ELSEVIER

Decision Support Systems 27 (1999) 213–235

Decision Support Systems

www.elsevier.com/locate/dsw

Supporting Collaborative Process Knowledge Management in New Product Development Teams

Balasubramaniam Ramesh [1], Amrit Tiwana *

J. Mack Robinson College of Business, Department of Computer Information Systems, Georgia State University, Atlanta, GA 30303, USA

Abstract

Knowledge centric activities of developing new products and services are becoming the primary source of sustainable competitive advantage in an era characterized by short product life cycles, dynamic markets and complex processes. We view new product development (NPD) as a knowledge-intensive activity. Based on a case study in the consumer electronics industry, we identify problems associated with knowledge management (KM) in the context of NPD by cross-functional collaborative teams. We map these problems to broad Information Technology enabled solutions and subsequently translate these into specific system characteristics and requirements. A prototype system that meets these requirements developed to capture and manage tacit and explicit process knowledge is further discussed. The functionalities of the system include functions for representing context with informal components, easy access to process knowledge, assumption surfacing, review of past knowledge, and management of dependencies. We demonstrate the validity our proposed solutions using scenarios drawn from our case study. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Knowledge management; Collaborative product development; Organizational memory; Organizational learning; New product development; Knowledge management systems; Decision support systems; Process knowledge; Design rationale

1. Introduction

As we move further into the information age, knowledge is becoming a critical component of competitive success of firms [22]. Nonaka [50] observed that, as markets shift, technologies proliferate, competitors multiply and products become rapidly obsolete, successful companies are characterized by their ability to consistently create new knowledge, quickly disseminate it, and embody it in new products and services. In the post-industrial era, Quinn et al. [54] maintain that the success of a corporation lies more deeply embedded in its intellectual systems, as knowledge based activities of developing new products and processes are becoming the primary internal functions of firms attempting to create the greatest potential for a competitive advantage [20]. Iansiti and MacCormack [35] contend that the consumer needs that a product should satisfy and technologies used in the development of such a product can change radically, even as the product is under development. This has necessitated a flexible product-development process where designers can continue to change and

---

* Corresponding author. Tel.: +1-404-262-3823; fax: +1-404-651-3842; e-mail: atiwana@gsu.edu

[1] E-mail: bramesh@gsu.edu.

0167-9236/99/$ - see front matter © 1999 Elsevier Science B.V. All rights reserved.
PII: S0167-9236(99)00045-7

shape products even after their implementation has been initiated [35]. The impact of the aforementioned forces is witnessed most prominently in high-technology environments, where (according to a survey by National Research Council [14]), the cost of development can account for up to 85% of the total cost of the product. We posit that providing effective decision support by making knowledge about past and current development efforts readily available and accessible can make a significant contribution towards ameliorating this process. Iansiti and MacCormack [35] suggest that as firms shift from a product centric form to a knowledge centric form, support that enables continuous flow of information about stakeholder needs and evolving technologies can reduce both the costs and time required for development.

Bohn [8] observes that in dynamic environments and industries, knowledge about the process of product development is incomplete in the beginning and develops gradually over time, through various modes of learning. The process of design is characterized by complex deliberations about a series of interdependent decisions that lead to design solutions. Based on a study of concurrent product development activities, Ramesh and Sengupta [56] observe that knowledge about these deliberations is typically lost as it is never recorded. Davenport and Prusak [20] suggest that better knowledge of past, similar product development processes can lead to assessable efficiencies in product development and its consequent production. Such knowledge utilization is innately a collaborative process [1]. Here, collaboration refers to informal cooperative relationships that build a shared vision and understanding. Neither within nor cross-firm utilization and transfer of knowledge can succeed without effectively supporting collaboration [18–20].

This paper addresses the problems faced in the retention and maintenance of process knowledge that is created in new product development (NPD). We define process knowledge as tacit and explicit knowledge about activities, steps, and procedures. Our research is based on the premise that current product-development methodologies do not adequately address the capture and use of this knowledge. Since much of the formal and informal knowledge along with the context associated with it is lost after the process is completed, development teams are unable to leverage knowledge actualized by earlier teams. We address the creation and use of a repository of information and knowledge derived from sources such as recorded decisions, text documents, images, audio, and video specifically relating to collaboration within teams.

The following section defines knowledge and identifies various types of knowledge involved in the new product-development process. We then examine the role of information systems to support knowledge management (KM). We discuss NPD as a knowledge-intensive activity. Then we discuss characteristics of NPD by collaborative, cross-functional teams, and identify the problems related to KM faced by these teams. In Section 6, we map the problems in Section 5 to general solutions. Further, we translate these general solutions into specific KM system requirements. This is followed by a description of the functionalities of a prototype system designed to meet these requirements. The functionalities of the system are described using scenarios derived from a case study on the development of a personal digital assistant (PDA). Here, we demonstrate the validity our proposed solutions using scenarios drawn from our case study. We discuss related work in Section 8, and conclude with a discussion of our contributions and directions for future research.

2. Toward a Definition and Understanding of Knowledge

Davenport and Prusak [20] define knowledge as a fluid mix of framed experience, values, contextual information, and expert insight that provides a framework for evaluating and incorporating new experiences and information. They suggest that it originates and is applied only in the mind of knowers (holders of tacit knowledge) in organizations. It is embodied in documents, repositories, organizational routines, processes, practices and norms. Leonard-Barton and Sensiper [42] point to the subtle difference between knowledge and information suggesting that knowledge in the business context comprises of relevant, actionable information that is partially based on experience. We differentiate process knowledge from process information in the sense that the former is actionable.

