An Ontological Framework of Method Engineering: An Overall Structure

Mauri Leppänen

Department of Computer Science and Information Systems
P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland
mauri@cs.jyu.fi

Abstract. A large number of strategies, approaches, meta models, techniques and procedures have been suggested to support method engineering (ME). Most of these artifacts, here called the ME artifacts, have been constructed, in an inductive manner, synthesizing ME practice and existing ISD methods without any theory-driven conceptual foundation. Also those ME artifacts which have some conceptual groundwork have been anchored on foundations that only partly cover ME. This paper presents an ontological framework, called OntoFrame, which can be used as a coherent conceptual foundation for the construction, analysis and comparison of ME artifacts. Due to its largeness, we describe here its modular structure composed of multiple ontologies. For each ontology, we highlight its purpose, sub-domains and theoretical foundations. We also mention the approaches and process by which OntoFrame has been constructed.

1 Introduction

Engineering of an information systems development (ISD) method is far from trivial in practice. In the first place, ISD methods are abstract things with divergent semantic and pragmatic meanings. The former implies that conceptions of what ISD methods should contain may vary substantially [11, 15, 25, 26, 30, 40]. The latter suggests that views of roles, both technical and political, that ISD methods play in ISD may be quite different [11, 72]. Existing methods also differ from one another in their fundamental assumptions and approaches [11, 30]. Second, it is often difficult to characterize the target ISD situation in a way which makes it possible to conduct a proper selection of and suitable adaptation in existing methods for an organization or a project. Third, it is frequently unclear which kind of strategies (i.e. from "scratch", integration, adaptation) and processes should be applied in each stage of the engineering of an ISD method. Fourth, most of the method engineering (ME) situations suffer from the lack of time and other resources, causing demands for carrying out ME actions in a more straightforward and effective manner.

A large array of ME strategies and approaches (e.g. [36, 58, 60]), meta models (e.g. [15, 21, 25, 32, 54]), ME techniques (e.g. [7, 33, 39, 57, 62]) and ME procedures (e.g. [21, 31]) have been suggested to support method engineering. These *ME artifacts*, as we call them here, sustain, however, several kinds of shortcomings and deficiencies [46]. One of the major limitations in them is the lack of a uniform and consistent conceptual foundation. Most of the ME artifacts have been derived, in an inductive manner, from ME practice and existing ISD methods without any theory-based conceptual ground. Also those ME artifacts that have a well-defined underpinning have been anchored on foundations that only partly cover the ME domain.

ME is a very multifaceted domain. It concerns not only ME activities, ME deliverables, ME tools, ME actors and organizational units, but, through its main outcome, an ISD method, also ISD and more specifically ISD activities, ISD deliverables, ISD actors, ISD tools, etc. Furthermore, ME involves indirectly, through information system (IS) models and their implementations, IS contexts as well as those contexts that utilize information services provided by the IS. Thus, in constructing an ME artifact it is necessary to anchor it on a coherent conceptualization that covers ME, ISD and IS, as well as the ISD and ME methods. In ontology engineering literature (e.g. [17]) a specification of the conceptualization of a domain is commonly called an *ontology*. Hence, what we need here is a coherent set of ontologies which cover all the aforementioned sub-domains of ME.

The purpose of this paper is to suggest an ontological framework, called OntoFrame, that serves as a coherent conceptual foundation for the construction, analysis and comparison of ME artifacts. OntoFrame is composed of multiple ontologies that together embrace all the sub-domains of ME. It has been constructed by searching for "universal" theoretic constructs in the literature (the *deductive* approach), by analyzing existing frameworks, reference models and meta models (the *inductive* approach), and by deriving more specific ontologies from generic ontologies above them in the framework (the *top-down* approach [70]). The construction has been directed by the goals stated in terms of extensiveness, modularity, consistency, coherence, clarity, naturalness, generativeness, theory basis and applicability. The ontological framework is quite large, comprising 16 individual ontologies [40]. Here, we are only able to describe its overall structure and outline the ontologies on a general level (Section 2). We also discuss the theoretical background and approaches followed in engineering it (Section 3). The paper ends with the discussion and conclusions (Section 4).

