Association for Information Systems
AIS Electronic Library (AISeL)

ICIS 2009 Proceedings

International Conference on Information Systems (ICIS)

2009

From "Method Fragments" to "Knowledge Units": Towards a Fine-Granular Approach

Arvind Karunakaran
The Pennsylvania State University, axk969@ist.psu.edu

Sandeep Purao
The Pennsylvania State University, spurao@ist.psu.edu

Brian Cameron
The Pennysladvania State University, bcameron@ist.psu.edu

FROM "METHOD FRAGMENTS" TO "KNOWLEDGE UNITS": TOWARDS A FINE-GRANULAR APPROACH

Research-in-Progress

Arvind Karunakaran
College of Information Sciences and Technology, The Pennsylvania State University,
University Park, PA - 16802
axk969@ist.psu.edu

Sandeep Purao
College of Information Sciences and Technology, The Pennsylvania State University,
University Park, PA - 16802
spurao@ist.psu.edu

Brian Cameron
College of Information Sciences and Technology,
The Pennsylvania State University,
University Park, PA – 16802
bcameron@ist.psu.edu

Abstract

We argue that a failure to consider real-world artifacts which are involved in day-to-day 'Information Systems Development' activities as a key reason which renders approaches like method engineering inadequate to practice. We aim to reduce the abstraction and granularity of "method fragments" by re-envisioning them as 'knowledge units'. By doing so, we hope to strike the right balance between the 'fluid' and the 'institutional' domains of knowledge that can be translated into practice with relative ease. We consider real-world 'project templates' used in information systems development as exemplars of 'best practices' accumulated from the past and develop a platform called ReKon, which consists of 'fine-grain project template chunks'. We map these 'knowledge units' against broad project phases and tasks that potential users can combine as needed. These knowledge units are extracted from more than 1,200 real-world project templates made available for this research project by four leading IT consulting organizations. The paper briefly describes the theoretical foundations of the platform (Method Engineering and SECI Framework), followed by the process used for chunking and codification of the templates, and discusses results of formative evaluation of the ReKon platform. We discuss future directions for ReKon platform that require extending Nonaka and Takeuchi's "combination quadrant" within the SECI model.

Keywords: Knowledge Management, Method Engineering, SECI, Method Fragments, Knowledge Units, Granularity.

Introduction

Despite the availability of numerous methodologies for 'Information Systems Development (ISD)', organizations are confronted with issues related to comparing and selecting an appropriate methodology for a specific project (Bubenko Jr, 1986). During the early eighties, it was estimated that there were at least a few hundred methodologies available for ISD, leading to a situation referred to as the Methodology Jungle (Hofstede and Weide,1997; Avison and Fitzgerald, 1995). During the late eighties and early nineties, there was an increasing awareness about the limitations of "one-size-fits-all" approach i.e. a realization that no single method is suitable for all problem domains(Avison and Fitzgerald, 1995; Bubenko Jr, 1986; Iivari et al., 1998), the so-called "Silver Bullet Problem"(Brooks, 1987). Since then, there has been a shift in focus from a "universal method" applicable across all problem domains to coming up with approaches that would allow methods and techniques to be customized for a specific project or a task-situation (Iivari et al., 1998). One such attempt is Method Engineering (ME), defined as "the engineering discipline to design, construct and adapt methods, tools and techniques for the development of information systems" (Kumar and Welke, 1992). Method Engineering covers a broad spectrum, including integration of existing methods like ER, PetriNets and FDM (Hofstede and Verhoef, 1997; van Hee and Verkoulen, 1991), adapting a method for the needs of a particular project (also referred to as Situational Method Engineering), and deriving various 'tool-kit approaches.' Situational Method Engineering contains various degrees of flexibility, ranging from use of a rigid methodology to selection from rigid methodologies to selection of a path within a methodology to selecting and turning of a method outline to modular method construction (Harmsen et al., 1997). Typically, a Method Engineer creates a personalized method in response to the project needs.

Although the potential benefits of ME are obvious, its feasibility still remains one of the main issues (Hofstede and Verhoef, 1997). Questions concerning 'Who is the customer of Method Engineering?' continue to be raised. Mathiassen et al. (1996), for example, argue that the primary customers of method engineering are not those who are 'working with methods' but those who are 'learning the methods' (see also Mathiassen and Purao, 2002). Thus, expert practitioners are considered only in a secondary role when it comes to structuring and presenting new methods (akin to an 'expert chef' who hardly refers to 'cook books'), though they still remain the ultimate judges of the method's strengths and weaknesses (Mathiassen et al., 1996). If the primary customers of ME remain learners as opposed to experts, then it raises the ongoing concerns regarding the minor impact IS research has towards guiding practice (Baskerville and Myers, 2004; Benbasat and Zmud, 1999; Davenport and Markus, 1999; Lyytinen and King, 2004). Though ME might prove to be useful for learners, one must also reflect on the reasons concerning its inability towards creating an impact with expert practitioners. Further, ME faces two significant concerns: *coarse-granularity*, and the inherently *abstracted* nature of knowledge about methodologies, limiting the usefulness of ME in practice A motivating example helps understand this further.