2.1. Components of Knowledge

Nonaka and Takeuchi [52] suggest that the cornerstone of the Theory of Organizational Knowledge Creation is the substantiate distinction between tacit and explicit knowledge. Several researchers [18,20,50], define tacit knowledge as personal, context specific knowledge that is difficult to formalize, record, articulate, or encode. Leonard-Barton and Sensiper [42], extending Polyani's concept of tacit knowledge from an individual to a group level, showed that the tacit knowledge is mainly developed through a process of trial and error encountered in practice. Explicit knowledge on the other hand, can be codified and transmitted in a systematic and formal representation or language. Nonaka and Takeuchi [52] reduce knowledge creation to conversion of tacit knowledge to explicit knowledge, often achieved through *externalization.* Externalization, defined as the process through which experiential subjective tacit knowledge is converted into objective explicit knowledge, is often driven by metaphors and analogy.

Ruggles [59] suggests that KM creates value by actively leveraging know-how, experience and judgment resident within and outside an organization. We posit that KM encompasses the activities surrounding the integration of this knowledge from different sources, in different forms, and maintaining it. The key to knowledge creation thus lies in the mobilization and conversion of this tacit knowledge into a form of explicit knowledge.

Davenport and Prusak [20] indicate that some knowledge is complex and initially tacit, but it can, however, be externalized and embedded in a firm's products and processes. Nonaka and Konno [51] point out that the cognitive dimension of knowledge that shapes perception consists of beliefs, ideals, values, schemata and mental models that are deeply ingrained in participants and that these are often taken for granted by them. They further suggest that some, if not all, of this knowledge should be extracted to retain context and fullness of the explicit component.

3. The Role of Information Systems in Managing Knowledge

The development of systems to assist in managing knowledge has been a topic of considerable interest [51]. Nonaka and Konno suggest information systems can assist knowledge activists (proponents and champions of KM systems) in serving as catalysts of knowledge creation and as *connectors* of present initiatives with those in the future. Teigland et al. [69] observe that facilities to capture, reuse, maintain and transfer knowledge are essential elements of such a system. They suggest that such a channel for supporting the demand for knowledge within an organization will be very valuable [69]. Davenport et al. [18] suggest three major categories of knowledge repositories, as shown in Table 1. This paper is primarily concerned with creating internal knowledge repositories. Whereas project management tools allow for capture of formally structured knowledge, we focus on the processes underlying capture and use of informal, internal knowledge including externalization of tacit knowledge. To transfer tacit knowledge from individuals to a repository, Davenport et al. [18] suggest support for some form of community-based electronic discussion. A key fea-

Table 1
Types of knowledge repositories (based on Ref. [20])

Type of repository	Type of knowledge supported	Examples
External knowledge	Formal and informal	Competitive intelligence
Structured, internal knowledge	Formal	Techniques, methods and reports
Informal, internal knowledge	Informal	Discussion databases, lessons learned

ture that would differentiate a KM support tool from a project management tool or organizational memory store is its ability to capture and retrieve uncodified or tacit knowledge. Hansen et al. [33] refer to this strategy as *codification* as opposed to *personalization*. Tacit knowledge, like explicit knowledge, can also become outdated [42], hence invalid. Therefore, it is critical for a KM system to ensure the applicability of tacit knowledge to the current situation [42].

4. New Product Development as a Knowledge-intensive Activity

Eder [24] suggests that much of the knowledge in NPD, such as knowledge about the strategic design approach, and knowledge about tactics and methods for designing is primarily tacit. Several researchers have described NPD as a knowledge-intensive activity [20,35,52,64]. NPD often involves cross-functional linkages, where different participants join a team with differing viewpoints. Such teams are often characterized by participants who achieve a high level of at-stakeness and synergy resulting from their interaction with other team members [36]. Morrison and Kennedy [47] suggest that this interaction brings in the need to organize, integrate, filter, condense and annotate [46] collaborative data and other relevant information that these team members contribute. Court [15] identifies three categories of knowledge that product designers use in the process of developing a new product.

(1) *General knowledge*: Knowledge that people gain through everyday experiences and apply without regard to any specific domain that they might be working in.

(2) *Domain specific knowledge*: Knowledge gained through study and experience within a specific domain. This is generally improved as the person(s) involved in projects gain more experience.

(3) *Procedural knowledge*: Procedural knowledge, the primary focus of this paper, is gained from experience of undertaking a task within the domain. Court [15] suggests that this is a combination of the above two types of knowledge. In the NPD context, this includes knowledge about the processes through which tasks are accomplished.

4.1. Codification vs. personalization centric knowledge management

Hansen et al. [33] have classified KM strategies into two distinct categories: codification and personalization. The codification strategy is defined as being more reliant on computers and networks. They suggest that the value of codification lies in its ability to create economics of reuse. The goal of such an approach, they suggest [33], is that of connecting people with reusable codified knowledge. Personalization, on the other hand, relies more on social networks that allow knowledge workers to share tacit knowledge. They suggest that personalization creates value by connecting people with relevant knowledge. The role of technology in such KM implementations is limited to that of enhancing social communications networks.

Previous research in product development (such as Refs. [35,64]) suggests that product development often builds upon an existing base product and rarely begins from scratch. Iansiti and MacCormack [35] discuss the case of Netscape Corporation's browsers that build on previous versions. Similarly, Song and Montoya-Weiss [64] provide examples of incremental products in several other industries. Further, personalization and codification strategies are not mutually exclusive, but can exist in combination even though the primary focus should be on *one* of these [33]. We, in concordance with Hansen et al.'s suggestion [33], therefore, posit that technology support for NPD must incorporate *both* these strategies; however, the primary focus (as also suggested in Ref. [33]) should be one of these two.