2. An Overall Structure of OntoFrame

A *conceptual framework* is a kind of intellectual structure which helps us determine which phenomena are meaningful and which are not. OntoFrame, as an ontological framework, aims to provide concepts and constructs by which we can conceive, understand, structure and represent relevant phenomena pertaining to method engineering. Deriving from two disciplines, ontology engineering (e.g. [13, 18]) and conceptual modeling (e.g. [9, 34]), OntoFrame has been set the following goals. To advance communication between people, the framework should be clear and natural. To balance different needs for specificity and generality, OntoFrame should be composed of ontologies that are located at different levels of generality. To build on some more stable and solid ground, the main building blocks of OntoFrame should be driven from relevant theories. To ease making extensions and still maintaining coherence and consistence, OntoFrame should be modular and generative. To cover the relevant phenomena of ME, OntoFrame should also be extensive. And last but not least, OntoFrame should be applicable in the construction, analysis and comparison of ME artifacts.

The ontological framework is composed of four main parts (Figure 1). The first main part, the core ontology, provides basic concepts and constructs to conceive and structure human conceptions of reality and use of a language in general. The second main part, the contextual ontologies, focuses on conceptualization of information processing as contexts or parts thereof, on certain processing layer, from certain perspective, and as being modeled on some model level. The third main part, the layer-specific ontologies, has been specialized from those above in the framework to cover the sub-domains of IS, ISD and ME with more special concepts and constructs. The fourth main part, the method ontologies, conceptualizes the nature, structure and contents of the ISD method and the ME method. In the following, we describe each of these main parts and their ontologies in terms of purposes, sub-domains, and theoretical foundations.

The purpose of the *core ontology* is to provide the basic concepts and constructs for conceiving, understanding, structuring and representing fundamentals of reality. It comprises seven ontologies: generic ontology, semiotic ontology, intension/extension ontology, language ontology, state transition ontology, abstraction ontology, and UoD ontology. Each of these ontologies has the scope, purpose and role of its own in the core ontology. The *generic ontology*, founded on the constructivist position [9], provides the most generic concepts from which all the other concepts have been derived by specialization. It is the top ontology [19] in our framework. The most elementary concept is ‘thing’, meaning any phenomenon in the “objective” or subjective reality. The *semiotic ontology* defines concepts that are needed to recognize the semiotic roles of and relationships between the things. The main concepts, adopted from semiotics [53], are ‘concept’, ‘sign’, and ‘referent’. The *intension / extension ontology* serves as a conceptual mechanism to specialize the notion of concept and defines its semantic meaning [22]. The notions of intension and extension enable to differentiate between ‘basic concept’, ‘derived concept’, ‘abstract concept’, ‘concrete concept’, ‘instance concept’ and ‘type concept’.

The *language ontology* provides concepts for defining the syntax and semantics of a language. Based on linguistics (e.g. [50]), it defines concepts such as ‘language’, ‘alphabet’, ‘symbol’, and ‘expression’. The *state transition ontology* is composed of concepts and constructs for the recognition of dynamic phenomena in reality in terms of states, state transitions, and events. The *abstraction ontology* serves concepts and constructs for abstraction by classification, generalization, aggregation, and grouping. Deriving from the intension/extension ontology, it also distinguishes between the first order abstraction and the second order abstraction (or the predicate abstraction). It is based on the philosophy of science and abstraction theories by e.g. [14], [22], [51] and [52]. The *UoD* (Universe of Discourse) *ontology* is composed of consolidated concepts through which reality can be conceived from a selected viewpoint. These concepts are ‘UoD state’, ‘UoD behavior’ and ‘UoD evolution’.

The *contextual ontologies* help us recognize, understand, structure and represent phenomena in reality (a) as some contexts or parts of contexts, (b) on some processing

