Business Flexibility and Operational Efficiency - Making Trade-Offs in Service Oriented Architecture

Yiwen Zhang
University of Arizona, yiwen@u.arizona.edu

Mohan Tanniru
University of Arizona, mtannriu@eller.arizona.edu

Follow this and additional works at: http://aisel.aisnet.org/amcis2005

Recommended Citation
http://aisel.aisnet.org/amcis2005/237

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2005 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
ABSTRACT:

Web service enabled Service Oriented Architecture has been an increasingly popular approach for enterprises to develop an agile and flexible architecture. However, very little guidance on the granularity of “services” has been provided, which has an impact on the trade-off between flexibility and coordination effort. We suggest that the lack of granularity guidance is due to the lack of the explicit positioning of “services” in the traditional enterprise architecture literature. After clarifying the relationship between “services” and “processes”, we provide an approach to define “process complexity” along certain dimensions and use this complexity metric to decide on the appropriate granularity of the “services”.

Keywords: SOA, web service, granularity, process complexity, enterprise architecture

INTRODUCTION

Facing increasing competitive and uncertain business environment, enterprises are looking for strategies and technologies to implement an agile and flexible architecture that will allow them to react quickly to market changes. SOA (Service Oriented Architecture) and web services are considered competent IT architecture and enabling technology to achieve the flexibility, based on which enterprise can connect to and optimize within the global value chain (Hagel 2002; Iyer, Freedman et al. 2003). According to researchers with IDC, spending on software related to Web services projects should hit US$11 billion by 2008, an increase from $1.1 billion in 2003. Even in conservative and risk-averse firms, web service is evolving as a critical technology (Andrews and Hotle 2003).

While there is significant discussion in the literature on web service technology improvement and standardization, such as security, coordination, and orchestration, little research is done from the business perspective, especially on identifying the right level of services to provide. According to an industrial analyst, “the tool set will get you only part of the way. … You need to know what services to build, how to build them at the right level of granularity, and how to build them loosely coupled” (Fontana 2004).

Unlike the concepts such as “processes”, and “information”, “services” in SOA is not an “inherent” concept in the traditional enterprise architecture literature. Aware of this conceptual gap, first we position “services” and SOA in the enterprise architecture in Section 2, from which we suggest that process-oriented view is applicable to “services” and SOA. In Section 3, we use an example to illustrate the importance of the granularity issue. In Section 4, based on a multi-perspective model of processes, we define the “process complexity” to determine whether the process is at the “right” level of granularity for service deployment. The last section presents a brief discussion of our future research plan.

“SERVICES” IN THE ENTERPRISE ARCHITECTURE

Enterprise architecture has been viewed as a stack of multiple architectures, mainly including organization architecture, process architecture, IT architecture, and information architecture (Lyer and Gottlieb 2004), or some variations of this stack, such as in (Sowa and Zachman 1992). How “services” fit into this architecture is based on the following analysis and is illustrated by Figure 1. IT architecture includes components such as platform technology (e.g., hardware and operating systems), network and telecommunication technologies, and core data-processing applications (Duncan 1995) (as illustrate by the 2nd box in Figure 1). The function of IT infrastructure is to connect, communicate, and integrate (as illustrated by the
Integration is roughly categorized into three types from bottom to top, information-oriented, service-oriented, and process-oriented integration (as illustrated by the 4th box in Figure 1).

As demonstrated by the two dotted lines in Figure 1, process-oriented integration can refer back to process architecture and information-oriented integration can refer back to information architecture. Based on different orientations, IT architecture can be process-oriented or information-oriented. However, the service-oriented integration lacks a corresponding artifact in enterprise architect. This missing architect is SOA. According to the mapping in Figure 1, SOA should lie between process-oriented IT architecture and information-oriented IT architecture.

Both academic and industry suggested close relationship between “processes” and “services”. For example, most practitioners argue that the distinction between “process” and “service” is vague. Many researchers thought “process-oriented” is one important feature of web services. Hence, we summarize that “process” is a general term to describe activities and tasks in an enterprise, and “service” is used when we specifically refer to an environment where processes need to communicate with each other. Thus, our discussion of the granularity of services is based on an understanding of processes.

