Chapter 8
Relational Service Quality Modeling

Vladimir A. Shekhovtsov  
National Technical University “Kharkiv Polytechnic Institute”, Ukraine

Roland Kaschek  
Information Science Research Center, New Zealand

Christian Kop  
Alpen-Adria-Universität Klagenfurt, Austria

Heinrich C. Mayr  
Alpen-Adria-Universität Klagenfurt, Austria

ABSTRACT
The paper argues that using non-first-normal-form (NF²) tables for requirements modeling is a suitable approach for communicating application system issues to stakeholders who have a business background. In particular, such tables are used in an intermediate predesign step residing between requirements elicitation and conceptual design for modeling functional and quality requirements of services and business processes. This approach extends to quality requirements of business processes used for service orchestration.

CONTEXT AND MOTIVATION
Our work focuses on the elicitation and modeling of service and business process quality requirements within the context of Service-Oriented Architecture and Process-aware Information Systems (PAIS) (Dumas, Van der Aalst, & ter Hofstede, 2005) that are facilitated by computer applications such as BPM engine (workflow management system).

Requirements concerning system quality are often referred to as non-functional requirements in order to distinguish them from what traditionally is just called requirements in the realm of information systems and only covers functional system aspects. We consider quality requirements...
Relational Service Quality Modeling

regarding a system under development (SUD) as a specification of the quality that the SUD is expected to have in order to be fit for use. Thus, featuring the right functionality, i.e. implementing the functional requirements, might be one quality aspect among others. Within the context of SOA, quality requirements are related to both individual services and orchestrations thereof, i.e. business processes.

Elicitation and modeling of quality requirements is important and challenging because:

1. System development is often driven by stated requirements rather than stakeholder needs; inappropriate quality requirements likely are going to waste resources spent for development.

2. Requirements engineering is about system branding, i.e., conceptually shaping or working out the related SUD. Using abstract design-time notions (classes, attributes, activities etc.) at that stage of the development lifecycle for describing system qualities might compromise that branding, as the chosen concepts might be inappropriate for stakeholder understanding and validation.

3. In future, automated service selection, based on automated negotiations, is likely to take place. These negotiations involve service quality characteristics such as performance, security, cost, risk, and similar. That approach obviously has the potential to fundamentally change the way web applications are created.

Our approach is built on the predesign technique described in (Kop & Mayr, 2002; Mayr & Kop, 1998; Shekhovtsov, Kop, & Mayr, 2008) which is about modeling functional requirements for information systems. Our approach aims at quality requirements as opposed to functional ones, and focuses on services and business processes. For empowering stakeholders to cope with service and business process quality issues we propose using a simple but sufficiently expressive semantic model for addressing the relevant concepts. That model is represented as a non-first-normal-form (NF²) relational model (Schek & Pistor, 1982) permitting repetition groups and nested relations. In the present context we call a NF² table glossary. Its advantage is that it allows for the specification of any relationship between any number of given concepts. In mathematics such a definition of concepts in terms of each other and relationships among them is called an implicit definition or axiomatization.

Experience from practice suggests that stakeholders having a business background with relative ease understand predesign models because they come along in the well-known shape of tables (Galle, Kop, & Mayr, 2008; Janicki, Parnas, & Zucker, 1997). The conceptual predesign model also is suitable for model mapping. For static and dynamic aspects it was shown that predesign modeling notions can be mapped into class diagrams, activity diagrams and state charts (Kop & Mayr, 2002; Mayr & Kop, 1998) or modeling notions of business process models can be derived from them (Mayr, Kop, & Esberger, 2007). The aim is to guarantee such a mapping flexibility also for quality requirements. Our approach, as suggested by (Mayr, 2006; Pastor, 2006), thus enables both: focusing on requirements and stakeholders’ needs, and model mapping onto design notation such as UML and further onto executable code.

In addition, using a unified tabular (relational) representation for requirements allows users to perform relational queries over requirements repositories using a domain-specific SQL-like language. This will be illustrated by examples later in the chapter.

The chapter is organized as follows: in the next section we discuss the foundations of our work and rationalize in favor of our approach. Then we apply our approach to capture service and process quality requirements. The paper is completed with a discussion of the related work and a conclusion.