A Motivating Example: Consider the following scenario faced by EntArch, Inc., who are engaged in a systems integration effort: They are deciding how to approach a new project for Painters R Us. It requires constructing a catalog of services from a scheduling application, and outlining procedures to replicate it with other legacy applications. Their concerns revolve around capturing and representing the current scheduling workflow of Painters R Us, and constructing a catalog of services from it. EntArch, Inc. have done somewhat similar projects in the past, and thought of aligning the outcomes of the project against their current processes. They consider techniques used in past projects such as ER Schemas, DFDs and Petrinets; and realize that many will need adjustments and integration with others to effectively address the project needs of Painters R Us. They are unsure about how they may be able to do this. On the other hand, they know that EntArch, Inc. has best practice templates. Ent Arch follows a process of codifying techniques and best practices used in past project in the form of Templates. These templates exemplify experiential knowledge accumulated over a period of time. Each runs into scores of pages, contains instructions and worksheets for tasks such as requirements gathering, business process modeling, database designing, and testing. The team knows that their project is likely to benefit from such Templates. They would ideally like to leverage parts from several Templates that would be best suited for this project. But their concerns are not directly addressed by the available Project Templates either. The sheer number of templates, and conflicting and overlapping choices, however, makes it difficult for them to move forward, and leverage these valuable systemic knowledge assets (Nonaka and Konno, 2005).

In situations like the one faced by EntArch, Inc, the ME mandate requires them to return to prescriptive methodologies, often ignoring the practice-related knowledge codified in Templates over a period of time. The problems faced by EntArch, Inc are not new. Templates have been traditionally used in several organizations (as opposed to Methods*)*. They represent, among other things, rules, routines and best practices accumulated from past projects (Garud and Kumaraswamy, 2005). To address their concerns better, we need to consider these artifacts which are involved in day-to-day 'Information Systems Development' activities, as opposed to abstract Methods and Method Fragments*.* They provide an alternative to the *high abstraction* and *coarse-level granularity* of method fragments, which can hinder their translation to actual practice (Karlsson and Wistrand, 2006; Tolvanen et al., 1996). We address these concerns by re-envisioning method fragments as *knowledge units*.

Our research objective, therefore, is to design a more grounded and fine-grain approach*,* which leverages past best practices contributed by expert practitioners as a realistic alternative. Our approach to addressing the problems starts with two fundamental positions: (a) casting templates as carriers of knowledge and as exemplars of accumulated best practices; and (b) positioning them as facilitators to balance between fluid and institutional domains. We map these 'knowledge units' against broad project phases and tasks that potential users, both expert as well as learners, can combine as needed. These knowledge units are extracted from more than 1,200 real-world project-related templates made available for this research project by four leading IT consulting organizations. The paper describes the theoretical foundations of the platform and process followed for chunking and codification, and preliminary formative evaluation of the ReKon platform. We discuss future research directions by adapting and extending Nonaka and Takeuchi's combination quadrant within the SECI model (Nonaka and Takeuchi, 1995).

Prior Work

This section briefly reviews research related to the problem area of information systems development. It is not meant to be a complete literature review; instead, it provides terminology and is meant to sensitize the readers to relevant constructs and approaches explored by prior research, and problematize extant literature. We review prior work in two research streams: (a) Method engineering, (b) Knowledge Management.

Method Engineering

Method engineering is an approach which aims to overcome the limitations posed by general and universally accepted text-book approaches for Information Systems Development (Brinkkemper, 1996; Brinkkemper,et al., 2008; Henderson-Sellers et al., 2005; Kumar and Welke, 1992; Qumer et al.,2008). Method*,* here, refers to a particular procedure for attaining something (Odell, 1996). Methodology*,* on the other hand, refers to the "body of methods employed by a discipline" (Odell, 1996). A method is conceived of not as a single intertwined and interdependent entity but as a set of disparate fragments (Kumar and Welke, 1992; Rossi et al., 2000; Tolvanen, 1998; Tolvanen et al., 1996; Brinkkemper, 1996). Thus, method fragments are first identified by examining existing methodologies. These method fragments are made to comply with an underlying metamodel and are also stored in a repository (Brinkkemper, 1996; Henderson-Sellers et al., 2005; Hofstede and Verhoef, 1997). These fragments are standardized by an independent body of repository inventors and are later made available to the larger IS development organization. A Method Engineer, who belongs to a particular project, has the challenge of creating a personalized method, which responds to the project needs (Brinkkemper, 1996). Method engineering advocates that ISD methods should be adapted to local situations, even if it requires significant modification of methods.

