Chapter III
Model-Centric Architecting Process

Tony C. Shan
IBM, USA

Winnie W. Hua
CTS, Inc., USA

ABSTRACT

This chapter defines a methodical approach, named model-centric architecting process (MAP), to effectively cope with the architecture design complexity and manage the architecting process and lifecycle of information systems development in a service-oriented paradigm. This comprehensive method comprises four dimensions of architecting activities: requirement analysis, specification, validation, and planning (RSVP). The process is broken down into nine interrelated models: meta architecture, conceptual architecture, logical architecture, physical architecture, deployment architecture, management architecture, information architecture, aspect architecture, and component architecture. A 2-D matrix serves as a blueprint to denote a step-by-step procedure to produce and manage the architectural artifacts and deliverables in the lifecycle of systems architecture design, development and governance. The holistic framework provides a multidisciplinary view of the design principles, tenets, idioms, and strategies in the IT architecting practices. The characteristics and features of the constituent elements in the MAP approach are articulated in great detail. Recommendations and future trends are also presented in the context. It helps build high-quality service-oriented solutions focused on different domains, and in the meantime keeps the agility, flexibility and adaptiveness of the overall method. This systematic framework may be customized in different formats to design various information systems in different industries.

INTRODUCTION

In today's on-demand business world, the electronic business models demand increasingly higher performance of information technology (IT) systems. We must do more with less, so as to provide a higher level of services at a lower cost for the business to compete and succeed. This means that IT has to build more complex, flexible, scalable, extensible, and forward-thinking technical solutions, to meet the ever-growing business needs.

In large organizations like worldwide financial institutions, virtually hundreds, if not thousands, of IT applications and systems have been built or purchased to provide electronic services for external customers and internal employees in the past years, utilizing heterogeneous technologies and architectures to satisfy diverse functional requirements from different lines of business. The banking industry is no exception. The business process operations generally contain different business sectors in retail, commercial, small business, wealth management, and capital management. In particular, services are delivered to different channels such as automated teller machines (ATMs), Web browsers, interactive voice response, agent assistance, emails, mobile devices, and so forth. To effectively manage the architecture complexity and optimize the design practices in such a disparate environment, a multidisciplinary design approach is crucial to abstract concerns, divide responsibilities, mitigate risks, encapsulate the complexity, and rationalize operating processes.

BACKGROUND

Previous studies in the last few decades have strived to address the issue of architecture design complexity, which has grown exponentially as the computing paradigm has evolved from a monolithic to a service-oriented architecture. The Zachman framework (Zachman, 1987) is a logical structure for classifying and organizing the descriptive representations of an enterprise IT environment that are significant to the management of the organization as well as to the development of the enterprise's information systems. It takes the form of the two-dimensional matrix, and has achieved a level of penetration in the domain of business and information systems architecture and modeling. It is mainly used as a planning or problem-solving tool. However, it tends to implicitly align with the data-driven approach and process-decomposition methods, and it operates above and across the individual project level. Extended enterprise architecture framework (E2AF) (IEAD, 2004) takes a very similar approach in the Zachman framework. Its scope contains business, information, system, and infrastructure in a 2-dimensional matrix. E2AF is more technology-oriented. Both of these approaches are heavyweight methodologies, which require a fairly steep learning curve to get started.

Rational unified process (RUP) (Kruchten, 2003) attempted to overcome these shortcomings by applying the unified modeling language (UML) in a use-case driven, object-oriented and component-based approach. The concept of 4+1 views interprets the overall system structure from multiple perspectives. RUP is more process-oriented, and rooted in a waterfall-like approach. RUP barely addresses software maintenance and operations, and lacks an in-depth coverage on physical topology and development/testing tools. It mainly operates at the individual project level. RUP has recently been expanded to enterprise unified process (EUP), part of which has become open source – OpenUP in Eclipse Process Framework (EPF) project (Eclipse, 2007).

Another heavyweight approach, The Open Group Architectural Framework (TOGAF) (The Open Group, 2007), is a detailed framework with a set of supporting tools for developing an enterprise architecture to meet the business and