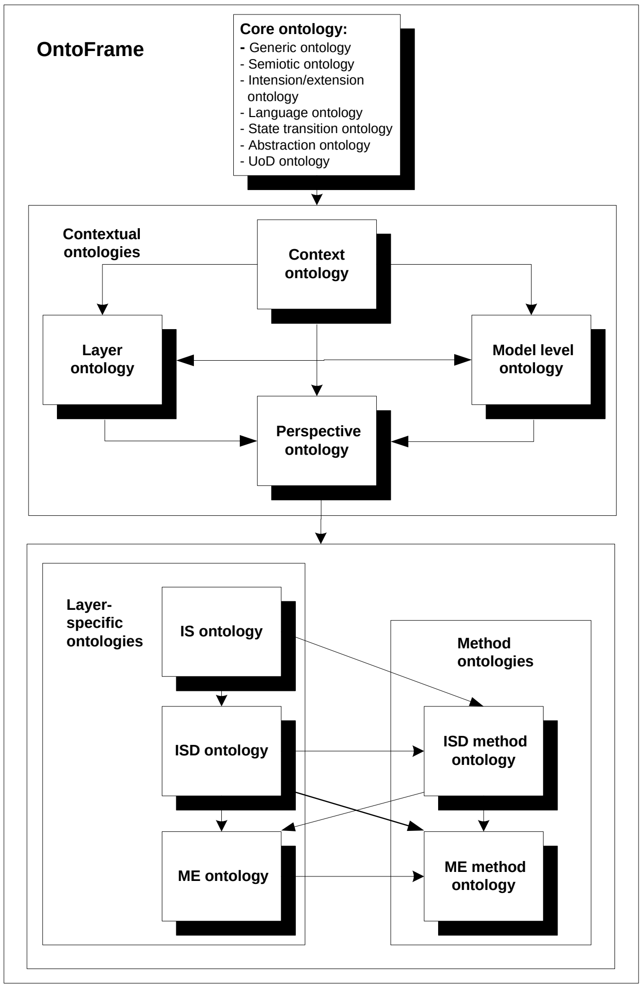

Fig. 1. Overall structure of OntoFrame

layers, (c) from some perspectives, and (d) as being modeled on some model levels. These ontologies, orthogonal to one another, are: context ontology, layer ontology, perspective ontology, and model level ontology. The *context ontology* defines seven related contextual domains, called the purpose domain, the actor domain, the action domain, the object domain, the facility domain, the location domain, and the time domain. For each domain, the most essential concepts and constructs are provided. The ontology is rooted in case grammar [10], pragmatics [48], and activity theory [8]. The *layer ontology* helps us structure and relate, on a general level, phenomena of information processing and its development at three layers, namely information systems, information systems development and method engineering. This ontology is based on systems theory [49] and information systems science (e.g. [9, 12, 26]). The *perspective ontology* provides a set of well-defined perspectives to focus and structure the conceptions of contextual phenomena. The perspectives are: systelogical, infological, conceptual, datalogical, and physical perspectives. The perspective ontology is based on systems theory [49], semiotics [55], and some seminal works (e.g. [27, 28, 73]). With the *model level ontology*, one is able to create, specify and present models about reality in different modes. The kernel of this ontology is a hierarchy composed of instance models, type models, meta models, and meta meta models. The ontology is based on works such as [6], [9] and [27], many of which have their roots in linguistics and philosophy of science.

The third main part of OntoFrame is called the *layer-specific ontologies.* While the layer ontology gives the basic structures to distinguish between and relating the information processing layers, the layer-specific ontologies elaborate the conceptualizations of IS, ISD and ME. These ontologies are the IS ontology, the ISD ontology and the ME ontology, correspondingly. Each of them is specified through the concepts and constructs of contextual domains and perspectives. The *IS ontology* helps us conceive, understand, structure, and represent phenomena in the IS, its object system and utilizing system. It has been derived from the context ontology, and by integrating constructs from multiple works (e.g. [1, 2, 24, 38, 49, 64, 68, 67]). The *ISD ontology* provides concepts for conceptualization of contextual phenomena in ISD. Besides deriving from more generic ontologies in the framework, it has been built by selecting, abstracting, modifying and integrating concepts from a large array of IS and ISD literature (e.g. [25, 35, 56, 59, 68]). Respectively, the *ME ontology* covers contextual phenomena in method engineering. The ontology has been built on works such as [3], [15], [20], [21], [36], [58], [61] and [69].

The fourth main part of the framework is the method ontologies, comprising the ISD method ontology and the ME method ontology. The ontologies provide concepts and constructs to conceive, understand, structure, and represent the nature, structure and contents of the methods. The contents of the methods are conceptualized by the ISD ontology and the ME ontology, correspondingly. The method ontologies have been structured in accordance with the semantic ladder [67] and derived from a number of frameworks of ISD methods [11, 15, 21, 25, 26, 30, 69]**.**

To summarize, OntoFrame provides a holistic view of the sub-domains involved in ME. The rationale behind the composition of OntoFrame from ontologies is based on modularity, contextuality and generativeness. Modularity and generativeness help one select an appropriate level of specificity on which phenomena in reality are to be conceived. Contextuality facilitates the use of concepts and constructs for capturing deeper meanings of single phenomena through relating them to other phenomena in the context(s). Generating more specific concepts from generic concepts by specialization advances the coherence and consistence of the framework.