Procedural knowledge is experiential and includes explicit aspects such as fundamental design concepts, criteria, specifications, theoretical tools, practical considerations, and design instrumentalities [15], as well as tacit aspects which Davenport and Prusak [20] suggest ''reside in the minds of people'' in an unarticulated format. Past design knowledge that can help product designers include both prescriptive knowledge such as knowledge about the general strategic approach to design, knowledge about techniques, and methods for designing [24], or descriptive knowledge such as knowledge about design processes and knowledge about working means [24]. Since a substantial portion of this knowledge is tacit, we focus on the development of KM tools to support the creation of internal, informal knowledge repositories containing such knowledge.

5. Knowledge Management in Collaborative new product Development

Collaboration is the centerpiece of product development processes. It is essential to distinguish between collaboration and interaction for the purpose of distinguishing knowledge involved in the two processes. While interaction refers to formal, transactional communication links, collaboration refers to informal, cooperative relationships that build a shared vision and understanding needed for conceptualizing cross-functional linkages in NPD contexts [36]. Collaboration is therefore imperative in knowledge generation and transfer [59].

In a well-managed development process, a cacophony of perspectives foster creative abrasion, which Leonard-Barton and Sensiper [42] define as an intellectual conflict, between diverse viewpoints that produces energy channeled into new ideas and products. We observe that to enable a high degree of cross-functional collaboration supporting the development of a shared vision and understanding is crucial. Jassawalla and Sashittal [36] state that in comparison a firm with low levels of collaboration in product-development teams, a firm with high levels of such collaboration has an equitable distribution of power among participants. This results in intrinsically motivated participants and synergistic outcomes. According to Ruggles [59], managing knowledge in collaborative teams allows cross-fertilization among sources of internal expertise and creates networks of knowledge workers within and outside the organization.

5.1. Need for Managing Knowledge in Product Development

Davenport and Prussak [20] suggest that innovation and speed to market that are essential for business success will become increasingly critical in the future. According to Quinn et al. [54], the intangibles that add most value to these activities are knowledge centric. Most product development is moving towards team-based structures, since teams are believed to increase individual commitment and performance, and as Galegher et al. [29] observe, are more effective in bringing a new product to the market in a short time-frame. As products and technologies become increasingly complex, NPD requires effective collaboration and synergistically integrated skills of several individuals.

5.2. Research Methodology

We have used the case study method for data collection and subsequent validation. Orlikowski and Baroudi [53] have suggested that this method is best suited to understanding the interactions between information technology related innovations and organizational contexts. Yin [75] describes this technique as ‘‘an empirical inquiry that investigates a contemporary phenomenon within its real life context, especially when the boundaries between phenomenon and context are not clearly evident and it relies on multiple sources of evidence’’. Yin suggests that a single case study is appropriate where it represents a critical case or meets all criteria for testing a theory, or where it is a revelatory case. A single case allows us to investigate the phenomenon in depth to provide rich description and understanding as suggested by previous case study methodological literature [17,53,75]. Darke et al. [17] caution that statistical generalization is not the goal of case studies; deep insight into dynamics of processes and situations, however, is [17,53]. For these reasons, we believe that our use of a case study is appropriate as it provides deep insights into knowledge related problems, albeit limited in generalizablity to a specific type of organization.

The problems associated with NPD are more pronounced in high-technology industries such as the electronics industry. Firms in such industries face considerably high rates of product obsolesce because of rapid advances in technology coupled with intensive competitive pressures [73]. Further, Von Glinow and Mohrman [73] note that they also invest proportionately larger sums into research and development while relying on rapid, efficient new product introductions to remain competitive. Teece [68] highlights sectoral differences between process intensive high technology industries (such as electronics) and traditional process intensive industries (such as chemicals). As shown in Table 2, high-technology products often involve a high degree of design complexity, contextual dependency, limited intellectual rights protection and short product life cycles. Further, the development of high-technology products also involves complex collaborative processes where KM is critical. For these reasons, our study has focused on NPD in the electronics industry.

5.3. Developing a Personal Digital Assistant: A Case Study

In this paper, we discuss a typical NPD process in the consumer electronics industry, viz., the PDA, to illustrate the need for KM. Our observations are based on a case study of one of the largest manufacturers in the industry and on existing literature [12,38]. Our study of this NPD effort involved a series of discussions with members of NPD teams involved in the development of a PDA. The data from this case study is used in understanding the characteristics of the NPD process, the problems faced by NPD teams (specifically, related to management of process knowledge), and the requirements of a KM system that can alleviate these problems, and finally the development and validation of a prototype implementation of a system to support KM through the identification of typical scenarios of NPD activities that can be supported by such a system. The use of such an approach to gain deep insight into the dynamics present in single settings is particularly useful when research and theory are at their early formative stages [17], as is the case with KM.

5.4. Characteristics of New Product Development

A review of recent research suggests that collaborative NPD processes have several key characteristics that result in a variety of KM problems. In this section we discuss the characteristics with examples drawn from our PDA development case study.

- *Short product and process life cycles*: Bettis and Hitt [6] observed that product life cycles in certain markets have significantly shortened thereby compressing the available time window for recouping the expenses associated with product development.

Time-to-market is considered a critical factor in the development of a PDA. Iansiti and Macormack [35] have demonstrated this for Internet products. The industry has experienced the introduction of nearly twenty competing products, three real time operating systems (RTOs), convergence of functionality of hand-held devices, palm devices, small phones, and car communication systems within a short time span of about 2 years. Frequent changes in the RTOs, communication protocols supported, communication and computing hardware and software are common in this market [12].

- *Cross-functional collaboration*: In order to respond to competitive challenges, organizational units have become more closely coupled than in the past, often working in parallel to complete assignments spanning traditional units [29] and functional areas. Leonard-Barton and Sensiper [42] suggest that creation of today's complex systems and products requires merging of knowledge from diverse disciplinary and personal skills-based perspectives where creative cooperation is crucial for innovation.