However, it needs substantial effort on the part of method engineers to examine the strengths and weaknesses of various methods, mapping these back to the characteristics of the project and the problem domain, formulating formal ways of describing method fragments and coming up with mechanisms to support the retrieval of method fragments (Hofstede and Verhoef, 1997). This, oftentimes, is impractical, because the method fragments defined are either too abstract to be mapped back to actual practice or because the method fragments involves a tight coupling between a task and a technique, leading to redundancy and integrity issues. As we saw in the motivating example, a method fragment could range from ER modelling techniques to DFDs to Petrinets, and the coarse-level granularity of these method fragments poses important challenges in the direction of making method engineering relevant to practice, in accordance to the fluid needs of the project. There were efforts in the past to re-define method fragments. For example, Hofstede and Verhoef (1997) define method fragments as a "coherent part of a metamodel, which may cover any of the modelling dimensions at any level of granularity." Still, the examples they illustrate for method fragments include a part of the meta-model dealing with Data Flow Diagrams, an ER schema that captures a part of the metamodel of the binary Entity Relationship approach and so on. Clearly, what they define as *method fragments* still remain at an abstract and coarse-granular level; and the 'artifact' involved in day-to-day ISD activities is overlooked, rendering approaches like method engineering inadequate to expert practitioners.

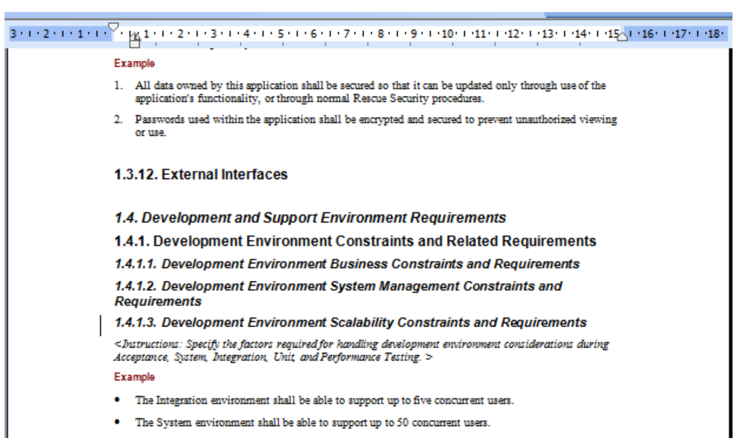
Knowledge Management

Several definitions of knowledge management (KM) have been offered in prior work, based on varying perspectives on knowledge. For the purpose of this paper, we simply define KM as concerned with identifying and leveraging the collective knowledge in an organization to help the organization compete (Von Krogh, 1998). Because of their poor track record (80% of the knowledge management systems fail to deliver (Tiwana, 2000)), there is also significant research related to enablers of KM and reasons for failure (Fahey and Prusak, 1998; Grover and Davenport, 2001; Kulkarni et al., 2007; Malhotra, 2002; Tiwana, 2000). An early framework from Nonaka and Takeuchi, popularly known as the 'SECI' framework, is useful to understand several of these in the context of an important distinction: tacit versus explicit knowledge (described as a continuous process that begins with the tacit knowledge of an individual (Nonaka, 1994; Nonaka and Takeuchi, 1995). Nonaka and Takeuchi formulate a view where knowledge creation is considered to be a continuous process that begins with the tacit knowledge of an individual (Nonaka, 1994; Nonaka and Takeuchi, 1995), and moves forward as a spiral through the phases of socialization, externalization, combination and internalization (returning to individual tacit knowledge) (Nonaka, 1994; Nonaka and Takeuchi, 1995; Nonaka and Konno, 2005). Within the context of ISD, the tacit and explicit knowledge tends to be 'mutually dependent' (Alavi and Leidner, 2005; Tuomi, 1999) and can have reinforcing qualities. These reinforcing qualities are not always exploited in the IT-dominated view of KM systems, which tend to focus more on the codifying explicit organizational knowledge (Alavi and Leidner, 2005; Schultze and Leidner, 2002) with few affordances to allow their recombination. The relation between knowledge needs of the system developers and managers involved in a project and the success/failure rate of the project is an important aspect to be considered about. Some of the inquiries conducted in this space previously were concerned more about chief executive's data needs (see, e.g. (Rockart, 1979)). Similar such inquiries for information systems development have been lacking (Lam, 2005; Popper et al., 2004; Purao et al., 2007; Brownsword et al., 2006; Boehm, 2006).

Although the SECI framework provides useful pointers in this direction, how it can be used to support the varying knowledge needs of different ISD projects within a single organization may require a re-thinking of the traditional approach to knowledge codification and knowledge reuse (Desouza, 2003). A possible approach to achieving this goal is enhancing the Combination phase (lower right quadrant of the SECI framework). This requires leveraging and some re-thinking of project templates as *carriers of knowledge*, cast as exemplars of accumulated best practices, and recasting them as the facilitators of balance between the fluid and institutional domains of knowledge (Graham and Pizzo, 1996). We elaborate this in the next section.

