Knowledge Entry Maps: structuring of method knowledge in the IT industry

Remko Helms, Sjaak Brinkkemper (Utrecht University)

Job van Oosterum, Frank de Nijs (Centric)

institute of information and computing sciences, utrecht university
technical report UU-CS-2004-065
www.cs.uu.nl

Knowledge Entry Maps: structuring of method knowledge in the IT industry

Remko Helms, Sjaak Brinkkemper
Institute of Information and Computing Sciences
University Utrecht, The Netherlands

Job van Oosterum, Frank de Nijs
Centric
IJsselstein, The Netherlands

Abstract. For professionals in the IT industry it is important to have easy access to knowledge about the day-to-day business processes. Such knowledge, which is called method knowledge, can be made easy accessible through a knowledge repository system. One factor that determines the accessibility of method knowledge in such a repository is the way in which this knowledge is structured and made accessible. Topic maps is a relatively new technique for structuring knowledge. When it comes to accessibility, a topic map results in more intuitive searching than traditional methods such as indexes and taxonomies. This paper presents a meta-modeling technique for modeling knowledge structures that is based on topic maps. The technique is called Knowledge Entry Map and supports the design of knowledge repository systems. To validate the technique it has been applied in two case studies in the professional IT industry, a software developer and a service provider. The case studies demonstrate the applicability of the meta-modeling technique for capturing the structure of method knowledge in IT organizations. Moreover, the case studies provided the idea of a generic method knowledge structure for project-oriented and product-oriented IT organizations.

1. Knowledge in the professional IT industry

Knowledge intensive companies in the IT industry such as software developers and service providers depend on their employees for the proper execution of their business processes. Hence, knowledge about the proper execution of these processes has become a valuable asset for these companies and can give them a competitive advantage over their competitors. Consequently, companies should leverage this knowledge and make it available to their employees. Knowledge about the execution of processes is referred to in this paper as method knowledge. A method can be defined as an integrated collection of procedures, techniques, product descriptions, and tools for effective, efficient and consistent support of business processes [1], [2]. Hence, method knowledge is the explicit representation of these procedures, techniques, product descriptions, and tools. For effective leverage of method knowledge in a company it is important to share this knowledge within the organization [3]. Knowledge repository systems can be used for this purpose [4], [5], [6]. Knowledge repository systems represent a category of knowledge systems that focus on sharing explicit knowledge between members of an organization [6], [7]. Types of knowledge repository systems that are widely discussed in literature are Lessons-Learned Systems and Expertise Locators. Typically, such a system consists of a repository in which all the knowledge resources, for example electronic documents and video files, are stored and a retrieval mechanism to search for the knowledge in the repository.

The companies that are involved in this research both used a knowledge repository system. As a retrieval mechanism the companies used a keyword and/or taxonomy mechanism. However, employees in both companies experienced problems in finding the right knowledge resource using these retrieval mechanisms. This was partly caused by the fact that the amount of knowledge in the repository gradually expanded over time. Hence, searches resulted in dozens of hits. To solve this problem it was suggested to use another way of retrieval that is based on a different way of structuring knowledge and emerged from the field of knowledge engineering: topic maps [8], [9], [10]. In a topic map knowledge is structured as a network of related topics that is optimized for navigation [11], [12]. As such, it provides a more powerful aid in exploring knowledge resources in a knowledge repository. The design of a knowledge repository system that is based on a topic map requires the design of the topic map of the knowledge domain. If each individual topic is modeled, a topic map can easily contain hundreds of topics. Therefore, a meta-model is required that captures the meta-structure of the topic map [17]. For this purpose we developed the Knowledge Entry Map that is presented in this paper. To validate the applicability of Knowledge Entry Maps in designing knowledge structures they are used in two case studies. The result is a validated technique to aid companies in designing a knowledge repository system based on topic maps.

The structure of this paper is as follows. In section 2 the research method is explained, followed by the description of the Knowledge Entry Map technique in section 3. The meta-modeling technique has been evaluated in two cases and the results are described in section 4. In section 5 the results of the case studies are discussed. Section 6 describes related work and in section 7 the conclusions are presented.

2. Research approach

The main goal of the research is to develop a technique for meta-modeling of knowledge structures that is based on topic maps. For this purpose we use a Design Research approach, a widely accepted method in the field of Information Systems research. The philosophy behind Design Research is that new scientific knowledge can be generated by means of constructing an artifact [13], [14], [15]. In this case the artifact is the meta-modeling technique called Knowledge Entry Map. The steps in developing the Knowledge Entry Map according a Design Research approach are described below.

Problem awareness: Awareness was raised through several discussions with professionals responsible for organizing method knowledge in the professional IT industry as well as with their management. Both indicated problems with respect to making method knowledge available to the employees in their organization. Because management is convinced that it is important that all employees have easy access to method knowledge it is committed to improve the current situation. Studying the 'old' knowledge systems at the case study companies it was found that they are based on search and/or taxonomy mechanisms. A disadvantage of the keyword search is that a search for a particular piece of method knowledge could result in dozens of hits. The same applies for a taxonomy search; it could also result in several dozens of hits per class. Although it is an option to increase the number of levels and classes in the taxonomy, both companies did not consider this as a viable option. Because increasing the number of levels and classes would also result in a more complex structure of the taxonomy. Moreover, it was considered a disadvantage that knowledge can only be explored top-down in case of taxonomy search (section 1).