In the development of a consumer electronics product, it is necessary to draw necessary expertise from a variety of functional areas including technical, manufacturing and marketing. Specifically, the development of a PDA requires technical expertise from both hardware and software design and development. In the hardware category alone, experts in the following areas are expected to work very closely:

Table 2
Sectoral differences in knowledge for process intensive vs. high technology process intensive product development (based on Ref. [68])

Challenge	Chemical industry	Electronics industry
Recognition	Manageable	Extremely complex; often impossible
Disclosure	Patents common	More difficult
Interface issues	Not critical	Compatibility with other components is generally critical
Dependency of value on context	Strong	Very strong
Patent protection/replicability	High	Sometimes very limited
Development cycle	Often long	Generally extremely short

design of radio frequency (RF) devices, digital signal processing (DSP), and printed circuit-board design. Software development in communication protocols, DSP, and user interfaces need to be closely coordinated. The development process is characterized by intricate interdependencies among all these areas, as well as with others such as manufacturing, marketing, and packaging. Further, knowledge from other independent groups that have experience with partially similar subcomponent technologies, such as voice paging is also needed.

- *Cross-institutional collaboration:* Besides spanning multiple functional areas within an organization, development of complex products also requires bringing together participants from across multiple collaborating organizations [1]. Expertise and skills might be distributed both within and outside the developing organization [35]. Davenport and Prusak [20] suggest that this brings in the need to facilitate knowledge growth, knowledge sharing and dissemination For example, in developing a PDA, it is common to employ outside expertise in specialized areas such as DSP. Other examples of cross-institutional knowledge required include knowledge about the protocols used by the organizations that provide the communications services for the product, knowledge about the operating system itself (such as Windows CE), knowledge about the dimensions and features of competing products, etc. Close collaboration among these cross-institutional teams, we believe, is essential for product success.

- *Transient existence of teams and high turnover*: In large projects, membership in development teams changes over time and across phases. A major threat to the collective knowledge in organizations is personnel turnover, since much of this knowledge is situated in the minds of individuals [66]. Carley [10] observes that when there is no repository for knowledge other than personnel, turnover leads to reduction in the organizational knowledge. Similarly, March observes growth of organizational code under conditions of low turnover and high socialization [44].

Rapid market growth and the specialized skills necessary are critical factors contributing to the severe shortage of qualified personnel and high turnover among PDA development teams. Further, members of ad-hoc teams return to their organizational units as their efforts are completed, causing loss of synergy developed over the course of the project.

5.5. Problems In the New Product Development process

The above characteristics of NPD lead to a variety of problems that suggest the need for better KM. As before, examples from PDA development are used to illustrate these problems.

- *Lack of shared understanding*: Uncertainties in NPD processes lead to dependencies among and between different functional areas and require cooperation to accomplish individual and joint objectives [64]. Szulanski [67] conjectures that the consequences of this problem are the lack of absorptive capacity of the recipient, and the inability to contextually understand ''best practices'' in development.

This problem is very pronounced in the development of a PDA. The new PDA design calls for the unification of features typically found in a personal computing device and in a communication device — products traditionally designed and manufactured independently. In addition to the traditional functional barriers that exist between marketing, design, purchasing, and manufacturing that can be observed in most industrial organizations, the diversity of the expertise needed to for NPD creates serious barriers for shared understanding. As mentioned earlier, even the design team consists of experts from a wide variety of hardware and software disciplines. This problem is exasperated as the degrees of freedom for each of the teams is severely restricted due to constraints on total cost of the product, its size, its weight, appearance, and time-to-market. The team members drawn from different disciplines lack understanding of the critical design factors for other areas. For example, the designers of the computing platform need to be aware of the signal interference problems that various components can cause for the RF components.

- *Overreliance on transmitting explicit rather than tacit design information* [20,21,25,52]. Nonaka and Takeuchi [52] have pointed out the importance and value of recognizing and capturing tacit information — such as know-how, judgment and intuition, which make up a critical component of information that needs to flow between members collaborating within a team. This highlights the need for a method to effectively transfer such knowledge (actionable information) in addition to explicit knowledge.

During the design and development of a PDA, participants from the manufacturing department might not realize that on-board surface mount component (SMC) chips cannot tolerate over a certain soldering temperature during the process of manufacturing. Though the hardware design group is well aware of this through their chipset specification literature, this knowledge was not shared with manufacturing.

- *Repeated mistakes*: Organizations have been frustrated by reinventing solutions and repeating mistakes due to their inability to identify or transfer lessons learned from failures from one location to another or one function to another. Transfer of knowledge from failed projects to new ones could substantially reduce the expenditure of resources and effort. Teece [68] suggests that innovations in product development involve a considerable degree of uncertainty and that knowledge about failed approaches is frequently forgotten, resulting in their repetition.

The designers of PDAs face a lot of uncertainties due to the lack of de-facto ‘standards’ in the operating systems and communication networks that they can chose from. To spread the risks, parallel development for different platforms needs to be undertaken. For example, several PDA manufacturers make products for both the Windows CE and Palm OS platforms. Many of the problems encountered by the software designers are common across these developments. However, as the two groups come from different backgrounds and use different software development tools and platforms, the common mistakes made by these teams are not readily shared. Similarly, the development of the communication components for the PDA shares similarities with the components in voice paging products. Past knowledge from the development of pagers such as design problems and market reactions can be highly beneficial to the PDA development effort. However, as the development is done by different teams across interdepartmental boundaries, such knowledge is not readily accessible.

- *Reinvention of solutions during product evolution*: Another problem in product development teams is that they expend resources into solving problems that might have already been solved either within or outside their collaborative group. Based on empirical observations in software development, Ramesh and Sengupta [56] conclude that work groups often repeatedly discuss the same issues that had been resolved earlier, as there may exist no reliable record of these discussions.

