Construction of Situational Information Systems Management Methods

Robert Winter, Institute of Information Management, University of St. Gallen, St. Gallen, Switzerland

ABSTRACT

Situational method engineering (SME) is an established approach to create situated methods which allows the systematic construction of software artifacts while considering specific project context and goals. The author’s motivation is to investigate whether and how SME can be applied to Information Systems management (ISM), i.e., if SME concepts can be extended in order to create situated ISM methods whose application allows the systematic design of certain ISM tasks while considering context and goals. Their contribution is the proposal of a generic approach that includes such extensions and can be regarded as a SME method for ISM. For the exemplary domain of Enterprise Architecture Management (EAM), the author illustrates and demonstrates the proposed approach by (a) analyzing existing EAM solutions to discover design factors and identify solution clusters, (b) specifying to-be solution clusters and implied transition paths, and (c) deriving activity modules whose composition supports relevant transition paths and constitutes situated, context and goal specific ISM methods. For the EAM example, They document the identification of (a) eight design factors and three as-is solution clusters, the specification of (b) three to-be solution clusters and four transition paths, and the derivation of (c) five method modules that allow to be composed into four situated EAM methods.

Keywords: Engineering, Enterprise Architecture Management, Information Systems Management, Situational Method Engineering, Software Artifacts

INTRODUCTION

Method Engineering (ME) as a discipline primarily aims at the systematic construction of methods that support the development of software artifacts (Brinkkemper, 1996). However, the general ME approach can also be applied to other domains that require complex solution engineering (Bucher et al., 2008). Situational Method Engineering (SME) enhances the utility of ME by supporting not only the design of a certain, specific method, but instead a mechanism that composes method modules or configures a base method so that a method is systematically derived which is situated to the problem situation at hand (Ralytė, Brinkkemper, & Henderson-Sellers, 2007), i.e., which considers problem-specific context and goals. Our motivation is to investigate whether and how SME can be applied to Information Systems management (ISM), i.e., whether and how SME concepts can be extended in order to create...
situated ISM methods whose application allows the systematic design of certain ISM artifacts (e.g., processes, capabilities, deliverables) while considering specific context and goals. Figure 1 illustrates the assumed applicability of SME to ISM problems.

For software development as well as for ISM, SME creates base methods or method modules that are generic within a class of similar problems. A method provides problem solving guidance by proposing procedural or result templates such as role, activity, result document, or sequence specifications (Bucher & Winter, 2008). These are then instantiated by concrete actions and results. As an analogy, a management method should guide choices within the “plan-act-control” solution space of an organization in order to achieve certain (management) goals. In the domain of ISM, methods are instantiated by concrete actions and results, e.g., for determining IS strategy, planning IS architecture, determining IS project portfolio, or controlling IS projects.

However, the extent to which a (management) solution helps to achieve a (management) goal often depends on some external contingency (Donaldson, 2001). As a consequence, ISM methods are not only related to a certain (management) goal, but also to a certain (problem) context (Bucher, Klesse, Kurpjuweit, & Winter, 2007). As a consequence, SME for ISM not only deals with the construction of method modules or base methods for a certain problem range, but also with the identification of problem situations (referring to goals and contexts) and contingencies. In his Ph.D. thesis on SME for process-oriented information logistics, e.g., Bucher (2009) identifies four relevant contingencies, four resulting problem situations (goal/context combinations), 21 method fragments and five development approaches that aggregate selected fragments using a set of proposed aggregation rules.

This paper presents a generic approach to engineer situational ISM methods that is inspired by SME concepts. For the exemplary domain of Enterprise Architecture Management.

---

*Figure 1. SME for software development vs. for ISM*
Discrete Event Simulation Process Validation, Verification, and Testing
www.igi-global.com/chapter/discrete-event-simulation-process-validation/30752?camid=4v1a

Galileo Case Study: A Collaborative Design Environment for the European Space Agency's Concurrent Design Facility
Aggelos Liapis and Evangelos Argyzoudis (2013). Designing, Engineering, and Analyzing Reliable and Efficient Software (pp. 251-282).
www.igi-global.com/chapter/galileo-case-study/74885?camid=4v1a