application interoperability and reuse of IT assets in distributed computing environments characterized as transformable by the visibility, interaction, and effect dimensions (Abuosba & El-Sheikh, 2008).

In reality, SOA is not an architecture, but an architectural pattern from which an infinite number of architectures can be derived - both good and bad (Lewis, Morris, Simanta, & Wrage, 2007).

**Characteristics**

**Definition 1**: An ‘Enterprise’ is a business association consisting of a recognized set of interacting business functions. It is capable to operate as an independent, standalone entity. With this Definition, there can be enterprises within enterprises. For instance, a business unit within the overall corporate entity may be considered an enterprise as long as it could be operated independently. The enterprise can also be seen as an ‘Extended Enterprise’, meaning that the scope of the impact of an enterprise architecture effort could also include interrelationships with external entities. Such as: suppliers, business partners, and customers. (12Manage, 2008)

**Definition 2**: ‘Architecture’ provides the underlying framework. This defines and describes the platform required by the enterprise so that it can attain its objectives and achieve its business vision. It can be defined as: the set of principles, guidelines, policies, models, standards, and processes that, aligned with business strategy and information requirements, that is guiding the selection, creation and implementation of solutions that are aligned with future business direction. (12Manage, 2008)

Laplante, Zhang, and Voas (Laplante et al., 2008) define SOA as a, ‘... software construction model’ within the context of their discussion of the differences between SOA and Software as a Service (SaaS). Use is made of Zachman’s architecture description approach (Zachman, 1987) to examine the differences.

The Zachman Framework is a classification scheme for the Definition, development and documenting of enterprise-wide Information Systems (Varga, 2003).

Zachman identifies players and their perspectives of an enterprise architecture framework. These perspectives are used for the rows of a grid against the information ‘category’ columns:

- Data (what)
- Function (how)
- Network (where)
- People (when)
- Time (who)
- Motivation (why)

‘The problem [with the Zachman method] is the lack of standard artefact descriptions for some of the Framework’s cells’ (Varga, 2003). Utilising Zachman’s framework, SaaS and SOA are distinguished (see Tables 1 and 2).

The hope is that the comparison will assist designers and developers make more informed architectural decisions (see Table 3).

**EVOLUTION**

From object distribution with local transparency and the responsibility of the software developer to cater for the location of the object themselves, the caching problem, through CORBA and DCOM still heavily influenced by distributed research again with local object transparency and no resolution to the caching problem but not well distributed, we have services.
Related Content

**A Case Studies Approach to the Analysis of Profiling and Framing Structures for Pervasive Information Systems**
[www.igi-global.com/article/case-studies-approach-analysis-profiling/73912?camid=4v1a](www.igi-global.com/article/case-studies-approach-analysis-profiling/73912?camid=4v1a)

**Adaptive Web Services Monitoring in Cloud Environments**
[www.igi-global.com/article/adaptive-web-services-monitoring-cloud/78350?camid=4v1a](www.igi-global.com/article/adaptive-web-services-monitoring-cloud/78350?camid=4v1a)

**Practitioner Case Study: Practical Challenges in Portal Implementation Projects**
[www.igi-global.com/article/practitioner-case-study/3033?camid=4v1a](www.igi-global.com/article/practitioner-case-study/3033?camid=4v1a)

**Paradox of Social Portals**
[www.igi-global.com/chapter/paradox-social-portals/17949?camid=4v1a](www.igi-global.com/chapter/paradox-social-portals/17949?camid=4v1a)