Teece [68] indicates that the annual aggregate ‘reinvention’ costs in the United States range between US$2 billion and US$100 billion. Court [15] offers support for the suggestion that product designers often tend to use the incomplete information they already possess, rather than seek expertise that does exist within the enterprise or is external to it.

In the organization where our case study was done, the design team involved in the development of a successful product has often been moved to the next high-profile project when the previous project was near completion. The expertise gained during development of a product is not readily available to design teams working the subsequent versions of the same product during its evolution. For example, a design team made critical design decisions about the location of several RF components based on the communication protocols that needed to be supported. However, as the product evolved, a new team assigned to the project made changes to the communication protocols without realizing their adverse effect on RF components.

- *Skills developed due to collaboration may be lost thereafter*: Quinn et al. [54] suggest that professional know-how is developed most rapidly through repeated exposure to the complexity of real problems. In a project-oriented team-based organizational structure, skills developed during the collaboration process might be lost after the team is broken up and redistributed among other teams or groups working on newer development projects [56]. When a team is disbanded, the process knowledge acquired by the team is lost and is not available for tasks such as product modification or maintenance [32].

Ad-hoc teams formed for NPD are often disbanded at the end of the development. Team members often get assigned to other projects wherein their functional expertise is more valued than the knowledge gained during the process of their collaboration with other functional and technical areas. For example, the domain knowledge gained by a RF engineer during the development of PDA may be 'wasted' if he is transferred to a project on voice pagers to utilize his functional skills.

- *Inability to transfer existing knowledge into other parts of the organization* [59]: Many organizations face difficulties in transferring knowledge from one organizational unit to another. Galegher et al. [29] highlight the problem in the diffusion phase wherein team members begin transferring technical data as well as a sense of ownership to other groups that must manufacture and market the new product. Maintaining motivation for knowledge transfer at this stage is challenging as all major product development decisions have already been made and what remains is the completion of product details. Continuing with the example of the PDA, innovative design solutions arrived at by the voice paging development for power management (conserving battery life when not in transmitting mode) is not shared with the PDA team. Similarly, the team working on the Palm OS version of the product is very experienced in developing hardware solutions for efficient power management, but does not share the core technology with the Windows CE team.
- *Inconsistency in multiple versions of information*: Recent research (such as Refs. [3,4,9,15]) suggests that an enabling condition for knowledge creation is redundancy. Redundancy offers an overlap in knowledge between different groups that promotes cross-functional collaboration. The need for redundancy, however, must be met simultaneously with the need for maintaining consistency across different versions of information that may be possessed by different team members.

As problems are encountered in the design of communication components during testing with protocol encoders, the hardware designers make 'minor' modifications to their designs. However, the hardware–software interface specifications agreed to between the hardware and software designers are not updated. The two teams, working with different versions of the specifications, run into serious incompatibilities during integration testing. The problem of integrating multiple sources of knowledge in a coherent manner that can be brought to bear upon design is illustrated by Robillard [58] in the case of software products, such as the PDA operating system.

- *Evolving assumptions*: Design decisions made in the process of developing a new product, might be based on some critical assumptions [8], both technical and nontechnical. Due to the dynamic nature of product development activities these assumptions often change [55], necessitating reevaluation of the decisions that depend on them.

The design team makes critical decisions about the processing power and battery-consumption requirements of the PDA. These are based on assumptions about the power-management functionalities provided by the operating system. A recent upgrade to the operating system significantly changes the power management services. However, the change in this critical assumption is not communicated to the various design teams, resulting in duplication of efforts to conserve power during both hardware and software design.

- *Loss of tacit knowledge* [59]: Tacit knowledge is difficult to articulate in a way that is meaningful and complete, hence it is often lost [68]. Teece [68] suggests that the larger the extent to which a unit of knowledge has been codified, the lower are its transfer costs. Uncodified or tacit knowledge is not only slow to transfer, but also leads to ambiguity [68].

The designers of the RF components assume that the designers of the computing processors are aware of the interference that the processor may cause with the communication components. This knowledge though obvious to the RF designers, is not commonly well understood by other team members. In the absence of explicit requirements for the RF team, the processor designers ignore the issue.

6. Requirements for a Knowledge Management System

The focus of our research is the development of a KM system to support collaborative NPD. As a first step, we identify the requirements for such a system by examining the various KM problems faced by NPD teams. We then identify general solutions to these problems suggested in the literature. Finally, we identify specific requirements for a KM system based on these general solution strategies. We present the above analysis in Table 3. Here, specific system requirements have been identified by func-

Table 3
Mapping problems to system requirements to support process KM

Problem faced by NDP teams	General solution	System requirements
Over reliance on transmitting explicit rather than tacit design knowledge [18,33,42,51]	Convert a part of tacit design knowledge into explicit, articulated and stated knowledge by asking participants to record assumptions and beliefs along with decisional steps [25,26,50]	Multimedia integration {MM} that forces users to think through the process by articulating dependencies on cross functionally significant aspects of the design [4,5,30]. Integration with underlying assumptions {IA} [56]. This would allow for the creation of a strategic agreement between the various participants. Multimedia support to capture knowledge that can not be explicitly codified or written down {IK}. This would facilitate more superlative knowledge transfer and exchange [49,56]. Support for recording assumptions {RA} and recording beliefs {RB} [55].
Barriers due to lack of absorptive capacity of the recipient [65,70]	Retain context along with information stored	The context of each decision [18,20] in the process can be captured with each concept {DC}. Multimedia support makes such context recordable even when it can not be codified. Some researchers have identified this type of knowledge as implicit knowledgethat can be captured but not formalized or codified
Changing team membership [18,20,31] repeated mistakes [68]; reinvention of solutions [68]	Divorce knowledge from the holder, convert to explicit knowledge	Each team member's augmentation to the design discussion process is captured in deliberation records {DR}. Change in team membership does not result in knowledge 'walkouts' as a significant portion of the team member's contribution to the team's synergy (see, for example, Refs. [28,30,37] and integrated understanding is captured in the system.
	Capture past design experiences in a manner useful for later reference during design processes	Design information from past projects {DP} and current projects {DN} is accessible to the present team. Such information is available both for past and ongoing projects throughout the enterprise by means of a distributed workspace enabled by a communications network {NE}. Design information from both the past projects as well as the current project is retrievable online, in real time, across a distributed workspace [62] (such as a client server implementation). This helps prevent repetition of old mistakes.