**AN ILLUSTRATION EXAMPLE**

The example discussed here represents a set of processes used in selecting a “source” for procuring material or an intermediate product as a part of supply chain management activity. Figure 2 provides a multi-level view of processes, with level 1 and level 2 processes extracted from a standard set of processes defined in the Supply Chain Operational Reference (SCOR) model, created by the Supply Chain Council (2004). In SCOR, the Level 1 processes include Plan, Source, Make, Deliver and Return, and a Level 1 process “Source” has nine Level 2 processes as show in the figure. The level 3 activities listed below came from early research on supply chain within the automotive sector (Aigbedo and Tanniru 2003).

Many level 3 activities may be mapped to each of the level 2 processes, and such a mapping depends on many factors. For example, if a firm pre-selects a vendor for commodity products, then the activities used to fill a requisition are A1, A4 and A18. If the goal is to select supplier for a strategic product, the firm may go through many activities such as A1, A4, A6-A10. Once a scenario for meeting the needs of a normal supply chain operation is complete, a firm may develop other scenarios to address contingencies such as disruptions in the supply chain. Figure 3 provides many alternate paths to execute activities under various contingencies.

In order to support all these contingencies, each lower level activity can be represented as an independent program module or a service, and these are invoked in certain sequence (a process model) to meet a particular request or a contingency. If the service is defined at this activity level or below, then various requests to address contingencies have to select alternate paths to execute these activities. This calls for an agent (human or computer) to coordinate and control the execution of these services. The cost of such coordination can be high when several activities are needed together for most normal supply chain operations. If, on the other hand, the service is defined at a higher level (a group of activities, shown in blocks, defined as
composite process) or at a level 2 process (e.g. “select supplier”), then a minor disruption in the supply chain (e.g. delay in shipment) may call on many more activities than are needed to address this contingency.

**Figure 2: Illustration of the Granularity in Process Decomposition**

<table>
<thead>
<tr>
<th>Level 1</th>
<th>Plan</th>
<th>Source</th>
<th>Make</th>
<th>Deliver</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Technical Evaluation</td>
<td>Select Supplier</td>
<td>Receive Product</td>
<td>Transfer Product</td>
<td>Authorize Payment</td>
</tr>
<tr>
<td></td>
<td>Identify Source</td>
<td>Source Negotiation</td>
<td>Schedule Delivery</td>
<td>Verify Product</td>
<td></td>
</tr>
<tr>
<td>Level 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>A18. Select Preferred Vendor</td>
<td>A17. Prepare &amp; Send POs</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3. Sequence Execution of the Activities**

The next section will take the first step to define process complexity to determine the “right” granularity for service deployment in building an SOA.

4. PROCESS COMPLEXITY FOR SERVICE

We measure process complexity based on five perspectives of a conceptual model of processes described in Stohr and Zhao (2001). The model includes functional, behavioral, informational, operational, and organizational perspectives. Those
perspectives have been considered sufficient to capture the essential dimensions of the process and have been widely cited, as in (Raghu, Jayaraman et al. 2004).

- The functional perspective simply describes what the process does. (F)
- The operational perspective specifies how this function is performed and the path it takes (the sequence of sub-tasks it takes) to complete the function. (OP)
- The behavioral perspective specifies the conditions for tasks to be executed, including the pre-conditions, trigger and post-conditions. (B)
- The informational perspective defines the data consumed and produced with respect to each activity within the process. (I)
- The organizational perspective describes the resources that are assigned to various job functions involved in completing this process. (OR)

Each of the processes in Figure 2 at various levels of granularity (at various levels 1, 2, 3 or lower) can be defined along each of these dimensions, and one may be able to assess the degree of complexity of creating a web service for this process based on the values a process takes on each of these dimensions. We will look at some early research of business process management and current development of web services to establish metrics to assess complexity on each dimension.

A. Functional Perspective

Traditional DSS literature has categorized processes as structured, semi-structured or un-structured to allow firms to decide on the degree of automation feasible. The more unstructured (or knowledge intensive) a process is, the greater the human interaction during the execution of the process. However, even in these cases, certain components of the process can be supported through automation (e.g. access to data, communication of decision, and quicker analysis of some data) and these can be supported through web services, especially when they call for information from remote locations. So, an unstructured process may require that we decompose it into lower level processes until these are either clearly structured or unstructured.

V (F): For simplicity, if we calibrate a process on a 1-10 scale (1 being highly unstructured and 10 being highly structured) and, after certain number of decompositions, only processes with a higher number are suitable for web service deployment.