Proposed Solution

The research views *templates* as *carriers of knowledge* and as exemplars of accumulated best practices (Nonaka and Takeuchi, 1995; Nonaka and Konno, 2005) that can be created, stored and retrieved.



Example

1. All data owned by this application shall be secured so that it can be updated only through use of the application's functionality, or through normal Rescue Security procedures.
2. Passwords used within the application shall be encrypted and secured to prevent unauthorized viewing or use.

1.3.12. External Interfaces

1.4. Development and Support Environment Requirements

1.4.1. Development Environment Constraints and Related Requirements

1.4.1.1. Development Environment Business Constraints and Requirements

1.4.1.2. Development Environment System Management Constraints and Requirements

1.4.1.3. Development Environment Scalability Constraints and Requirements

<Instructions: Specify the factors required for handling development environment considerations during Acceptance, System, Integration, Unit, and Performance Testing. >

Example

- The Integration environment shall be able to support up to five concurrent users.
- The System environment shall be able to support up to 50 concurrent users.

Figure 1. An example of a Project Template for Gathering Requirements (source withheld)

Consider, for example, the template for specifying requirements (see Figure 1). It captures best practices for a requirements gathering task, and includes pointers that project participants can use to structure the task, including potential sub-tasks. Such templates represent the dominant form in which the templates are specified by IT consulting organizations. They represent systemic knowledge assets (Nonaka and Konno, 2005), i.e., explicit knowledge systematized and packaged in forms such as documents, specifications and manuals. They represent a key milestone in the SECI spiral that starts with tacit knowledge, obtained through work experience (experiential knowledge), articulated in symbolic form (conceptual knowledge), captured in templates (systemic knowledge). This milestone, in Nonaka and Takeuchi's conceptualization (Nonaka and Konno, 2005), is followed by moving the templates to practice via routines (routine knowledge). These templates, as systemic knowledge assets, therefore, provide a visible point of entry as carriers of knowledge. They provide the opportunity to move across the tacit-explicit as well as the individual-collective dimensions, as they contribute to the knowledge creation spiral.

Balancing the "Fluid" and the "Institutional"

The research challenge, therefore, is not solely focused on capturing the tacit dimensions of knowledge and codifying them into explicit knowledge in the form of templates. Instead, the research challenge is conceptualized as designing an approach that would balance the 'fluid' and the 'institutional' domains of knowledge (Graham and Pizzo, 1996).

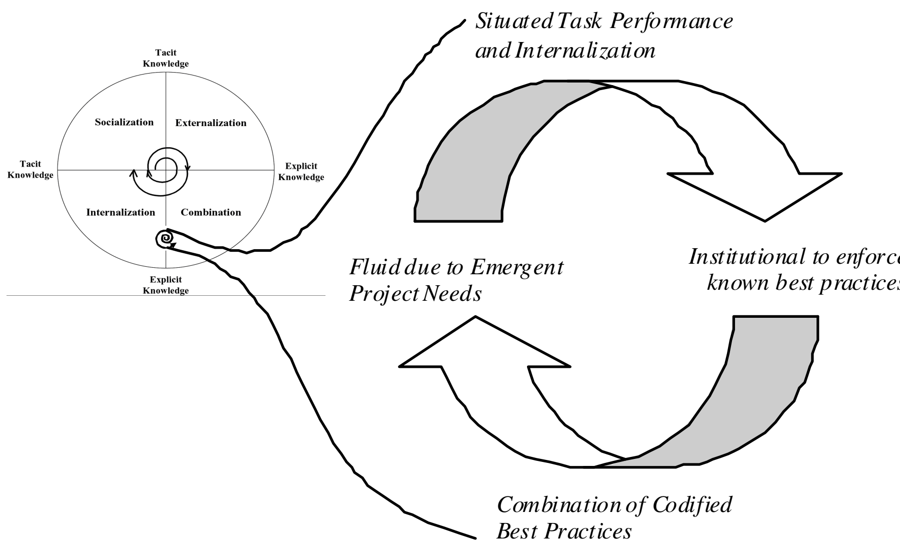


Figure 2. The Mini-Spiral within SECI

The 'fluid-ness' emphasizes individual intuition, personal networks, chance encounters or improvisation. The 'institutional' nature emphasizes knowledge as well defined in the form of standard procedures and templates. Finding a balance between these two domains (Graham and Pizzo, 1996) remains a challenge that ReKon tries to address by developing and making available recombinable and granular knowledge assets. It attempts engineer a balance between the creative (fluid) and codified (institutionalized) necessary to address the emergent knowledge needs of ISD projects (Garud et al., 2006). This balancing is conceptualized as a mini-spiral within the larger SECI spiral (see Figure 2). This mini-spiral – that straddles the Combination and Internalization quadrants – is intended to facilitate reuse of knowledge contained in the templates on specific projects. The ReKon platform, described next, therefore, includes granular knowledge units that allow such combination.