Suggestion and Development: Literature research learned that another way of structuring knowledge, based on topic maps, could be a viable alternative. The design of a knowledge repository system based on topics maps requires the design of the topic map itself. Such a topic map represents the knowledge in a specific domain, i.e., in this case the method knowledge domain, including its structure. Because such a domain can easily consist of hundreds of topics, the design of such a topic map becomes cumbersome and might even negatively influence the quality of the topic map's design. To improve the design process of the topic map, and hence the quality of the topic map, the use of a meta-model is proposed. The technique that is used for creating the meta-model is called Knowledge Entry Map (section 3). The advantage of such a meta-model is that it is smaller because it only shows the structure of the topic map instead of all the details. In the end it should result in a better design of the topic map and hence in a better accessibility of the method knowledge.

Evaluation: The Knowledge Entry Map (KEM) modeling technique has been applied in two case studies in two different IT organizations (section 4). The goal of applying the technique is to validate the applicability of the technique in capturing the structure of method knowledge. The applicability of the model was observed while participating in the process of creating the KEM. Besides observation, the people that were involved have been asked about their experiences in several interviews. Validation of the technique focused on the following three aspects: KEM should be able to model the complexity at the case study companies, KEM should be easy to learn and KEM should be easy to use (section 5).

Conclusion: Evaluating the results of the case studies resulted in suggestions for further improvement of KEM. Moreover, the KEM's of both case study companies are compared to analyze the possible existence of a generic structure for structuring method knowledge in IT organizations. The conclusions and directions for further research are formulated (section 7).

3. Knowledge Entry Map: structuring method knowledge

3.1. Structuring knowledge using topic maps

Before we present the Knowledge Entry Map it is required to introduce topics maps a little bit further using [8], [9]. The basic elements of a topic map are: topics, occurrences, and associations. Topics can be any "thing" in the real world about which knowledge needs to be stored. In other words, it is the subject that is being referred to and that is of interest in a particular knowledge domain. Examples of topics in the method knowledge domain involve the *products* that a company delivers, e.g. software products or services, the *processes* that are required to create the different products and *deliverables* that are the result of particular processes. These topics refer to things in the real world about which a company want to store and share knowledge. This knowledge is modeled as occurrences in a topic map and represent the knowledge resources that contain the knowledge. Examples of occurrences for a particular topic, e.g. a *product*, might involve a *brochure* in Word format (.doc), a *user manual* in Html format (.html) and an *instruction video* in Avi format (.avi). Finally, associations are used in a topic map to link topics to each other. These associations indicate that the topics are related to each other to some extent. The result of associating topics to each other is a network of related topics that is referred to as the knowledge structure. Besides linking topics, associations can also be used to link occurrences to one ore more topics. As such the knowledge resources are associated with the topics about which they contain knowledge.

In figure 1 this is also shown schematically. It distinguishes between a knowledge structure level and a knowledge resource level. At the knowledge structure level all topics and the associations between these topics are shown. While at the knowledge resource level all occurrences are shown. The two levels are linked to each other by associating occurrences at the knowledge resource level to topics at the knowledge structure level. This separation in levels is used to stress that the knowledge structure is separated from the knowledge resources. This has the advantage that relations between topics can be changed/updated independently of the knowledge resources. Moreover, new occurrences can be easily added after which they are associated to one or more topics. By implementing topic maps in a knowledge repository system, users of the system can locate method knowledge by browsing the topics and looking up related occurrences that contain a fragment of the total method knowledge of the company.

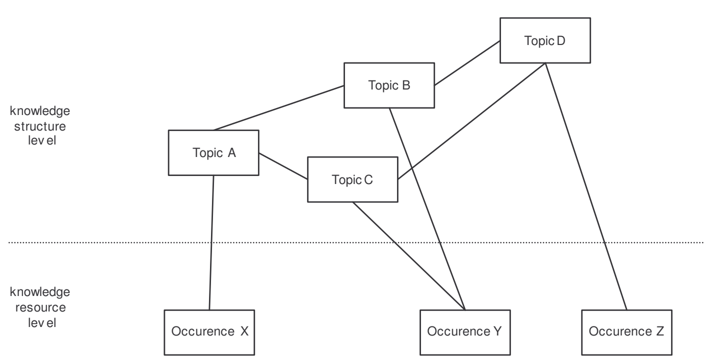

Fig. 1. Topics maps: Topics, occurrences, and associations

3.2. Knowledge Entry Map

Implementing a topic map in a knowledge repository system requires the design of the topic map. To facilitate the design process we propose a two-step approach. First, the topic map should be defined on a meta-level. The meta-level defines the types of topics and the types of associations that can be used for creating a topic map in a particular domain. Secondly, the topic map is created on instances level, by using instances of topic types that are linked to each other using instances of association types from the meta-model. Hence, the meta-model serves as a guideline in creating the topic map.