3. The Approaches and Process of Engineering OntoFrame

Ontology engineering means categorizing, naming and relating phenomena in reality in an explicit way. There are two sources of ontological categories [66]: observation and reasoning. Observation provides knowledge of the physical world, and reasoning makes sense of observation by generating a framework of abstraction. OntoFrame is based on the extensive reasoning from the large literature on universal theories such as semiotics, linguistics and systems theory, and works related more specifically to IS, ISD and ME.

There are two approaches to deriving from the literature in ontology engineering. In the *inductive approach*, source material is collected from single instance-level artifacts (e.g. ontologies, frameworks, and methods) to abstract a more generic one. In the *deductive approach* some universal-like theoretic constructs are first selected and then deployed as an underlying groundwork for an ontology. We applied both of these approaches. First, in building the core ontology we made a thorough analysis of generic frameworks and ontologies (e.g. [4, 5, 9, 66, 71]) and derived the ontology from them by selection, integration, and customization. In contrast, in engineering the context ontology we first searched for disciplines and theories that address meanings in sentence contexts [10], conversation contexts [48] and action contexts [8], and derived the fundamental categorization of concepts into seven contextual domains. After that we enriched the contents and structure of each domain with constructs from existing artifacts. The fundamental structures in the perspective ontology, the model level ontology and the layer ontology were also inferred from the relevant theories (e.g. systems theory, semiotics, linguistics). For the rest of OntoFrame we mostly applied the deductive approach to generate lower-level ontologies from higher-level ontologies. In this process the existing literature was heavily utilized to complete and customize the derived concepts and constructs for the concerned sub-domains.

Many of the conceptual frameworks in the ISD literature have been constructed applying the inductive approach (e.g. [21, 63, 65]). Harmsen [21], for instance, built his MDM model (Methodology Data Model) by deriving from existing classifications and frameworks, resulting in a large set of IS-specific and ISD-specific concepts that were justified through their source artifacts. A drawback of this kind of approach is that it does not encourage bringing forward novel insights. In contrast, the BWW model [71] has been anchored in Bunge's ontology [4]. Through this deductive approach the model pursues "universality" of concepts and constructs. In the approach such as this there is, however, a risk that the theories originally crafted for different domains may not cover the whole range of phenomena in the concerned domain. We have tried to overcome these problems and risks by applying both of the approaches. Theory-based constructs provided an underpinning that was tested, enhanced and elaborated by the inductive derivation from current artifacts. The use of the theories advanced not only the soundness of the framework but also innovations.

Another way to characterize our process of engineering OntoFrame is to use the categorization of the approaches into top-down, bottom-up and mixed approaches [70]. Our process mainly proceeded in a top-down manner through the following stages: (1) building the core ontology, (2) deriving the contextual ontologies, (3) establishing the layer-specific ontologies, and (4) deriving the method ontologies (see Figure 1). From the main strategies available for ontology engineering we applied the integration strategy whenever possible. In this way we could import existing knowledge from those sub-fields in which views and concepts are relatively stable and fit our main premises. Adaptation was carried out when needed.

For each of the ontologies in the framework we applied, in an iterative manner, an ontology engineering procedure with the following steps (cf. [70]): (a) determine the purpose and domain of an ontology, (b) consider reusing existing artifacts, (c) conceptualize, (d) formalize (i.e. present in a graphical model), (e) evaluate, and (f) document.

The ontological framework is aimed at a means of communication between human beings, not for the use of computers. It comprises a large set of concepts and constructs, and most of them are highly abstract. For these reasons, it is important to present the framework in a concise yet understandable form. We deploy two presentation forms: informal and formal (cf. [70]). The concepts are defined in a vocabulary presented in a natural language. In addition, each of the ontologies is expressed in a graphical model. From a large set of semi-formal languages we selected a graphical language, and preferred the UML language to special ontology representation languages (e.g. CLEO, LINGO, DAML+OIL and OWL) because of its large and rapidly expanding user community, intrinsic mechanism for defining extensions, and largely available computer-support.

