



Reference architecture, metamodel, and modeling principles for architectural knowledge management in information technology services

Olaf Zimmermann^{a,b,*}, Christoph Miksovics^a, Jochen M. Küster^a

^a IBM Research GmbH, IBM Research – Zurich, Säumerstrasse 4, 8803 Rüschlikon, Switzerland

^b ABB Corporate Research (since February 2012), ABB Switzerland Ltd., Segelhofstrasse 1K, 5405 Baden-Dättwil, Switzerland

ARTICLE INFO

Article history:

Received 2 December 2011

Received in revised form 28 April 2012

Accepted 1 May 2012

Available online 14 May 2012

Keywords:

Architectural decisions

Architectural principles

DSL

Knowledge management

Model-driven engineering

Outsourcing

SOA

Workflow

ABSTRACT

Capturing and sharing design knowledge such as architectural decisions is becoming increasingly important in firms providing professional Information Technology (IT) services such as enterprise application development and strategic outsourcing. Methods, models, and tools supporting explicit knowledge management strategies have been proposed in recent years; however, several challenges remain unaddressed. In this paper, we extend our previous work to overcome these challenges and to satisfy the requirements of an additional user group, presales architects that are responsible for IT service solution proposals. In strategic outsourcing, such solution proposals require complex, contractually relevant design decisions concerning many different resources such as IT infrastructures, people, and real estate. To support both presales and project architects, we define a common reference architecture and a decision process-oriented metamodel. We also present a tool implementation of these concepts and discuss their application to outsourcing proposals and application development projects. Finally, we establish twelve decision modeling principles and practices that capture the practical experience gained and lessons learned during the application of our decision modeling concepts to both proposal development and architecture design work on projects.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

The Information Technology (IT) service business comprises a diverse set of practices such as IT consulting, enterprise application development, software package integration, and strategic outsourcing.

Our previous publications focused on modeling and reusing knowledge about the design of enterprise applications that apply Service-Oriented Architecture (SOA) principles and patterns on enterprise application development projects (Zimmermann, 2011; Zimmermann et al., 2009). Since then, we have also applied these concepts to other IT services practices and technologies such as software package integration and cloud computing (Zimmermann, 2010). In this paper, we extend our focus in two ways: (1) we investigate an additional practice and sub-domain of IT services, strategic outsourcing (Holcomb and Hitt, 2007) and (2) we transfer our concepts to the early stages of the solution lifecycle (i.e., from post-contract projects to presales proposal work).

Outsourcing solution design is related and similar to the design of software-intensive systems such as SOA-based enterprise

applications; the fundamental decisions that are made during the solution design process of an outsourcing proposal are important determinants of complexity and uncertainty. Hence, outsourcing solution design can be viewed as a superset of software design. Not only software applications and software-intensive systems are in scope, but also the IT infrastructures supporting these applications and systems, as well as human resources and physical assets such as real estate. Due to the need to estimate costs accurately, an outsourcing solution design has to be detailed before contracts are signed and billable projects are initiated. The complexity of the outsourcing business would suggest to analyze requirements and client environment in great detail before quoting any exact pricing information; however, market forces dictate a highly agile design approach (Dhar and Balakrishnan, 2006). Hence, a lot of uncertainty has to be dealt with throughout the proposal work; business decisions and technical decisions go hand in hand. In addition to architects, stakeholders from other fields, including sales executives, project managers, and domain specialists (e.g., for legal matters) are also involved in the solution design (e.g., when deciding how to deal with existing software licenses, contracts, and buildings).

Architectural knowledge management (AKM) solutions for software design projects have been developed and successfully applied in recent years (Lago et al., 2009). In this paper, we leverage and extend these AKM concepts so that the pre-contract designers of strategic outsourcing solutions are empowered to manage the complexity and uncertainty that is inherent to this business domain

* Corresponding author at: ABB Corporate Research (since February 2012), ABB Switzerland Ltd., Segelhofstrasse 1K, 5405 Baden-Dättwil, Switzerland.

E-mail addresses: olaf.zimmermann@ch.abb.com (O. Zimmermann), cmi@zurich.ibm.com (C. Miksovics), jku@zurich.ibm.com (J.M. Küster).

and stage of design evolution. Specifically, we present a knowledge management solution that aims at increasing design productivity and proposal quality. This solution combines previous work in the AKM community with concepts from business process management and Domain-Specific Languages (DSLs). Our approach to presenting the solution is:

1. First we compile a set of Architecturally Significant Requirements (ASRs) that characterize the AKM needs in IT services (both in presales and on billable projects). To frame the detailed design and development work, we derive a conceptual tool *reference architecture* (i.e., a set of logical building blocks with their responsibilities and collaborations) from these ASRs next. Our reference architecture is founded on business process management concepts (Leymann and Roller, 2000), workflow patterns (van der Aalst and ter Hofsted), and our previous work (Zimmermann, 2009; Zimmermann et al., 2009).
2. To refine the design of the novel components in the reference architecture, we present a decision process-oriented *metamodel*. This metamodel defines a Domain-Specific Language (DSL) that structures the interfaces between (and shapes the internal design of) these components.
3. We feature an implementation of the reference architecture and the metamodel in the *Solution Decision Advisor (SDA)*, first introduced in Miksovich and Zimmermann (2011).
4. We report how we validated concepts and implementation during presales usage (in strategic outsourcing) and on billable projects (involving enterprise application development).
5. To ease the task of creating architectural decision knowledge, we distill twelve *modeling principles and practices* from user feedback gathered and lessons learned during our knowledge engineering activities on two projects conducted from 2006 to 2011, SDA development and SOA *Decision Modeling (SOAD)* (Zimmermann, 2011).