To support the creation of a meta-level topic map we developed the Knowledge Entry Map. This modeling technique is based on the modeling technique for topic maps (also known as concept maps). Consequently, topic types are modeled as boxes and association types are modeled as lines. In addition to this we introduced two new constructs: *rules* and *entry points*. Rules are restrictions to associations between particular topic types. Such restrictions are needed in the method knowledge domain to indicate that some topics cannot be created without also creating another topic of another type. For example, it might not make sense to introduce a new product in a company's method without adding the processes and deliverables that are required to produce this product. But of course these restrictions depend on the requirements of an individual company. The rules are modeled using the cardinality notation that is derived from UML class diagrams. Entry points are the most common starting for exploring the topic map. In taxonomy it is clear where to start a search, i.e., at the top. However, in the case of a topic map a search can start at any of the topics. But it can be difficult to decide where to start your search, especially for inexperienced users. Therefore, entry points are defined that indicate the most common starting point for a search.

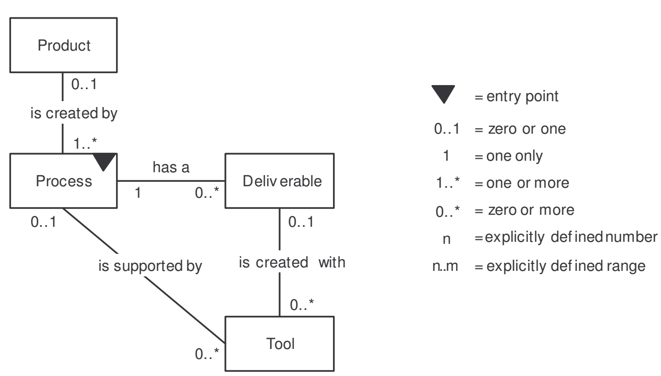

Fig. 2. Example of knowledge entry map

Figure 2 shows an example of a Knowledge Entry Map. The basic constructs in this Knowledge Entry Map are as follows:

1. **Concept**: A concept represents a category of topics with similar properties, i.e., topic types. Concepts represent an element of a method about which a company wants to store knowledge and they are typically described using a noun, examples are *Product*, *Process*, and *Deliverable*, and are represented using a box.
2. **Association**: An association represents the type of relation that can be used to link two different concepts. Associations are typically described using a verb, examples are <*is created by*> and <*is supported by*>, and are represented as lines between different concepts. Each association has two directions. Consequently, a *Product* is associated to a *Process* and vice versa. In the Knowledge Entry Map only one direction is mentioned and should be read from left-to-right or top-down. Therefore, the association between *Product* and *Process* is read as follows *Product* <*is created by*> *Process*. When reading the association the other way around it reads as follows: *Process* < *creates a*> *Product*. Finally, there is no restriction to the number of relations with other concepts.
3. **Rule**: A rule is a restriction on an association. Rules can be expressed as cardinality constraints in the KEM diagram, or should be expressed in textual form as an addendum to the KEM diagram. An example of the application of a rule using cardinality is: *Process* <*is supported by: zero or more*> *Tool*. The cardinality expression is included in the association after the ':'. In this case the rule indicates that any number of tools can be associated to a single process. For expressing the cardinality in the Knowledge Entry Map the UML notation for Class diagrams is used. An example of a rule that cannot be expressed in cardinality, and therefore should be expressed in textual form, is that "only UML compliant design tools can be associated to a process". These textual rules are not shown in the Knowledge Entry Map in order to keep it clean and simple.
4. **Entry points**: Entry points represent the most common starting points for exploring a topic map. Only concepts in a Knowledge Entry Map can be defined as entry points. In the example shown in figure 2, the *Process* concept is defined as an entry point, which is indicated using a black triangle in the upper right corner of the concept. There is no limitation to the number of entry points. Although, from a user point of view it is desirable to keep the number of entry points limited.

There are also some similarities between ontology and the Knowledge Entry Map we just presented. An ontology is defined as “a formal, explicit specification of a shared conceptualization” [16], [18], [20]. In this definition a conceptualization refers to the concepts that exist in some domain as well as to the relationships that link these concepts. A conceptualization is called ‘shared’ when people working in the domain agree that it is a fair representation of the domain. It is also called ‘explicit’ when the concepts, and the constraints on their use, are explicitly defined using some kind of modeling technique. Finally, it is also called ‘formal’ when it is machine-readable. For this purpose the Web Ontology Language has been developed [21].

Our Knowledge Entry Map can be called an ontology because we use the idea of concepts in a similar manner. It is also explicit because we developed a specific modeling technique. Moreover, it is shared because the people in the organization have to agree that the Knowledge Entry Map is a fair representation of the knowledge in the method domain. But it is not yet formal because it is not machine-readable. However, this could be realized using the Web Ontology Language [21], but that is considered to be beyond the scope of this research.

4. Using KEM to structure method knowledge – case studies

4.1. Background of case studies

In order to validate the Knowledge Entry Map technique, two case studies are conducted in two different types of companies in the professional IT industry. The first company is Baan, which is a major developer of ERP software. Around 2000 Baan was an independent company with about 5,000 employees, 8,000 customers, and 15,000 operational sites. After some downsizing it has recently been acquired by SSA Global. The Baan Research and Development unit develops the Baan products with offices in the Netherlands, India, and Germany. This department employs about 600 software engineers.