4. Discussion and Conclusions

In this paper we have described the overall structure of the large ontological framework called Ontoframe which serves as a conceptual foundation for the construction, analysis and comparison of ME artifacts. We have also brought out the purposes, sub-domains and theoretical bases of the ontologies contained in OntoFrame, as well as the approaches and process by which OntoFrame has been constructed.

As far as we know [40], there is no other presentation that would cover such a large spectrum of sub-domains, on such a detailed level, as OntoFrame does. We have intentionally aspired after this kind of holistic view in order to avoid the fragmentation of views and conceptions that is typical of most of the research in our field. The holistic view enables the recognition, comparison and integration of current artifacts that have been built upon more limited foundations and views. Only some of the existing representations (e.g. [9, 16, 28, 29, 71]) have been explicitly founded on some theories.

OntoFrame is of benefit to both research and practice. With the ontologies contained in OntoFrame it is possible to achieve a better understanding of the contextual phenomena in IS, ISD and ME. OntoFrame provides a reference background for scientists and professionals, thus enabling them to express themselves about matters in the concerned sub-domains in a structured way, and based on that, to analyze, compare and construct ME artifacts. The comprehensive and unified framework establishes bridges between various approaches, disciplines and views across three decades. OntoFrame also provides teachers and students with a large collection of models of sub-domains and a comprehensive vocabulary with clear definitions. OntoFrame is very large. It is not our intention that the whole range of ontologies is applied in every case. Contrary to that, relevant parts of OntoFrame should be selected depending on the problem at hand.

Assessment of a large ontological framework such as OntoFrame is difficult. Due to the space limit, we discuss it here on a general level in terms of the goals set up for OntoFrame. Coherence, generativeness, modularity, and balance between specificity and generality of OntoFrame has been basically advanced by applying the top-down approach by which concepts and constructs of ontologies on lower levels have been derived by specialization from those on higher levels. Extensiveness has been aspired at by anchoring the skeleton of the framework in universal theories. Achievement of coherence and consistency has been aided by cross-checking the definitions in the vocabulary and using graphical models to represent the ontologies. Clarity, naturalness and applicability are goals that should be validated, preferably through empirical tests.

Validation of applicability should, in the first place, involve each of the ontologies in the framework individually. We have responded to this demand with the discussion of validity of individual ontologies in separate articles (i.e. the context ontology [44], the ISD ontology [43, 46], the abstraction ontology [45], the perspective ontology [47] and the model level ontology [41]). Second, validation should concern the framework as the whole. Some work for that has also been done. The applicability of OntoFrame has been demonstrated in the analysis of ME artifacts in [42]. In [40] we have deployed OntoFrame to construct methodical support for ME in the form of a methodical skeleton. Although some evidence of the applicability of individual ontologies, as well as of the whole framework has been got, more experience from the use of OntoFrame in different kinds of ME contexts is definitely needed.

References

1. Allen, J. 1984. Towards a general theory of action and time. *Artificial Intelligence* 23(2), 123-154.
2. Borgo, S., Guarino, N., Masolo C. 1996. Towards an ontological theory of physical objects. In *Proc. of IMACS-IEEE/SMC Conference on Computational Engineering in Systems Applications (CESA'96), Symposium of Modelling, Analysis and Simulation*, Lille, France, 535-540.
3. Brinkkemper, S. 1990. *Formalization of information systems modeling*. Dissertation Thesis, University of Nijmegen, Amsterdam: Thesis Publishers.
4. Bunge, M. 1977. *Treatise on basic philosophy, Vol. 3: Ontology I: The furniture of the world*. Dortrecht: D. Reidel Publishing Company.
5. Chisholm, R. 1996. *A realistic theory of categories – an essay on ontology*. 1st edition, Cambridge: Cambridge University Press.
6. Dietz, J. 1987. Modelling and specification of information systems (In Dutch: Modelleren en specificeren van informatiesystemen). Dissertation Thesis, Technical University of Eindhove, The Netherlands.
7. Dominguez, E., Zapata, M.A. 2007. Noesis: Towards a situational method engineering technique. *Information Systems* 32, 181-222.
8. Engeström, Y. 1987. *Learning by expanding: an activity theoretical approach to developmental research.* Helsinki: Orienta-Konsultit.
9. Falkenberg, E:, Hesse, W., Lindgreen, P., Nilsson, B., Oei, J. L. H., Rolland, C., Stamper, R., van Asche, F., Verrijn-Stuart, A., Voss, K. 1998. *A framework of information system concepts,* The FRISCO Report (Web edition), IFIP.
10. Fillmore, C. 1968. The case for case. In E. Bach & R. T. Harms (Eds.) *Universals in Linguistic Theory.* New York: Holt, Rinehart and Winston, 1-88.
11. Fitzgerald, B., Russo, N. & Stolterman, E. 2002. *Information systems development – methods in action*. London: McGraw Hill.