	Create well-indexed knowledge of similar problems faced in earlier groups and teams.	Design knowledge — both formal and a part of the informal — from past projects {DP} {DN} is readily available and captured within the system. Ad-hoc retrieval of informal information is supported using meta tags.
Shared medium	Shared medium between cross-cultural team members provides a common discussion field.	Shared medium {SM}; consistently interpretable (across functional and national boundaries) forms of representation using icons {GI}; retention of credit to the original contributor (for contribution credit and reward matching) are supported by the system {AT}
Loss of collaborative skills [54,60,61]	Support capture and reuse of knowledge created during the collaborative process itself.	Collaborative design dialog {CD} throughout the design process is captured as a process. Such a process can be replayed or reenacted to reveal the sequential and parallel activities and contexts of past design decisions {RE}.
Versioning of information [59,60]	Store multiple and identifiable versions of content at a single central remotely accessible repository.	Versioning of process knowledge is supported by the system {VC} [51,52,76].
Process knowledge might be lost after the project is completed [32]	Retain dialog between the members of the design team as a part of captured knowledge [26]	Preserve antecedent dialog between the members of the design team as a part of captured knowledge {DR}.
High development costs [14]	Reuse knowledge, processes and design artifacts from past projects.	Decisions from both the current and past projects are available {DP} {DN} {NE}. This can potentially reduce the time spent in reinventing solutions and the costs incurred in the process [20,23].
Unstated assumptions [55,56]	Accommodate assumptions made in the design process and dynamically link them to decisions made throughout the process. Effects of changing assumptions must be supported by a system supporting knowledge capture and codification, etc.	Each decision made in the design process can based on multiple assumptions which are linked {LA} to it [25]. The effect of a change in any assumption automatically ripples through the rest of the design decisions and by processes to show the effect of changing assumptions or levels of belief on the rest of the design process {TA}.

tionality codes (within {braces}). Mapping of these requirements with the functionalities of a prototype system will be presented in Section 7.

7. Prototype System to Support Process Knowledge Management

We have developed a prototype system to support KM tasks in collaborative NPD. Functionalities of this system are grounded in the 'requirements' identified in Table 4. Based on data collected from our case study, we propose and validate using a prototype, system characteristics that can help overcome KM problems faced by NPD teams. The discussion in this section is organized around the critical functionalities provided by the system. Table 3 maps these functionalities to the requirements identified in Table 4.

We describe our prototype system using scenarios of usage of the system in the design of a PDA. The scenarios were identified and refined by our case study. They were validated by participants in our study as representative of the situations encountered by them in their tasks. As the focus here is on illustrating the functionalities of the system, only small snapshots from the complex product development activities for a PDA are presented in the following discussion.

7.1. Implementation

The prototype system is based on ConceptBase, an implementation of the high-level conceptual modeling language *Telos* [48]. Telos is based on a first order assertion language and provides facilities for specifying meta-concepts, semantic integrity constraints temporal knowledge and deductive rules. Using meta-concepts, meta models that represent various classes of knowledge can be specified and instantiated. The prototype is based on a client server architecture and supports group work. Users of the system spread across a network can communicate with each other through a centralized knowledge server. This server maintains the integrity of the process knowledge components. The system provides a graphical user interface for communication with the server to retrieve and modify the contents of the knowledge base. Further, the browser provides links to external tools such as a WWW gateway.

7.2. Definition of Concept Maps

As a first step in supporting the capture and use of process knowledge for NPD involves the identification of the critical components of knowledge. Our prototype system provides the ability to define meta models in terms of objects representing knowledge components of interest. Further, associations among these components can also be represented. Finally, characteristics or attributes of concepts can also be easily specified. Once a meta model (or a schema) is defined, users of the system can instantiate these models.

Based on discussions with members of the NPD in our case study, the meta model shown in Fig. 1 was selected as a candidate scheme. In this model,

Table 4
System functionality and resulting requirements that are specified by such functionality

System functionality	Requirement(s) based on Table 3
Definition of concept maps	{GI}, {VC}, {DB}
Support for knowledge capture	{DR}, {NE}, {SM}, {NE}, {AT}, {CD}
Representation of context	{DC}, {NK}
· Links to sources	{MM}
· Informal and formal components	{MM}
Knowledge access	{NK}
Assumption surfacing	{IA}, {LA}
Review of past knowledge	{DP}, {DN}, {RE}
Agents for dependency management	{LA}, {TA}