The second company is Centric, which is a provider of IT services. Its headquarters are located in the Netherlands and there are offices in Belgium and Germany as well. In total Centric has 2,750 employees, but this case study was conducted in the Managed ICT Services (MIS) division that has about 1,000 employees. The MIS division is specialized in office automation products and services, such as consultancy, helpdesk services, migration services, ICT management and Outsourcing.

4.2. KEM of Baan

The Knowledge Entry Map that is constructed by the team at Baan is presented in figure 3. We will discuss only part of the KEM in this section for illustration purposes, for a more detailed description is referred to [19].

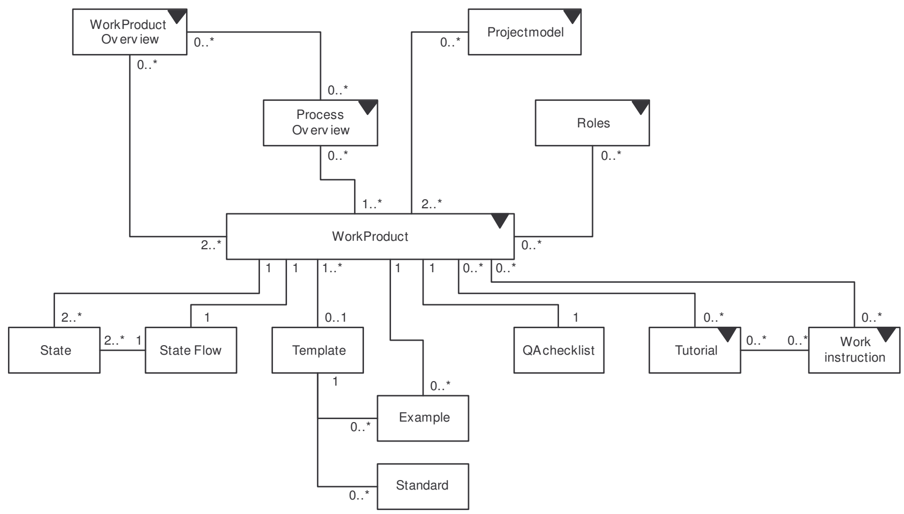

Fig. 3. Knowledge Entry Map of Baan

The main concepts in the Knowledge Entry Map at Baan are:

Project Model: A Project Model involves all deliverables that are needed to complete a project milestone. Project models are available for such things as Release delivery, Software development, Knowledge transfer, Integration test, and System test.

Process Overview: The Process Overview involves an overview of the main phases or steps in the software development process. Examples of phases are Feasibility study and Design and System Test.

Work Product: Work Products are standards for documentation of deliverables. Examples of deliverables are project plan, version definition, definition study, software unit, and test design.

Work Instructions: Work instructions are detailed procedures for the completion of well-defined smaller tasks, such as performing a software unit test, or a project audit, or risk management.

Roles: A role is a set of responsibilities that is defined within the organization. Examples of roles are Software Engineer, Test Coordinator, and Project Leader

Tutorials: Tutorials are training materials to learn how to execute certain tasks. For example a tutorial on how to do a risk assessment.

In total approximately 140 instances have been defined of the concepts that are shown in the KEM of Baan. Each of these instances has its own web page in the system at Baan. At such a page is referred to one or more knowledge resources that contain information concerning the instance. In total all the knowledge resources resemble approximately 5,000 pages of A4 at Baan Development. Moreover, each of the instances is associated to one or more other instances, which results in a network of associated web pages. There are many rules that apply to the associations and some examples of these rules are:

Rule 1: *Work Product <has: always two or more> State*

States are formally defined versions of the Work Products (WP), such that every employee in the R&D department can judge the status of the contents and impact for the ongoing job. The States are Initial, Actual, Historic plus a number of WP specific states. In the state Initial there is just an empty template or a copy from an old version for the WP. The Actual state refers to a WP in which the content is available to be used. Finally, the Historic state is used to indicate the end of the lifecycle of the WP.

Rule 2: *Work Product <can have: zero or one> Template*

In case the WP is a document, a Template for creating the WP is mandatory. If the WP is not a document (for example a change request, which is a record in a database), a Template may be omitted.

At Baan users identified the following concepts as the main starting points for their searches: Work Product Overview, Process overview, Project model, Work Product, Work Instruction, Tutorial and Role (figure 3). At Baan they build a knowledge repository, called DMethod, for managing their method knowledge that is based on the KEM shown in figure 3. It is built using html and the homepage shows the entry points as mentioned (figure 4). Each of the entry points is implemented as a drop-down list and shows for instance all the instances of Work Product in the system. When selecting an instance of Work Product, a description of that particular Work Product is shown including the associations to other instances.