12. Gasser, L. 1986. The integration of computing and routine work. *ACM Trans. on Office Information Systems* 4(3), 205-225.
13. Gomez-Perez, A., Fernandez-Lopez, O. 2004. *Ontological engineering*. London: Springer.
14. Goldstein, R., Storey, V. 1999. Data abstraction: Why and how? Data & Knowledge Engineering 29(3), 293-311.
15. Graham, I., Henderson-Sellers, B., Younessi, H. 1997. *The OPEN process specification.* Reading: Addison-Wesley.
16. Grosz, G., Rolland, C., Schwer, S., Souveyet, C., Plihon, V., Si-Said, S., Achour, C., Gnaho, C. 1997. Modelling and engineering the requirements engineering process: an overview of the NATURE approach. *Requirements Engineering* 2(2), 115-131.
17. Gruber, T. 1993. A translation approach to portable ontology specification. *Knowledge Acquisition* 5(2), 119-220.
18. Gruber, T. 1995. Towards principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies* 43(5/6), 907-928.
19. Guarino, N. 1998. Formal ontology and information systems. In N. Guarino (Ed.) *Proc. of Conf. on Formal Ontology in Information Systems (FOIS'98).* IOS Press, 3-15.
20. Gupta, D., Prakash, N. 2001. Engineering methods from method requirements specifications. *Requirements Engineering* 6(3), 135-160.
21. Harmsen, F. 1997. *Situational method engineering*. Dissertation Thesis, University of Twente, Moret Ernst & Young Management Consultants, The Netherlands.
22. Hautamäki, A. 1986. *Points of views and their logical analysis*. Helsinki: Acta Philosophica Fennica, Vol. 41.
23. Henderson-Sellers, B., Barbier, F. 1999. What is this thing called aggregation? In R. Mitchell, A. C. Wills, J. Bosch & B. MeyerProc. (Eds.) *Proc. of TOOLS EUROPE'99*. Silver Spring, MD: IEEE Computer Society Press, 236-250.
24. Herbst, H. 1995. A meta-model for business rules in systems analysis. In J. Iivari, K. Lyytinen & M. Rossi (Eds.) *Advanced Information Systems Engineering*. LNCS 932, Berlin: Springer, 186-199.
25. Heym, M., Österle, H. 1992. A reference model for information systems development. In K. Kendall, K. Lyytinen & J. DeGross (Eds.) *Proc. of Conf. on the Impacts on Computer Supported Technologies on Information Systems Development*. Amsterdam: Elsevier Science, 215-240.
26. Hirschheim, R., Klein, H., Lyytinen, K. 1995. *Information systems development – conceptual and philosophical foundations*. Cambridge: Cambridge University Press.
27. Hofstede ter, A.,Verhoef, T. 1997. On the feasibility of situational method engineering. *Information Systems* 22(6/7), 401-422.
28. Iivari, J. 1989. Levels of abstraction as a conceptual framework for an information system. In E. Falkenberg & P. Lindgren (Eds.) *Information System Concepts: An In-Depth Analysis.* Amsterdam: Elsevier Science, 323-352.
29. Iivari, J. 1990. Hierarchical spiral model for information system and software development. Part 2: Design process. *Information and Software Technology* 32(7), 450-458.
30. Iivari, J., Hirschheim, R., Klein, H. 2001. A dynamic framework for classifying information systems development methodologies and approaches. *Journal of Management Information Systems* 17(3), 179-218.
31. Karlsson, F,. Ågerfalk, P. 2004. Method configuration: adapting to situational characteristics while creating reusable assets. *Information and Software Technology* 46(9), 619-633.
32. Kelly, S., Lyytinen, K., Rossi, M. 1996. MetaEdit+: a fully configurable multi-user and multi-tool CASE and CAME environment. In Y. Vassiliou & J. Mylopoulos (Eds.) *Proc. of the 8th Conf. on Advanced Information Systems Engineering (CAiSE'96).* Berlin: Springer, 1-21.

