Chapter 2.14
Facilitating Design of Efficient Components by Bridging Gaps Between Data Model and Business Process via Analysis of Service Traits of Data

Ning Chen
Xi'an Polytechnic University, China

ABSTRACT

In many large-scale enterprise information system solutions, process design, data modeling and software component design are performed relatively independently by different people using various tools and methodologies. This usually leads to gaps among business process modeling, component design and data modeling. Currently, these functional or non-functional disconnections are fixed manually, which increases the complexity and decrease the efficiency and quality of development. In this chapter, a pattern-based approach is proposed to bridge the gaps with automatically generated data access components. Data access rules and patterns are applied to optimize these data access components. In addition, the authors present the design of a toolkit that automatically applies these patterns to bridge the gaps to ensure reduced development time, and higher solution quality.

DOI: 10.4018/978-1-60566-330-2.ch007

INTRODUCTION

With the development of information technology, enterprise information becomes more complex and tends to change more frequently; consequently enterprise should adjust its business according to market, which requires enterprise IT system to be flexible and agile enough to response to the changes. Now, business process modeling consists of service modeling, data modeling and component modeling, which are the three main threads in enterprise IT system solution design (Ivica, 2002;

Mei, 2003). They are usually performed relatively independently, for different roles employ different methodologies. The result is in a gap among process model, data model and components, which requires significant amount of efforts to fill in the gap. Enterprise information system is an application with dense data (Martin, 2002) and mass data access. Both functional and non-functional aspects, such as system response time and data throughput etc., are satisfied in system integration in order to provide efficient data access within process execution. Meeting these requirements is a challenge presented to the solution designed, which will greatly affect the efficiency of system development. Therefore, how to build the relationship model between business process and data model, and how to use the orchestration model to automatically generate data access components are two questions that have great impact to software development.

RELATION WORKS

The existing enterprise modeling approaches are focused on two domains including peer-to-peer enterprise system and multilayer Enterprise Modeling.

David (2004) presents a loosely coupled service-composition paradigm. This paradigm employs a distributed data flow that differs markedly from centralized information flow adopted by current service integration frameworks, such as CORBA, J2EE and SOAP. Distributed data flows support direct data transmission to avoid many performance bottlenecks of centralized processing. In addition, active mediation is used in applications employing multiple web services that are not fully compatible in terms of data formats and contents.

Martin Fowler and Clifton Nock summarize customary patterns of enterprise application architecture to accelerate development of enterprise modeling (Martin, 2002; Clifton, 2003).

However, the existing enterprise modeling methods remain largely unharnessed due to the following shortages: (1) They lack the automation of analysis mechanism which makes the enterprise unresponsive to the enterprise changes and increases the maintaining overhead of the evolution of these models; (2) Some enterprise models are just conceptual models and should be analyzed by hand. Others employ the complex mathematical models for analysis, which are hard for the business users to comprehend and manipulate. (3) The knowledge reuse is difficult for the business users due to the heterogeneity of the enterprise models.

In order to tackle the above problems, through deep analysis of business process modeling and data modeling, we extract process data mapping and data access flow to build data access components for bridging business process and data model. Furthermore, a pattern is automatically applied to data access component for facilitating an efficient service.

PROCESS/DATA RELATIONSHIP MODEL

In present environment for software development, different tools are used by separate roles in business process modeling, data modeling, software component designing and coding. These tasks are so independent that the whole software development becomes rather complex. Take the IBM develop studio as an example, we need to use modeling and programming tools such as WBI-Modeler, Rational Software Architect and WSAD-IE (Osamu, 2003). The development procedure contains the following steps:

1. The analyst will analyze requirement to design the use case using UML.
2. By analyzing the relationship between enterprise entities, the data model designer will design the data model, and create the database on the basis of UML.