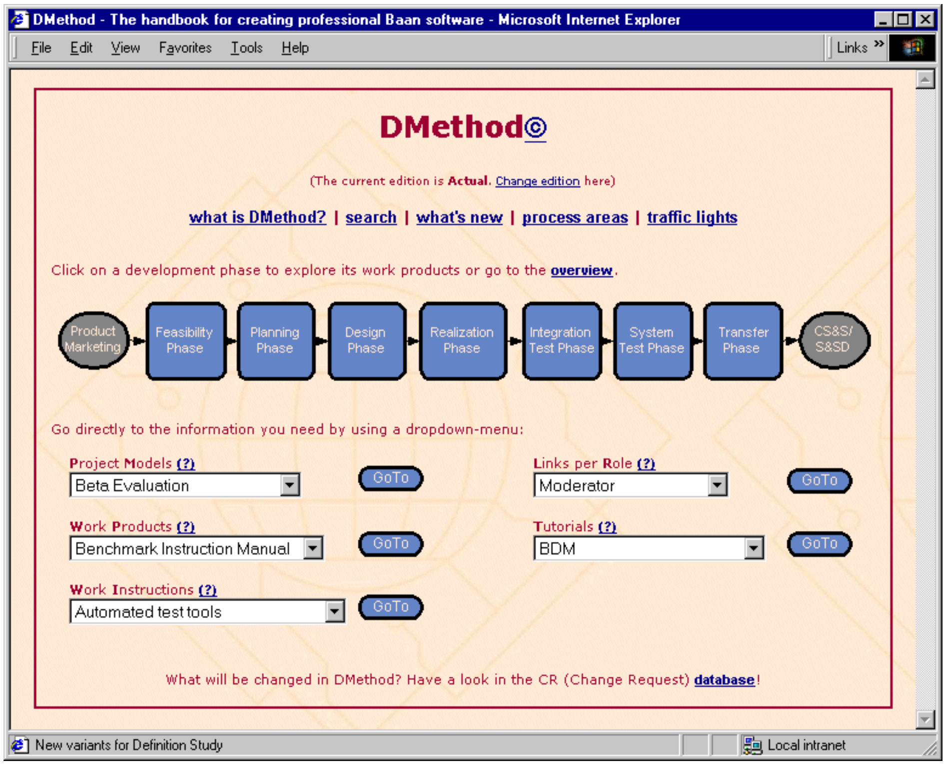

Fig 4. Homepage of DMethod

4.3. KEM of Centric

The Knowledge Entry Map that is constructed by the team at Centric is presented in figure 5. We will discuss only part of the KEM in this section for illustration purposes.

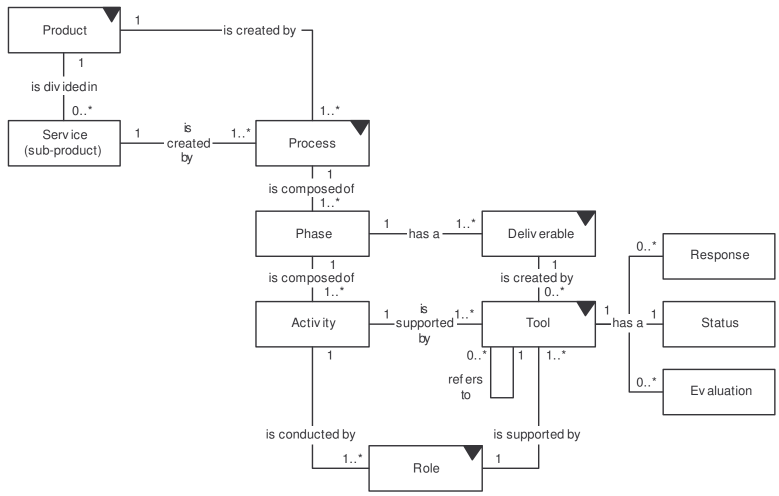

Fig. 5. Knowledge Entry Map of Centric

The main *concepts* in the Knowledge Entry Map at Centric are:

Product: These are the services that are sold to the customer, examples of services are ICT management and Helpdesk services. Centric has defined standard services that are altered for specific customers.

Process: A process is a defined set of activities to produce a specified output for a particular customer or market. An example of a process is the Execution process that is part of the ICT management product.

Phase: Phases are the defined stages or steps in a process. For example, the phases in the Execution process are Implementation preparation, Implementation and Exploitation,.

Deliverable: A deliverable is the standardized end product(s) of a phase, examples are project plan, proposal, and service level agreement.

Tool: A tool is an aid that is used to create a deliverable. Examples are a ROI spreadsheet, a proposal template, and a project plan guideline.

Role: A role is a set of responsibilities that is defined within the organization. Examples of roles are account manager, product manager, and project leader.

At Centric the old knowledge repository system that is based on Lotus Notes is being replaced by a new system. Consequently, all instances of concepts and knowledge resources from the old system are transferred to the new system. In total, the new system contains approximately 1,000 instances. Each of these instances refers to one or more knowledge resources that contain information on the associated instance. In practice, a knowledge resource is a document consisting of several pages A4. In other words, the system contains several thousands of documents, which resembles a multiple of this number in A4 pages. Also in this case, instances are associated to other instances resulting in a network of associated instances. There are many rules that apply to the associations and some examples of these rules are:

Rule 1: *Deliverable <is created by: zero or more> Tool.*

Although standard Deliverables have been defined there is not a Tool for every Deliverable available in Centric's method. An example of a Deliverable might be a data model. When there is no Tool attached to it, the user can select any Tool he likes.