33. Kinnunen, K., Leppänen, M. 1996. O/A matrix and a technique for methodology engineering. *Journal of Systems and Software* 33(2), 141-152.
34. Krogstie, J. 1995. *Conceptual modeling for computerized information systems support in organizations*. Dissertation Thesis, NTH, University of Trondheim, Norway.
35. Kruchten, P. 2000. *The Rational Unified Process: An introduction*. Reading: Addison-Wesley.
36. Kumar, K., Welke, R. 1992. Methodology engineering: a proposal for situation specific methodology construction. In W. Kottermann & J. Senn (Eds.) *Challenges and Strategies for Research in Systems Development*. Chichester: John Wiley & Sons, 257-269.
37. Langefors, B., Sundgren, B. 1975. *Information systems architecture*. New York: Petrocelli.
38. Lee, R. 1983. Epistemological aspects of knowledge-based decision support systems. In H. Sol (Ed.) *Proc. of Int. Conf. on Processes and Tools for Decision Support Systems*. Amsterdam: Elsevier Science, 25-36.
39. Leppänen, M. 2000. Toward a method engineering (ME) method with an emphasis on the consistency of ISD methods. In E. Siau K (Ed.) *Proc. of the Fifth Workshop on Evaluation of Modeling Methods in Systems Analysis and Design (EMMSAD'00*), Stockholm: Sweden.
40. Leppänen, M. 2005. *An Ontological Framework and a Methodical Skeleton for Method Engineering, Dissertation Thesis*. Jyväskylä Studies in Computing 52, University of Jyväskylä, Finland.
41. Leppänen, M. 2006. An Integrated Framework for Meta Modeling. In Manolopoulos Y., Pokomy J., Sellis T. (Eds.) *Proc. of Conf. on Advances in Databases and Information Systems (ADBIS'2006)*, LNCS 4152, Springer-Verlag, 141-154.
42. Leppänen, M. 2006. Conceptual evaluation of methods for engineering situational ISD methods. *Software Process: Improvement and Practice* 11(5), 539-555.
43. Leppänen, M. 2007. A Contextual Method Integration. In *Proc. of 15th Conf. on Information Systems Development – New Methods and Practices for the Networked Society (ISD 2006)*, Berlin: Springer-Verlag (in print).
44. Leppänen, M. 2007 A Context-Based Enterprise Ontology. In: Abramowicz W. (ed.) Proc. of *10th International Conference on Business Information Systems (BIS 2007*), LNCS 4439, Berlin: Springer, 273-286
45. Leppänen M. 2007. Towards an Abstraction Ontology. In Duzi M., Jaakkola H., Kiyoki Y., Kangassalo H. (Eds.) I*nformation Modelling and Knowledge Bases XVIII*, Amsterdam: IOS Press, 166-185.
46. Leppänen, M. 2007. Towards an Ontology for Information Systems Development – A Contextual Approach. In Siau K. (Ed.) *Contemporary Issues in Database Design and Information Systems Development*, Idea Group Inc. (in print).
47. Leppänen, M. 2007. A perspective ontology and IS perspectives. In *Proc. of The 17th European–Japanese Conference on Information Modelling and Knowledge Bases (EJC 2007)*, to be printed by IOS.
48. Levinson, S. 1983. Pragmatics. London: Cambridge University Press.
49. Mesarovic, M., Macko, D., Takahara, Y. 1970. *Theory of hierarchical, multilevel, systems.* New York: Academic Press.
50. Morris, C. W. 1938. Foundations of the theory of signs. In O. Neurath, R. Carnap & C. Morris (Eds.) *International Encyclopedia of Unified Science*. Chicago: University of Chicago Press, 77-138.
51. Motschnig-Pitrik, R., Kaasboll, J. 1999. Part-whole relationship categories and their application in object-oriented analysis. *IEEE Trans. on Knowledge and Data Engineering* 11(5), 779-797.
52. Motschnig-Pitrik, R., Storey, V. 1995. Modelling of set membership: the notion and the issues. *Data & Knowledge Engineering* 16(2), 147-185.
53. Ogden, C., Richards I. 1923. *The meaning of meaning*. London: Kegan Paul.