The ReKon Platform

The ReKon model builds on the above foundations. It conceptualizes components needed for ISD efforts. Before we move to the details of ReKon, we emphasize key differences between our approach and method engineering:

i. Method Engineering assumes pre-established, formally laid out methodologies that an organization will follow. It derives its methods from these methodologies. Instead of such idealization (Parnas and Clements, 1986), our approach emphasizes best practices accumulated from the past, through contributions from expert practitioners.

ii. In method engineering, Method Fragments covers a broad spectrum, including integration of existing methods (Hofstede and Verhoef, 1997), adapting a method for the needs of a particular project, etc. On the other hand, our approach is concerned lowering the abstraction and granularity by re-envisioning method fragments as knowledge units and by directing attention to real-world artifacts employed in everyday ISD activities.

iii. Method fragment, following method engineering, is identified by examining existing methodologies (Brinkkemper, 1996; Harmsen et al., 1997; Henderson-Sellers et al., 2005). Knowledge Units*,* on the other hand, are identified by examining accumulated best practices, exemplified in the form of Templates.

To populate ReKon, a large set of templates (~1220) were contributed by four leading leading IT consulting organizations. The classification of templates was facilitated by a matrix of *Phases* and *Tasks* constructed by consulting Project Management Institute's PMBOK (Duncan, 1996), and Lam and Shankarraman's 'Enterprise Integration' methodology (Lam and Shankararaman, 2004). Phases include Planning, Market Research, Requirements Gathering, Tool Comparison, Design, Development, Assessment, Implementation, Testing, and Deployment; and Tasks include IP Waiver, Status Reports, Reviews, Statement of Work, Requirements Declaration, R.F.P Development, Tool Guidelines, Client Interaction (Knowledge Transfer) and Client Interaction (Knowledge Elicitation). At the intersection of the Phases and Tasks are *Cells*, where the template chunks (called Knowledge Units) are placed. For example, a client interview protocol represents a logical knowledge unit for conducting interviews (task) during gathering requirements (phase). Interview protocols available (physical knowledge units) in multiple templates – say, for different variety of clients (e.g., SMEs, Large Enterprises), or for different types of projects (for e.g., Web Development, Legacy System Maintenance) - can be separated and made available. Project members may access these (retrieval) and combine to create new templates (target template). A random sample of 122 documents (approximately 10% of the set) was chosen for the initial classification. Prior to coding, the coders established common terms. After the first round, the classifications were compared to check consistency and differences were resolved via discussion. The common terms were enhanced based on discrepancies and the following discussion. For the second round, another coder was added, and coding was done on a separate random sample of 122 documents (another 10% of the set). Inter-coder agreement obtained during the two rounds of coding were 78% and 86% respectively and thereby, shows high levels of agreement (Cohen, 1960). The complete set (~1220 templates) were then divided and randomly assigned to coders who assessed the fit for each template to a particular cell in the matrix. Finally, the templates assigned to each cell were examined to select the best templates. This involved rules of thumb such as number of sections, thoroughness of descriptions, and availability of examples.

A Prototype Implementation

The knowledge units were used to populate the ReKon platform. The prototype contained 90 cells. Of these, knowledge units could be created for 36 cells from the templates contributed by the consulting organizations.

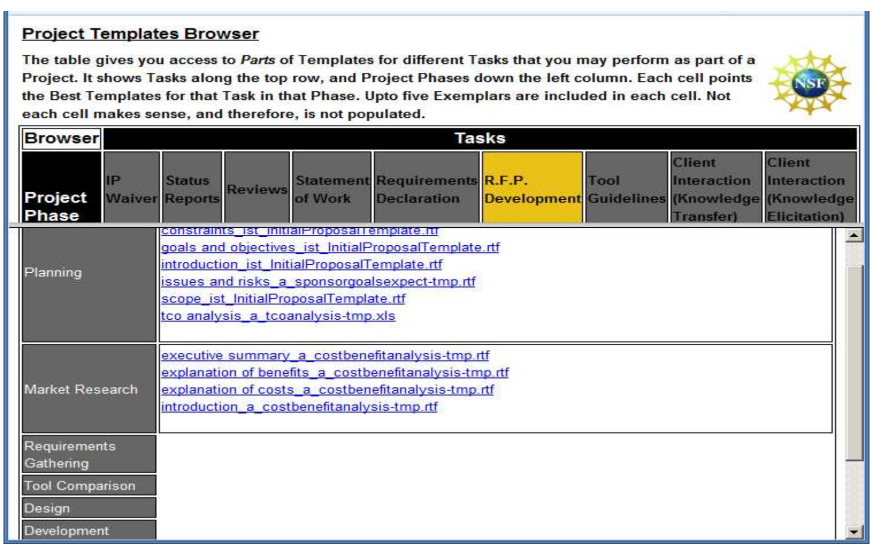


Figure 3. **ReKon Prototype Screenshot**

The total number of knowledge units in the matrix was 92, that is, an average of 2.5 knowledge units in each non-empty cell. The ReKon implementation was a conceptualized as a simple browser that would give potential users access to the Knowledge Units (Template Chunks) based on a selection of the broad project phase, and a task within that phase. Two versions of the ReKon project Template Browser were created to ensure that the front-end did not play a significant role in how it was perceived. Figure 3 shows one of the implementations. It outlines the Phases along the leftmost column, and the Tasks along the top row. Choosing a Task in the top row shows the Knowledge Units available to structure the task for each Phase. The figure shows the Task RFP development, and Knowledge Units available for this task for the Phases: Planning and Market Research. The two versions of the prototypes formed the basis for the evaluation effort.

