The Quality of Knowledge: Knowledge Patterns and Knowledge Refactorings

Jörg Rech,
Fraunhofer Institute for Experimental Software Engineering, Germany

Björn Decker,
Fraunhofer Institute for Experimental Software Engineering, Germany

Eric Ras,
Fraunhofer Institute for Experimental Software Engineering, Germany

Andreas Jedlitschka,
Fraunhofer Institute for Experimental Software Engineering, Germany

Raimund L. Feldmann,
Fraunhofer USA, Center for Experimental Software Engineering, USA

ABSTRACT

Knowledge management is a relatively young discipline. Nevertheless, it has accumulated a valuable body of knowledge in the structuring of knowledge and in the design of socio-technical knowledge management systems. However, concepts to describe common, recurring patterns of how to describe, structure, interrelate, group, or manage knowledge elements are still missing. In this article, we introduce the concepts “knowledge pattern” and “knowledge antipattern” to describe best and worst practices in knowledge management, “knowledge refactoring” to improve or change knowledge antipatterns, and “quality of knowledge” to describe desirable characteristics of knowledge in knowledge management systems. The concepts are transferred from software engineering to the field of knowledge management based on our experience from several knowledge management projects.

Keywords: knowledge management best practices; knowledge management patterns; knowledge patterns; knowledge refactoring; knowledge quality; quality of knowledge

INTRODUCTION

Today, knowledge management (KM), as well as learning management (LM), consists of a multitude of models, theories, and systems comprised of valuable and recurring knowledge that waits to be reused in new KM systems. However, the quality of the knowledge gained, the technical KM system used, or the social KM method applied is neither easily evaluated nor improved. Many best practices in the form of success factors (Mathi, 2004; Morisio, Ezran & Tully, 2002; Thomas, 2006), success models (Jennex & Olfman, 2005, 2006), success measures (Wu & Wang, 2006), reference architectures for KM
systems (Davenport & Probst, 2000; Mertins, 2003), or worst practices (Fahey & Prusak, 1998) are known in KM and typically preserve knowledge about the whole KM system or initiative. But commonly accepted best practices on how to structure knowledge, how to design an interface for a KM system, or how to start a storytelling session are hard to find, and concepts to describe common, recurring patterns on how to describe, structure, interrelate, group, or manage knowledge are still missing.

During the mid-1990s the concept “design pattern” was developed in software engineering to describe best practices regarding the design of software systems in a structured way. Design patterns are used to represent knowledge that is based on experiences captured in several real-world projects and which is widely accepted. This semi-formal representation is often used for describing and presenting the gained knowledge.

In this article, we transfer the concepts quality, patterns, and refactoring from software engineering to describe best practices regarding the design of software systems in a structured way. Design patterns are used to represent knowledge that is based on experiences captured in several real-world projects and which is widely accepted. This semi-formal representation is often used for describing and presenting the gained knowledge.

In this article, we transfer the concepts quality, patterns, and refactoring from software engineering to the field of KM and introduce the concepts of knowledge patterns and knowledge refactorings in the context of knowledge quality. We describe an approach to structure knowledge in knowledge management systems in the form of knowledge patterns. These patterns and antipatterns can be used to develop KM systems and improve the quality of the systems themselves as well as that of the knowledge within (i.e., the quality of knowledge). Furthermore, we transfer the concepts of software refactoring and software quality to describe the effect of knowledge patterns as well as countermeasures (i.e., knowledge refactorings) to remove knowledge antipatterns. To illustrate the concept of knowledge patterns, we provide examples that are based on our observations from developing and operating several knowledge management systems (i.e., they do not represent empirically validated findings). Knowledge patterns state lessons learned and best practices for the structuring of knowledge, the design of KM systems, and the development of underlying ontologies. These should be kept in mind when building high-quality knowledge and KM systems. Furthermore, patterns in KM represent a way of structuring knowledge as well as a form of language that helps knowledge engineers to communicate about knowledge and KM systems. With this article we also want to stimulate the discussion about the meaning of quality in the context of KM, how knowledge should or should not be described in a KM system, and what is needed to generate a fruitful socio-technical KM system.

Relevant background information concerning KM, best practices in KM, software engineering, and software patterns are presented in the next section. Section 0 describes several desirable quality aspects of knowledge in KM systems that are affected by patterns. The core of this article—the knowledge patterns and antipatterns—are described in section 0, followed by a section about how these patterns might be implemented (c.f. section 0). Finally, we conclude and give an outlook of future work in section 0.

**BACKGROUND**

The relevant background for knowledge patterns is comprised of knowledge and learning management, software engineering and reuse, KM in software engineering, as well as patterns in software engineering. The following sections will focus on these fields and their relations to patterns in general.

**Knowledge and Learning Management**

KM and learning management both serve the same purpose: facilitating learning and competence development of individuals, in projects and in organizations, but, they follow two different perspectives. KM is related to an organizational perspective because it addresses the lack of sharing knowledge among members of the organizations by encouraging individuals to make their knowledge explicit by creating knowledge elements, which can be stored in knowledge bases for later reuse or for participating in communities of practice. Learning management emphasizes an individual perspective, as it focuses on the individual acquisition of
Supporting Knowledge Evaluation to Increase Quality in Electronic Knowledge Repositories


[www.igi-global.com/article/supporting-knowledge-evaluation-increase-quality/39089?camid=4v1a](www.igi-global.com/article/supporting-knowledge-evaluation-increase-quality/39089?camid=4v1a)