B. Operational Perspective

Modularization and decomposition have been a fundamental approach in software engineering for easy development, effective maintenance and reuse. High cohesion and low coupling are two basic criteria. The greater the cohesion (among the functions associated with a module), the less complex it is when it has to be coupled with other modules. The cohesion is implicitly measured by reducing the number of inputs/outputs with other modules or by reducing the number of exceptions the module has to address. For example, both A7 and A9 may not be cohesive. In the case of A7, there is wide variability in the interfaces and in the case of A9, the process is not reusable for other domains because it has many embedded criteria for selecting a supplier.

One approach to increasing cohesion is to structure the domain knowledge into independent objects or processes so that lower level artifacts can be used with minimal change or customization (Sugumaran, Tanniru et al. 2000). For example, if A9 is further divided into two sub-activities: A9.1: define criteria and weights, and A9.2: compute the net ranking, then A9.2 is cohesive.

To determine complexity on the operational dimension, we use cohesion of the process and the metric used here is the number of interfaces with other modules or web services. The fewer it has to interact, the greater the cohesion. In a highly dynamic market environment, flexibility to react to changes takes on greater importance, and in an inter-organizational context, more inter-organizational interactions a process has, the less cohesive it is.

V (OP) 1-10, 1 being least cohesive (most number of interfaces), and 10 being most cohesive (fewer the number of external interfaces).

C. The behavioral perspective
Each process has a certain degree of sensitivity to the market dynamics. We use “process elasticity” to refer to the sensitivity of a business process to changes in a particular market. For example, as described in Section 3, the activities needed to perform “select a supplier” depends on multiple market characteristics such as customer demand, supply network disruptions, changing relationships with partners, etc.

It is critical that a firm first decides what processes in the value chain are highly elastic to the market and how it wants to address this elasticity. Creating exceptions in process is one way to meet market dynamics. More exceptions means more alternative paths within the process and this may call for further decomposition of the process to allow for addressing each path separately if possible.

\[ V(B): \quad 1-10, \quad 1 \text{ if there are many exceptions to the process, and } 10 \text{ if there are very few. Processes with a 1 may be decomposed to see if the variety can be reduced.} \]

### D. Informational Perspective

The complexity of informational perspective can be captured by the following aspects, based on our analysis of the web service user senarios published by web service architecture working group (Group 2004).

- Pattern of message exchange: one direction vs. two direction, intensive interaction vs. simple flow, intensity of interaction, additional flow of electronic form, synchronous vs asynchronous, etc
- Number of partners: one to one, one to many, one intermediary between two parties, multiple intermediary between two parties, etc
- Security issues: authentication, message integrity, etc

\[ V(I): \quad 1-10, \quad 1 \text{ for high complexity and 10 for low complexity.} \]

### E. The organizational perspective

An organization expends its resources, such as people and physical assets, to complete processes. These resources may belong to different departments. Processes consuming resources from multiple departments will meet problems, such as cost allocation and coordination, and will be difficult to modify if changes occur. Further decomposition is preferred before building the services.

An organization may consider outsourcing based on organizational perspective. Given that there are a great number of ISPs providing a big range of services, an organization may outsource a process, when both the process and the involving resources are not critical but very expensive. For example, a firm may view “selecting supplier for commodity products” as a non value-added process, which consumes too many resources such as a frequently updated database and database experts. The organization may decide to have an external party manage this process. However, it may need several different interfaces from the external partner to connect to its internal processes. In this case, the organization needs to negotiate with the external party to decide the granularity of services.

\[ V(OR): \quad 1-10, \quad 1 \text{ if the resources expended on the process involves multiple departments and 10 if they does not. Outsourcing decision will also be considered according to the importance and expense of the process and resources, and the granularity of services needs to be negotiated with external partners.} \]

### 5. A统fk Metric To Assess The Degree Of Granularity Based On Process Complexity

Based on the five different perspectives discussed in the previous section, the complexity of each process is calibrated on a scale of 1-10.

\[ \text{Process Complexity of } A(i) = \text{Weighted total of } (v(F), v(OP), v(B), v(I), \text{ and } v(OR)) \]
Based on this rating, a firm decides on the granularity of the process appropriate for developing a service within the SOA. Application of this rating system on an application is currently under the way. The research is looking to develop a more effective way to use such a rating not only to make a decision on process granularity appropriate for web service development, but also develop strategies to make the appropriate trade-offs when the rating across dimensions is conflicting in nature.

References