Chapter 13
Semi-Automatic Composition of Situational Methods

Anat Aharoni
Kinneret College, Israel

Iris Reinhartz-Berger
University of Haifa, Israel

ABSTRACT

Situational methods are approaches to the development of software systems that are designed and constructed to fit particular circumstances that often refer to project characteristics. One common way to create situational methods is to reuse method components, which are the building blocks of development methods. For this purpose, method components must be stored in a method base, and then retrieved and composed specifically for the situation in hand. Most approaches in the field of situational method engineering require the expertise of method engineers to support the retrieval and composition of method components. Furthermore, this is usually done in an ad-hoc manner and for pre-defined situations. In this paper, the authors propose an approach, supported by a tool that creates situational methods semi-automatically. This approach refers to structural and behavioral considerations and a wide variety of characteristics when comparing method components and composing them into situational methods. The resultant situational methods are stored in the method base for future usage and composition. Based on an experimental study of the approach, the authors show that it provides correct and suitable draft situational methods, which human evaluators have assessed as relevant for the given situations.

INTRODUCTION

Method engineering deals with the design, construction, and adaption of approaches, techniques, and tools for the development of information and software systems (Brinkkemper, 1996). Siau, Long, and Ling (2010) claim that development methods are one of the key factors for the success of information systems development. However, since projects vary in their characteristics, standard development methods in a textbook or manual may require specific adaptations so that they will support all software development properly. Situational method engineering (SME), which is a sub-field of method engineering, focuses on

DOI: 10.4018/978-1-4666-2044-5.ch013

in-house construction of organization- or project-specific development methods (Kumar & Welke, 1992; Brinkkemper, 1996; Domínguez & Zapata, 2007; Henderson-Sellers & Ralyté, 2010). The main terms used in SME are method components, situations, and situational methods (Mirbel & Ralyté, 2006; Henderson-Sellers & Ralyté, 2010). *Method components*, the building blocks of SME, are development methods, or any coherent parts of them. A *situation* can be defined as a vector of characteristics that relate to various entities in software development, such as the project in hand, the software development organization, the software development team, and so on. Finally, a *situational method* is an approach used in the development of software systems that is designed and constructed to fit particular situations.

In their review of twenty method engineering approaches, Becker, Janiesch, and Pfeiffer (2007) found five important mechanisms for composing method components into situational methods. The most utilized mechanism is *aggregation*, which combines independent method components to create a "larger" method component. This mechanism, which is also called assembly, construction, or integration, appeared in 70% of the reviewed approaches. *Specialization*, which is sometimes called tailoring, was found to be the second most popular mechanism, appearing in 45% of the approaches. *Analogy construction* (van Offenbeek & Koopman, 2006; Ralyté & Rolland, 2001; Raylte, Deneckere, & Rolland, 2003), *configuration* (Karlsson & Ågerfalk, 2005; Becker, Knackstedt, Pfeiffer, & Janiesch, 2007), and *instantiation* (Nuseibeh, 1994) were utilized much less frequently and usually in addition to aggregation or specialization. Becker et al. (2007) further claim that aggregation and specialization, which are classified by Ralyté et al. (2003) as assembly-based method engineering, can be used in a much wider variety of situations than the other mechanisms, as they provide flexible means to adapt a solution to the specific needs of a given situation.

Recently, due to the increasing number and variety of development methods and the emerging requirements of development processes (e.g., in the form of the CMMI model) (Chrissis, Konrad, & Shrum, 2003), efforts have been made to standardize the area of method engineering. These efforts have yielded the OPEN Process Framework (OPF) (Firesmith & Henderson-Sellers, 2001; Henderson-Sellers & Serour, 2005), OMG's Software Process Engineering Metamodel (SPEM) (OMG, 2005), and ISO/IEC 24744 (Gonzalez-Perez, 2007; ISO, 2007). These frameworks and approaches specify the core terminology of development methods and divide method components primarily into structural and behavioral ones. Structural method components, also called products or work products, represent the different possible artifacts in the development methods, such as documents, while behavioral method components, which are also called work units or processes, represent tasks, techniques, and activities in the development lifecycles. Other aspects, e.g., the stakeholder's involvement in software development, temporal aspects, and the language and modeling units, are also handled in these frameworks and approaches.

In order to guide the retrieval and composition of situational methods, the method components are associated with various situational characteristics, i.e., features that characterize certain situations. Examples of situational characteristics mentioned in the literature are: type of development, stakeholder cohesion or contention, project scale, distribution of project organization, domain experience of development team, degree of novelty, technical complexity, management complexity, architectural risk, incremental method evolution, company conditions, and organizational culture (Park, Na, Park, & Sugumaran, 2006; van de Weerd, Versendaal, & Brinkkemper, 2006). Mirbel and Ralyté (2006) propose what they term a 'reuse frame' for aggregating different situational characteristics relevant to a single critical development aspect. According to their proposal, situational