Rule 2: *Tool <is supported by: exactly one> Role.*

For every Tool it should be known which Role is using the tool. In the case of Centric it is not possible to add a Tool to the method base without specifying the Role that uses it. This rule makes it possible to make an overview of all the Tools that can be used by a single Role.

The entry points of the Centric KEM are Product, Process, Deliverable, Tool, and Role (see also figure 5). During the interviews users indicated that these concepts are the main starting points for their searches. Instead of building a system based on html, like Baan, Centric is looking for commercial-off-the-shelf (COTS) software product that can support their requirements. They have developed a first prototype of a knowledge repository based on Microsoft Sharepoint Portal server. However, it was not possible to support topic maps with the system. At the moment they are evaluating if there is another COTS software products that is capable of supporting topic maps as specified in section 3.

5. Discussion

5.1. Applicability of Knowledge Entry Map

The first goal of the case studies is to study the applicability of the Knowledge Entry Map for modeling knowledge structures. Modeling these structures is one of the first steps in the development process of a knowledge repository. The applicability of KEM is evaluated on three aspects: KEM should be able to model the complexity at the case study companies, KEM should be easy to learn and KEM should be easy to use. The applicability of the model was observed while participating in the process of creating the KEM. Besides observation, the people that were involved have been asked about their experiences in several interviews. The evaluation of each of the three aspects is discussed in this section.

The first aspect of evaluation is the ability of KEM to model the complexity of the knowledge structures in both companies. In section 4 it has been demonstrated that both companies succeeded in applying KEM to model their knowledge structure on a meta-level. At Centric the basic constructs of KEM were sufficient. However, at Baan minor additions were required. We consider it minor changes because it only required an extension of the current constructs instead of introducing completely new constructs. Considering these minor changes we are convinced that KEM can also successfully applied at other companies, especially software developers and service providers. In the following, the extensions to KEM are presented:

Variants of concepts: There can be different variants of the same concept, which is illustrated using the following example. The different product lines of Baan R&D used different development platforms, which gave necessity to accommodate variants of Templates, and therefore of Work Products and Instructions. In the design phase conventional 4GL development used Entity-Relationship modeling, whereas object-oriented approaches needed Class diagrams. This resulted in a Work Product for a Functional Design in two variants, i.e., Entity-Relationship and Class diagrams. Variants are indicated in the Knowledge Entry Map by adding a colored background to a specific concept, as shown in figure 6.

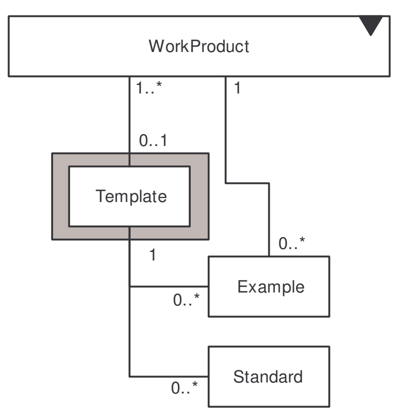

Fig. 6. Modeling variants of a concept in KEM

Conditional associations: There can be certain conditions that apply to an association, which is illustrated using the following example. At Baan we encountered the situation that an Example only needed to be associated with a Work Product if there was no Template associated with a Work Product and vice versa. In other words the existence of an association between the concepts Example and Work Product was dependent on the non-existence of an association between concepts Template and Work Product. This resulted in the introduction of the 'OR' condition on two inter-dependent associations. In the KEM this is modeled by placing a box on the involved associations. Text in this box is used to indicate the type of logical condition that applies to the associations (see figure 7).

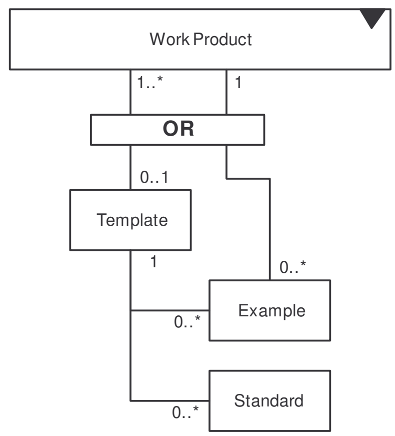

Fig. 7. Modeling of conditional associations in KEM

The second aspect of evaluation is that KEM should be easy to learn. It was observed by the researchers and indicated by the people involved that it is rather easy to learn the basics of KEM. A possible explanation is that people in the IT industry are familiar with data modeling (i.e. UML class diagrams) and therefore find it rather easy to master KEM. On the other hand, this knowledge of data modeling is actually a disadvantage. After all, KEM is *not* a data model, but a meta-level model of a knowledge structure. So after the initial (easy) introduction of KEM we had to discuss this difference in more detail with the team members so that they fully understand KEM. Also people that were not heavily involved in the project indicated that the KEM models are easy to understand.