Formative Evaluation

Evaluation was conducted in two phases, broadly described as pre- and post- ReKon. The pre-ReKon phase was carried out to assess users' evaluation of coarse-grain templates. The post-ReKon phase then assessed users' evaluation of knowledge units contained in ReKon. In the Post-ReKon evaluation, we tried to gather some formative feedback by on various elements of the platform, including granularity, size, appropriateness of classification, appropriateness of labels used in classification and relevance of knowledge units, as well as their usefulness for project needs. Users were recruited from a second course in a series (titled Advanced Enterprise Integration) engaged in working on real-world projects to implement integration solutions. The pre-ReKon evaluation, conducted prior to the introduction of the ReKon Platform assessed the use and relevance of (sections within) coarse-grain templates traditionally used. The students were also asked to comment on the comparative usefulness of a coarse-grain template versus a hypothetical scenario where different sections from the template would be available separately.

Results from the Pre-ReKon Assessment

The results from the pre-ReKon assessment provide an interesting snapshot of user perceptions. The results, (n=28) are summarized in Table 1. The responses indicate strong ambivalence for the users. The apparent paradox underscores the arguments about balancing the 'fluid' and the 'institutional' domains of knowledge.

Table 1. Coarse-Grain Templates versus Hypothetical Availability of Knowledge Units

Selected Questions	**Average** (1=agree, 5=disagree)	**Representative Comments** (Positives indicated with + ; Negatives indicated with -)
Although I may not have used all sections, it is useful to have the complete template	**1.42**	(+) … useful to have the template because most of the information we have to come up with ourselves so to have a guide line to fill in is very helpful to the success of this project (-) …it is difficult to determine if a section is relevant or not. Figuring out what needs to be included is *work in itself [emphasis added]*.
It is better to have each section available separately, so we can create the document we need by combining the sections relevant to our project.	**2.42**	(+) Most groups will not use all sections and it may be easier to make your own document. (-) … I'd rather error on the side of caution when it comes to including all possible sub-sections. Having the sections available separately poses the risk of missing something

Although some sections of the Template were found useful across the board, others were not considered relevant. Each element in the template was considered not relevant by at least one respondent. These responses provided a baseline that allowed a move to understand whether the effort to create and classify knowledge units, the ReKon platform, would be perceived to be appropriately structured. Because the users were allowed to explore the platform for a few weeks, and assessing because the platform continues to go through refinement, final assessment such as usefulness and effectiveness were not evaluated.

Results from the Post-ReKon Assessment

The post-ReKon evaluation, conducted after allowing the students a few weeks to explore the platform. The evaluation was considered formative (Weston et al., 1995) because its intent was to provide input to further improving ReKon. The post-ReKon assessment was, therefore, aimed at assessing properties such as granularity, size, appropriateness of classification, and relevance of knowledge units for project needs.

Table 2. Assessment of Knowledge Units

Criteria	Outcome (N=29)
1. Size of Knowledge Unit (1 – Too Small ; 5 – Too Long)	2.65 (SD=0.93)
2. Number of Knowledge Units in a Cell (1 – Too Many ; 5 – Too Few)	
Instance 1 – *Several units*	4.27 (SD = 0.64)
Instance 2 – *Some units*	3.27 (SD = 1.1)
3. Knowledge Needs satisfied by a Knowledge Unit (1 – All needs ; 5 – Very little Needs)	2.71 (SD = 1.01)
4. **Relevance of Knowledge Unit** (1 – All relevant ; 5 – None relevant)	2.82 (SD = 1.02)

Together, these responses not only provided formative feedback, but also added further support to the underlying proposition of the need to balance the 'fluid' and 'institutional' domains of knowledge (Graham and Pizzo, 1996).

Conclusion

In this paper, we have described an ongoing research project aimed at creating and making available fine granular yet specific knowledge to support knowledge needs of ISD. We have done this by re-directing our attention to real-world knowledge artifacts which are used to support in day-to-day ISD activities. We also argued that the coarse-level granularity of method fragments and its abstract definition hinders its translation to actual practice. To overcome this, we re-envisioned method fragments as knowledge units, deriving these from project templates. The intent was to create fine-granular approaches which could be easily translated to practice. We developed a platform called ReKon, which consists of fine-level granular template chunks, which could be recombined based on the needs of the project. The platform is conceptualized as a mini-spiral within the larger knowledge management spiral (Nonaka and Takeuchi, 1995), and facilitated by a platform that contains knowledge units constructed from project templates that can be combined, as needed, by the project participants. The fine-grain knowledge units it contains can be recombined in response to emergent situations.