54. OMG. 2005. *Software process engineering metamodel specification*, Version 1.1, January 2005. Available at URL: <http://www.omg.org/technology/ documents/formal/ spem.htm>.
55. Peirce, C. 1955. *Philosophical writings of Peirce*, edited by J. Buchle. New York: Dover.
56. Pohl, K. 1994. The three dimensions of requirements engineering: a framework and its application. *Information Systems* 19(3), 243-258.
57. Punter, T., Lemmen, K. 1996. The MEMA-model: towards a new approach for methods engineering. *Journal of Information and Software Technology* 38(4), 295-305.
58. Ralyte, J., Deneckere, R. & Rolland, C. 2003. Towards a generic model for situational method engineering. In J. Eder & M. Missikoff (Eds.) *Proc. of the 15th Int. Conf. on Advanced Information Systems Engineering (CAiSE'03).* LNCS 2681, Berlin: Springer-Verlag, 95-110.
59. Ramesh, B., Jarke, M. 2001. Towards reference models for requirements traceability. *IEEE Trans. on Software Engineering*, 27(1), 58-93.
60. Rolland, C., Prakash, N. 1996. A proposal for context-specific method engineering. In S. Brinkkemper, K. Lyytinen & R. Welke (Eds.) *Proc. of Conf. on Method Engineering: Principles of Method Construction and Tool Support*. London: Chapman & Hall, 191-208.
61. Rolland, C., Prakash, N., Benjamen, A. 1999. A multi-model view of process modeling. *Requirements Engineering* 4(4), 169-187.
62. Saeki, M. 2003. Embedding metrics into information systems development methods: an application of method engineering technique. In Eder J., Missikiff M. (Eds.) *Proc. of the 15th Intern. Conf. on Advanced Information Systems Engineering (CAiSE 2003),* LNCS 2681, Springer-Verlag: Berlin, 374-389.
63. Saeki, M., Iguchi, K., Wen-yin, K., Shinokara, M. 1993. A meta-model for representing software specification & design methods. In N. Prakash, C. Rolland & B. Pernici (Eds.) *Proc. of Conf. on Information Systems Development Process*. Amsterdam: Elsevier Science, 149-166.
64. Simon, H. 1960. *The new science of management decisions*. New York: Harper & Row.
65. Song, X., Osterweil, L. 1992. Towards objective, systematic design-method comparison. *IEEE Software* 9(3), 43-53.
66. Sowa, J. 2000. *Knowledge representation – logical, philosophical, and computational foundations.* Pacific Grove, CA: Brooks/Cole.
67. Stamper R. 1996. Signs, information, norms and information systems. In B. Holmqvist, P. Andersen, H. Klein & R. Posner (Eds.) *Signs are Work: Semiosis and Information Processing in Organisations.* Berlin: De Gruyter, 349-392.
68. Thayer, R. 1987. Software engineering project management – a top-down view. In R. Thayer (Ed.) *Tutorial: Software Engineering Project Management*. IEEE Computer Society Press, 15-56.
69. Tolvanen, J.-P. 1998. *Incremental method engineering with modeling tools – Theoretical principles and empirical evidence*. Dissertation Thesis, Jyväskylä Studies in Computer Science, Economics and Statistics, No. 47, University of Jyväskylä, Finland.
70. Uschold, M., Gruninger, M. 1996. Ontologies: principles, methods and applications. *Knowledge Engineering Review* 11(2), 93-155.
71. Wand, Y., Weber, R. 1990. An ontological model of an information system. *IEEE Trans. on Software Engineering* 16(11), 1282-1292.
72. Wastell, D. 1996. The fetish of technique: methodology as a social defence. Information Systems Journal 6(1), 25-40.
73. Welke, R. 1977. Current information system analysis and design approaches: framework, overview, comments and conclusions for large – complex information system education. In R. Buckingham (Ed.) *Education and Large Information Systems*. Amsterdam: Elsevier Science, 149-166.