The third aspect of evaluation is that KEM should be easy to use. The people involved in creating the Knowledge Entry Map indicated that the technique was easy to use. The Knowledge Entry Map technique does not require sophisticated diagramming tools and therefore the diagrams are easy to share with others. Moreover, the ease of use was also clear from the quality of the Knowledge Entry Map they produced, which complied with the syntax of KEM. But it was also indicated that it can be hard to identify concepts in the organization. The modeling technique itself does not prescribe how to elicit this information from the organization, except that a concept should be described using a noun. In both case studies it was decided to gather information on concepts using known techniques for business analysis, e.g., using interviews and complimentary models such as value chain models and process models. Although these are not focused on identifying concepts it provided satisfactory input for identifying them.

5.2. Generic structures of method knowledge

The second goal of the case studies is to compare the Knowledge Entry Maps of Baan and Centric to analyze if there is such a thing as a generic structure for method knowledge in the professional IT industry. Comparing both KEM's shows that there are similarities because they contain to a large extent similar concepts and associations. For example, both contain concepts such as Process (i.e. Process versus Process Overview) and Deliverable (Deliverable versus Work Product). These similarities can be caused by the fact that the case studies were conducted sequentially. However, the influence of the outcome of the first case study on the second case study is low because the business requirements of each company have been the source for constructing the KEM.

Despite the similarities that might support the possible existence of a generic structure, there are also some major differences. First of all, at Baan they use a generic description for their projects. Consequently, all projects have the same phases, milestones and work products. Therefore, a key concept in their KEM is Project Model. Centric on the other hand uses a different approach for different products. Consequently, the phases in a project depend on the type of product. Therefore, a key concept in their KEM is Product.

Secondly, studying the associations in both KEM's reveals that for Centric most of the associations are mandatory (i.e. from the type 1 or 1..*). While for Baan most of the associations are optional (i.e. from the type 0..1 or 0..*). This results in a more rigid structure for Centric and a more flexible structure at Baan. Hence, at Baan it is easier to introduce a new instance of a concept because it is not necessary to check for mandatory associations to other instances.

The fundamental difference between a project-oriented KEM at Baan and a product-oriented KEM at Centric does not support the idea that there is one generic structure that applies to all organizations in IT industry. On the other hand, it shows that the knowledge structure of a company is influenced by the way in which the company organizes its business, i.e. project-oriented vs. product-oriented. But it goes too far to say that we have found two generic structures because it is supported by only two case studies. Further research would be required to validate if we have found generic structures for project-oriented and product-oriented companies.

6. Related work

Topic maps are a relatively new field in knowledge engineering. Examples of the application of topic maps in literature are [11], [22]. In both studies topic maps are applied to the clinical domain. In one study the application is limited to modeling the topic map according the ISO standard on topic maps. While in the other study a topic map is implemented using the Ontopia Knowledge Suite (OKS).

There are several other fields in knowledge engineering that are closely related to topic maps. The related fields involve concept diagrams, associative networks, and semantic networks [6], [23], [24]. All focus on modeling knowledge structures by modeling concepts as well as associations between these concepts. In the case of semantic networks concepts are associated based on semantics. However, these techniques focus on modeling the knowledge structure and do not take into account related information resources, as is the case for topic maps.

Finally, also in the field of ontology applications can be found that are to some extent similar to that of topic maps. Like topic maps also ontologies are used for browsing information [25], [26]. In the studies ontologies are used to enable ontology based browsing for finding knowledge on the Internet. It enables the user to browse through the ontology and to find knowledge related to a specific concept. This option is offered next to full text search and attribute search, giving the user several options to find the knowledge that he needs. Another study shows how an ontology browser can be used on top of a document management system [27]. This is relevant for our research because much of the method knowledge is stored in documents. Therefore, document management functionality is also an important system requirement at the case study companies. However, there is a fundamental difference between using an ontology for browsing and using a topic map. In the case of ontology browsing is limited to browsing concepts instead of browsing instances of concepts as is the case for topic maps. Hence, topic maps result in a more detailed level of structuring and consequently browsing.

7. Conclusion

In this paper we introduced Knowledge Entry Maps as a meta-modeling technique for knowledge structures in the method knowledge domain of IT organizations. The technique supports the design process of a knowledge repository system that is based on topic maps. KEM is a technique that is based on the technique for modeling topic maps, extended with cardinality notations that are derived from the UML class diagram. Two case studies haven been conduct to validate the applicability of KEM on three different aspects: KEM should be able to model the complexity at the case study companies, KEM should be easy to learn and KEM should be easy to use.

The case studies showed that only small changes to the existing constructs of the KEM were required in order to be able to model the complexity of the structures of method knowledge at both case studies. But the case studies also revealed that further research is needed to develop a technique for identifying concepts in the organization, because both companies experienced difficulties in doing this. Furthermore, the KEM's of the two case study companies have been compared. The comparison did not provide evidence that there is a generic structure for method knowledge in professional IT organizations. Further research is needed to validate if we have found a generic structure for project-oriented organizations and a generic structure for product-oriented organizations.

Another direction for future research is the development of a knowledge repository system for sharing method knowledge in the organization. Our experience in the case studies points in the direction of document management systems, because most of the information is stored in Office documents (i.e. Pdf, Word, Excel, and Powerpoint). However, such systems should support the idea of topic maps as discussed in the paper.