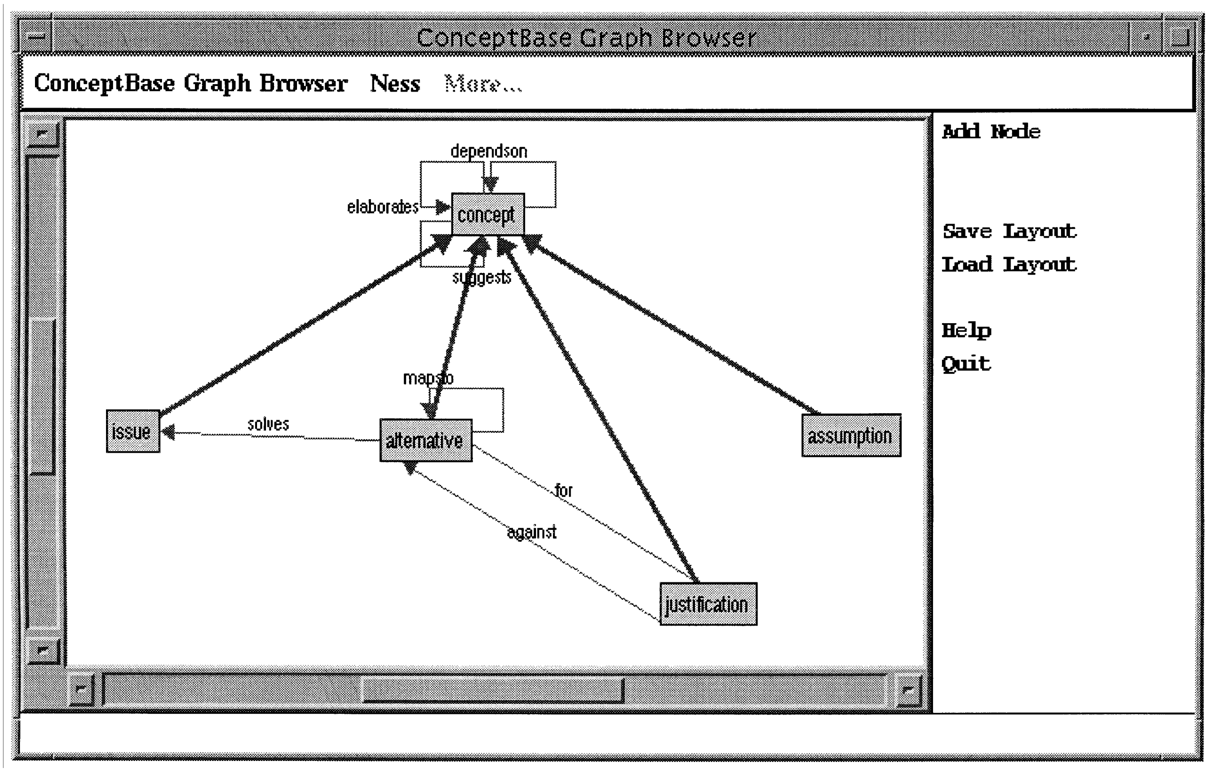

Fig. 1. Concepts representing various knowledge components.

concepts are knowledge components that can be used to represent the participants views of interests, concerns and tasks [40]. A concept may suggest other concepts, elaborate on others, and even depend on others. A few specializations of concepts were also identified. These include issues, alternatives, justifications and assumptions. Issues are questions or concerns that need to be resolved to arrive at decisions in NPD, and alternatives are various answers or solutions to these issues. Justifications that are for or against these alternatives are also included in the model. Finally, assumptions underlying concepts are also represented. This model is similar to the issue-based information systems (IBIS) model of argumentation [13] that has been used successfully in a wide range of domains to represent complex problem solving processes. It should be noted that the choice of the specific meta model is entirely up to the users of the tool. The system supports the definition and instantiation of any model chosen by the users. Using parent–child relationships (or IS-A hierarchies) and treating attributes as first-class objects (so that they can have attributes of their own) complex models can be easily specified. Our system provides a graphical editor/browser to define and navigate through knowledge components. Context sensitive menus are provided for the users to define instantiate, modify, and link objects in the meta model.

7.3. Support for Knowledge Capture

After the meta model in Fig. 1 is defined, users can create and modify process knowledge components and relationships among them using a using the graphical browser/editor. Each user may invoke the client GUI and connect to the same knowledge base maintained by the server. Thus, multiple users connected to the same server may conduct 'conversations' in terms of the primitives specified in the meta model. In these structured conversations each team member can add and modify various concepts and relationships among them. They may seek clarifications of concepts proposed by others. Using this facility, the team members can communicate their viewpoints and expertise [27] and map their views of the problem with those of others. In our example, the users may propose, suggest, elaborate on various issues, alternatives, justifications and assumptions. These knowledge components may be viewed (and modified, if permitted) by other members of the team. They may, in turn, respond by proposing other concepts and relationships. With such a conversation, various viewpoints are exchanged among the members of the NPD team.

Fig. 2 represents the output of such a conversation derived from our case study, to represent a typical scenario in PDA design. Here, three stakeholders (the product manager, the marketing specialist and an RF engineer) contribute to the definition of the various concepts. First, the product manager proposes a discussion between marketing and RF engineer(s) on the factors that are critical in the development of a PDA. He posts/raises the issue ''What are critical factors'' in the system requesting responses