Formative evaluation is carried out in two phases. The first phase establishes the ambivalence that project participants express in having access to complete templates versus granular knowledge units that they can combine as needed – a result that fits the need to balance the conflict described above. The second phase evaluates the ReKon platform to evaluate the quality, relevance, size and number of knowledge units contained in it. The combined results indicate that the fundamental ideas underlying recombinable, granular knowledge assets are likely to be valuable for meeting the knowledge needs of ISD and for balancing between the rigid/institutional and the fluid/flexible domains of knowledge.

Our future research is aimed at improving the ReKon platform, based on feedback obtained (such as adding a navigation mechanism to facilitate search and access), and longitudinal, empirical studies. We also aim to test the platform in practitioner settings and assess the potential for such fine-grain approaches for translation to practice.

Acknowledgements

The work reported has been funded by the National Science Foundation under award numbers 722112 and 722141. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF). We acknowledge the assistance of Matthew Peters, Ben Karchner and Alex Pearre in Coding and Data Analysis. We also thank the Four IT Consulting Organizations who provided us their project-related templates.

References

Alavi, M., and Leidner, D. "Knowledge management systems: issues, challenges, and benefits," *Communications of the AIS* (1:2) 1999.

Avison, D., and Fitzgerald, G. *Information systems development: methodologies, techniques and tools* Paul & Company Publishers Consortium, Inc., 1995.

Baskerville, R., and Myers, M.D. "Special Issue On Action Research In Information Systems: Making Is Research Relevant To Practice--Foreword," *MIS Quarterly* (28:3) 2004, pp 329-335.

Benbasat, I., and Zmud, R.W. "Empirical Research In Information Systems: The Practice Of Relevance," *MIS Quarterly* (23:1) 1999, pp 3-16.

Boehm, B. "Some future trends and implications for systems and software engineering processes," *Systems Engineering* (9:1) 2006, pp 1-19.

Brinkkemper, S. "Method engineering: engineering of information systems development methods and tools," *Information and Software Technology* (38:4) 1996, pp 275-280.

Brinkkemper, S., Weerd, I., Saeki, M., and Versendaal, J. "Process Improvement in Requirements Management: A Method Engineering Approach," *Lecture Notes in Computer Science* (5025) 2008, pp 6-22.

Brooks, F. "No silver bullet: Essence and accidents of software engineering," *IEEE Computer* (20:4) 1987, pp 10-19.

Brownsword, L., Fisher, D., Morris, E., Smith, J., Kirwan, P., and INST, C.-M.U.P.P.S.E. "System-of-Systems Navigator: An Approach for Managing System-of-Systems Interoperability," Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst, 2006.

Bubenko Jr, J. "Information system methodologies—a research view," North-Holland Publishing Co. Amsterdam, The Netherlands, The Netherlands, 1986, pp. 289-318.

Cohen, J. "A coefficient of agreement for nominal scales," *Educational and psychological measurement* (20:1) 1960, p 37.

Davenport, T.H., and Markus, M.L. "Rigor Vs. Relevance Revisited: Response To Benbasat And Zmud," *MIS Quarterly* (23:1) 1999, pp 19-23.

Desouza, K. "Barriers to effective use of knowledge management systems in software engineering," *Communications of the ACM* (46:1) 2003, pp 99-101.

Duncan, W. "PMBOK–A Guide to the Project Management Body of Knowledge," *ZDA: Project Management Institute (PMI)*) 1996.

Fahey, L., and Prusak, L. "The Eleven Deadliest Sins of Knowledge Management," *California Management Review* (40) 1998, pp 265-276.

Garud, R., and Kumaraswamy, A. "Vicious and Virtuous Circles in the Management of Knowledge: The Case of Infosys Technologies," *MIS Quarterly* (29:1) 2005, pp 9-33.

Garud, R., Kumaraswamy, A., and Sambamurthy, V. "Emergent by Design: Performance and Transformation at Infosys Technologies," *Organization Science* (17:2) 2006, p 277.

Graham, A., and Pizzo, V. "A question of balance: case studies in strategic knowledge management," *European Management Journal* (14:4) 1996, pp 338-346.

Grover, V., and Davenport, T. "General Perspectives on Knowledge Management: Fostering a Research Agenda," *Journal of Management Information Systems* (18:1) 2001, pp 5-21.

Harmsen, A., Ernst, M., and Twente, U. *Situational method engineering* Moret Ernst & Young Management Consultants, 1997.

Henderson-Sellers, B., Gonzalez-Perez, C., Serour, M., and Firesmith, D. "Method engineering and COTS evaluation," ACM New York, NY, USA, 2005, pp. 1-4.