References

[1] Kumar, K., Welke, R.J., Methodology Engineering: A Proposal for Situation-Specific Methodology Construction. In: W.W. Cotterman, J.A. Senn (Eds.), Challenges and Strategies for Research in Systems Development. Wiley, 1992.

[2] Brinkkemper, S., Lyytinen, K., Welke, R.J. (Eds.), Method Engineering: Prin-ciples of Method Construction and Tool Support. Chapman and Hall, 1996.

[3] Slooten, C. van, Brinkkemper, S., A Method Engineering Approach to Information Systems Development. In: Information Systems Development Process. Elsevier Science Publishers (A-30), pp. 167-186, September 1993.

[4] Davenport, T., Prusak, L., Working Knowledge – How organizations manage what they now. Harvard Business School Press, 1998.

[5] Dixon, N.M., Common Knowledge: How companies thrive by sharing what they know, Harvard Business School Press , 2000.

[6] Becerra-Fernandez, I., Gonzalez, A., Sabherwal, R., Knowledge Management – Challenges, Solutions and Technologies. Pearson/Prentice Hall, 2004.

[7] Weber, R.O., Aha, D.W., Intelligent delivery of military lessons learned. Decision Support Systems 34, pp. 287-304, 2002.

[8] International Organization for Standardization, ISO/IEC 13250, Information Technology, SGML Applications, Topic Maps, Geneva, 1998, http://www.y12.doe.gov/sgml/sc34/document/0058.htm, 30-11-2004.

[9] Pepper, S., The TAO of Topic Maps, presented at XML Europe, Paris, France, 2000, http://www.gca.org/papers/xmleurope2000/pdf/s11-01.pdf, 30-11-2004.

[10] Park, J., Hunting, S., XML Topic Maps: creating and using topic maps for the web, Addison Wesley Professional, 2002.

[11] Strauss, M., Lemke, H.U., Knowledge structure and templates for information elements in validation procedures for CARS, International Congress Series, Vol. 1256, pp. 577-583, 2003.

[12] Dong, Y., Li, M., HyO-XTM: a set of hyper graph operations on XML Topic Map toward knowledge management, Future Generation Computer Systems, Vol. 20, pp. 81-100, 2004.

[13] Guba, E. and Lincoln, Y., Competing Paradigms in Qualitative Research, in The Handbook of Qualitative Research by N. Denzin and Y. Lincoln, Thousand Oaks, CA, Sage: 105-117, 1994.

[14] March, S. and Smith, G., Design and Natural Science Research on Information Technology. Decision Support Systems 15, pp. 251 – 266, 1995.

[15] IS World, Design Research Methodology. http://www.isworld.org/Researchdesign/ drisISworld.htm, July 8, 2004.

[16] Gruber, T.R., A translation approach to portable ontologies. Knowledge Acquisition 5, pp. 199-220, 1993.

[17] Brinkkemper, S., Saeki, M., Harmsen, F., Meta-Modelling Based Assembly Techniques for Situational Method Engineering, Information Systems, Vol. 24, No. 3, pp. 209-228, 1999.

[18] Uschold, M., Gruninger, M., Ontologies: principles, methods, and application, Knowledge Engineering Review, Vol. 11, Issue 2, pp. 93-155, 1996.

[19] Brinkkemper, S., Method Engineering with Web-enabled Methods. In: Information Systems Engineering: State of the Art and Research Themes, Brinkkemper, S., Lindencrona, E., Sølvberg, A., (Eds.) pp. 123-133. Springer Verlag, London, June 2000.

[20] Benjamins, V. R., Fensel, D., Gomez Perez, A., Knowledge Management Through Ontologies. In: Proceedings of the Second International Conference on Practical Aspects of Knowledge Management (PAKM'98), Basel, Switzerland, 1998.

[21] McGuinness, D.L., Harmelen, F. van, OWL Web Ontology Language Overview, http://www.w3.org/TR/owl-features/, November 10, 2004.

[22] Schwieger, R. Hoelzer, S., Rudolf, D., Riger, J., Dudeck, J., Linking clinical data using XML topic maps, Artificial Intelligence in Medicine, Vol. 28. pp. 105-115, 2003.

[23] Gordon, J.L., Creating knowledge maps by exploiting dependent relationships, Knowledge-Based Systems, Vol. 13, pp.71-79, 2000.

[24] Curtis, M.B., Davis, M.A., Assessing knowledge structure in accounting education: an application of Pathfinder Associative Networks, Journal of Accounting Education, Vol. 21, pp. 185-195, 2003.
[25] Maedche, M., Web Information Tracking Using Ontologies. In: Proceedings of the 4th International Conference on Practical Aspects of Knowledge Management. Lecture Notes in Computer Science 2569, pp. 201-212, Springer Verlag, 2002.
[26] Spyns, P., Oberle, D., Volz, R., et al. OntoWeb - A Semantic Web Community Portal. In: Proceedings of the 4th International Conference on Practical Aspects of Knowledge Management. Lecture Notes in Computer Science 2569, pp. 189-200, Springer Verlag, 2002.
[27] Kim, H.-L., Kim, H.-G., Park, K.-M., Ontalk: Ontology-Based Personal Document Management System, In: Proceedings of 13th Interational WWW Conference, New York, May 17-22, 2004.