The combination of reference architecture, the metamodel, and the twelve modeling principles and practices for architectural knowledge management in IT services (addressing the extended scope of both presales design activities and architecture design on projects) is the core contribution of this paper.

The remainder of the paper is structured in the following way. Section 2 introduces the problem domain of knowledge management in IT services and establishes the challenges, functional requirements, and quality attributes that frame our design work. It also provides a solution outline. Section 3 then presents the reference architecture and Section 4 specifies the decision processing metamodel and DSL. Section 5 covers the implementation of our concepts in SDA, Section 6 their validation on real-world proposals/projects. Section 7 reports on the modeling practices and principles distilled from the SOAD and the SDA projects; Section 8 investigates related work. Finally, Section 9 presents our conclusions from this work and gives an outlook to future work.

2. Architectural knowledge management in IT services: domain context, challenges, and requirements

In our work, we primarily target two particularly knowledge-intensive domains and practices in IT services, enterprise application development and strategic outsourcing (SO) solution design (a secondary design goal is that all concepts are general enough to be applicable for other domains and practices). Enterprise application development and integration is investigated in detail in our previous publications, with special emphasis on the SOA style (Zimmermann, 2009; Zimmermann et al., 2009); we

highlight the specific characteristics of the SO domain now. We also emphasize the importance of pre-contract design in this context.

Decision making for strategic outsourcing (SO) proposals. Our user community in SO are solution architects and business decision makers working for presales organizations of IT service providers offering SO services. Not only software applications and software-intensive systems are in scope of such services, but also the IT infrastructures (e.g., servers, storage and networking devices, operating systems, and middleware) supporting these applications and systems, as well as human resources (e.g., help desk staff and system administrators) and physical assets such as real estate (e.g., data centers and offices).

Outsourced IT services comprise a wide range of IT infrastructure such as servers and storage, but also include labor-intensive tasks such as help desk and service management (Rowan et al., 2011). Designing such SO solutions is a complex undertaking; a large body of knowledge is required to deal with this complexity (Everest Group, 2006). Much of the required knowledge has to be gained and applied during the proposal phase (i.e., when scoping a solution prior to contract signature); deep experience in the domain is required in order to be competitive. Proposal teams may have few individual team members, but also dozens of them; many subject matter experts participate only temporarily. SO contracts typically run for several years; the preparation of a contract proposal is a project in its own right (Blackmore et al., 2009). An example of such a project is the formal response to an official request for proposal (Outsourcingsforum). Proposal parameters include technology platform types (e.g., PCs, UNIX, and mainframe), volume (e.g., number of help desk requests, servers, and business transactions) and Service Level Agreements (SLAs) (Outsourcingsforum). Two examples of SO solutions from opposite ends of the complexity spectrum are a standardized service to host a software package for a local client and a one-of-a-kind service management solution involving a number of data centers and technology platforms that have to comply with regulatory compliance requirements in multiple countries (Blackmore et al., 2009).

In their proposal work, SO solution architects have to follow a rather complex design process that investigates many design concerns, such as country- and industry-specific legal requirements, skill transfer, real estate takeover, and IT system handover (to name just a few) (Halvey and Murphy Melby, 2005). To address these concerns, a number of fundamental solution design decisions have to be made throughout the project; many of these decisions are interdependent. While many outsourcing requirements and constraints are specified explicitly, e.g., in a request for proposal, other decision drivers are less tangible, e.g., company-internal policies about intellectual property rights and the desire to reuse standard offerings in order to be able to operate in a cost-efficient manner.

What is needed in this context is knowledge management support for presales (and project) solution architects making the fundamental solution design decisions that are relevant for the following handover activities (Dhar and Balakrishnan, 2006).¹

2.1. Knowledge management challenges (proposals/projects) and example

Solution design both in enterprise application development and in SO requires numerous interrelated business decisions as well as technical decisions to be made. Due to market dynamics and the

¹ In SO, handover means that the IT service provider takes over legal responsibility and daily operations from the IT service requestor after a contract has been signed; this is followed by a series of infrastructural and organizational changes that are applied so that enterprise architecture guidelines are met and the contracted services are delivered in a cost-efficient manner.

Download English Version:

<https://daneshyari.com/en/article/461154>

Download Persian Version:

<https://daneshyari.com/article/461154>

[Daneshyari.com](https://daneshyari.com)