Iivari, J., Hirschheim, R., and Klein, H. "A paradigmatic analysis contrasting information systems development approaches and methodologies," *Information Systems Research* (9) 1998, pp 164-193.

Karlsson, F., and Wistrand, K. "Combining method engineering with activity theory: theoretical grounding of the method component concept," *European Journal of Information Systems* (15:1) 2006, pp 82-90.

Kulkarni, U., Ravindran, S., and Freeze, R. "A Knowledge Management Success Model: Theoretical Development and Empirical Validation," *Journal of Management Information Systems* (23:3) 2007, pp 309-347.

Kumar, K., and Welke, R. "Methodology Engineering : a proposal for situation-specific methodology construction," *Challenges and Strategies for Research in SystemsDevelopment,* John Wiley,1992, pp 257-269.

Lam, W. "Investigating success factors in enterprise application integration: a case-driven analysis," *European Journal of Information Systems* (14:2) 2005, pp 175–187.

Lam, W., and Shankararaman, V. "An enterprise integration methodology," *IT professional* (6:2) 2004, pp 40-48.

Lyytinen, K., and King, J.L. "Nothing At The Center?: Academic Legitimacy in the Information Systems Field 12," *Journal of the Association for Information Systems* (5:6) 2004, pp 220-246.

Malhotra, Y. "Integrating knowledge management technologies in organizational business processes: getting real time enterprises to deliver real business performance," *Journal of Knowledge Management* (9:1) 2005, pp 7-28.

Mathiassen, L., Munk-Madsen, A., Nielsen, P., and Stage, J. "Method engineering: who's the customer?," Chapman & Hall, Ltd. London, UK, UK, 1996, pp. 232-245.

Mathiassen, L., and Purao, S. "Educating reflective systems developers," *Information Systems Journal* (12:2) 2002, pp 81-102.

Nonaka, I. "A dynamic theory of organizational knowledge," *Organization Science* (5:1) 1994, pp 14-37.

Nonaka, I., and Takeuchi, H. *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation* Oxford University Press, USA, 1995.

Nonaka, I., and Konno, N. "The Concept Of Building A Foundation For Knowledge Creation," *Knowledge Management: Critical Perspectives on Business and Management* (40:3) 2005, pp 40-54.

Odell, J. "A primer to method engineering," Chapman & Hall, Ltd. London, UK, UK, 1996, pp. 1-7.

Parnas, D., and Clements, P. "A rational design process: How and why to fake it," *IEEE Transactions on Software Engineering* (12:2) 1986, pp 251-257.

Popper, S., Bankes, S., Callaway, R., and DeLaurentis, D. "System of systems symposium: Report on a summer conversation," *Potomac Institute for Policy Studies, Arlington, VA*) 2004, pp 21–22.

Purao, S., Paul, S., and Smith, S. "Understanding enterprise integration project risks: A focus group study," 2007, pp. 850-854.

Purao, S., Storey, V., and Han, T. "Improving analysis pattern reuse in conceptual design: Augmenting automated processes with supervised learning," *Information Systems Research* (14:3) 2003, pp 269-290.

Qumer, A., and Henderson-Sellers, B. "An evaluation of the degree of agility in six agile methods and its applicability for method engineering," *Information and Software Technology* (50:4) 2008, pp 280-295.

Rockart, J. "Chief executives define their own data needs," *Harvard Business Review* (57:2), p 81.

Rossi, M., Tolvanen, J., Ramesh, B., Lyytinen, K., and Kaipala, J. "Method rationale in method engineering," 2000, p. 10.

Schultze, U., and Leidner, D. "Studying Knowledge Management in Information Systems Research: Discourses and Theoretical Assumptions," *MIS Quarterly* (26:3) 2002, pp 213-242.

Ter Hofstede, A., and Van Der Weide, T. *Formalisation of techniques: chopping down the methodology jungle* University of Nijmegen, Department of Informatics, 1997.

Ter Hofstede, A., and Verhoef, T. "On the feasibility of situational method engineering," *Information Systems* (22:6-7) 1997, pp 401-422.

Tiwana, A. *The knowledge management toolkit: practical techniques for building a knowledge management system* Prentice Hall PTR Upper Saddle River, NJ, USA, 2000.

Tolvanen, J. *Incremental method engineering with modeling tools: theoretical principles and empirical evidence* University of Jyväskylä, 1998.

Tolvanen, J., Rossi, M., and Liu, H. "Method engineering: current research directions and implications for future research," Springer, 1996, p. 296.

Tuomi, I. "Data is more than knowledge: implications of the reversed knowledgehierarchy for knowledge management and organizational memory," 1999.

van Hee, K., and Verkoulen, P. "Integration of a data model and high-level petri nets," 1991, pp. 410–431.

Von Krogh, G. "Care in knowledge creation," *California Management Review* (40) 1998, pp 133-153.

Weston, C., McAlpine, L., and Bordonaro, T. "A model for understanding formative evaluation in instructional design," *Educational Technology Research and Development* (43:3) 1995, pp 29-48.