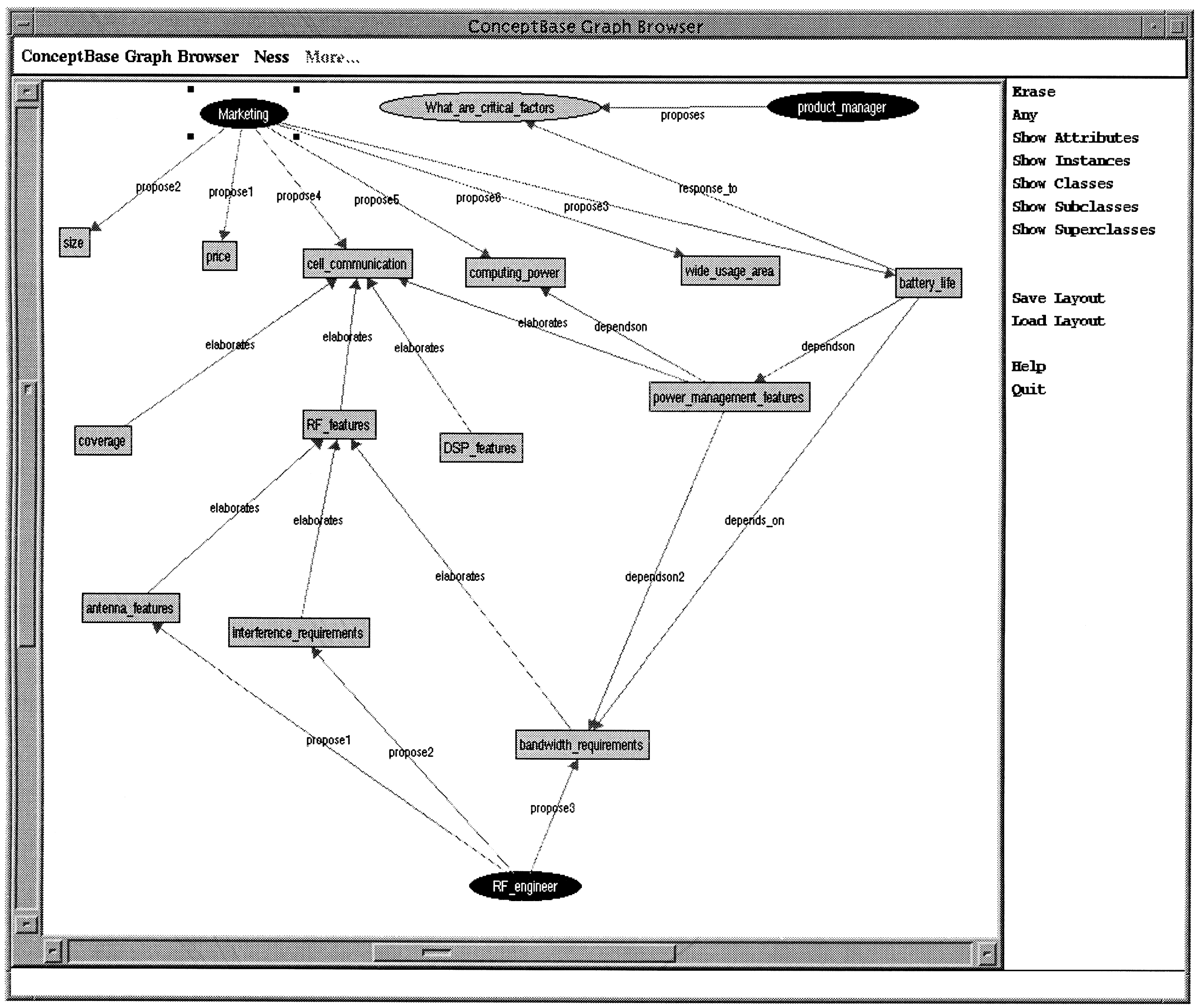

Fig. 2. Conversation among group members.

from the two groups. Marketing proposes that size, price, cellular communication capabilities, computing power, wide usage area, and battery life are all important factors to the market segment that the PDA is designed for. Many of these factors are stated in a language that may not be readily understood by the design team that needs to implement appropriate design features. To foster better understanding, the product manager seeks clarifications from marketing on the requirements for cellular communication. Marketing elaborates with details on the coverage desired, RF features desired, DSP features, power management features, etc.

At this time, the product manager asks the RF engineer to state how the RF features proposed by marketing can be implemented. The RF engineer elaborates on the specifications for the antenna, interference requirements, and bandwidth requirements that are to be addressed. He also wants to highlight the fact that many of these requirements have strong dependencies on others. For example, the bandwidth requirements will have a strong influence on battery usage and therefore is of critical importance while developing power management features. This is further related to the computing power that is required by the user. These are identified using the ‘depends on’ links between these components in Fig. 1. It should be noted that each node in the diagram could be linked to an external document, a Web page or an artifact to provide further details (as discussed in detail in Section 7.4).

The discussions such as the above help the team clearly state their viewpoints, understand the viewpoints of others, map their views to those of others such that the team develops a shared understanding of the problem being solved. Even when the discussions are conducted in the context of structured processes like the development of a house of quality, our system can be used to capture the process knowledge behind these processes.

7.4. Representation of Context

The larger context in which a participant has developed a particular perspective can be better understood by others if they have access to the details such as work products, supporting documents, etc. These ‘sources’ may also be available in various levels of formality (ranging from hypermedia documents to formal definitions). Quinn et al. [54] refer to this as an elevation from *know how* to *know why* (knowing why a given choice was made over another), which he argues, makes a team more flexible and innovative when faced with a previously unseen problem. Using our system, a user can specify an exhaustive variety of information about concepts that they specify; These knowledge chunks include:

- *What* information is represented — including salient attributes or characteristics.
- *How* this information is represented both by formal and informal means and how it relates to other components of knowledge.
- *Who* are the stakeholders that played different roles in its creation, maintenance, validation and use.
- *When* this information was captured, modified and evolved.
- *Where* it is represented—in terms of sources that ‘contain’ this information.
- *Why* a certain concept evolved, or was created.

7.4.1. Links to sources

Our system provides the ability to link most of the above information as attributes of any concept. Also, a user can link a concept to the sources that provide additional information. For example, the RF engineer, by viewing the video clip of a focus group session in which the battery life of different competing products and the reactions of the consumers are discussed, may better understand the requirements proposed by marketing. This clip is attached to the specifications from marketing as an important rationale.

Each object in the knowledge base can be linked to static documents or to documents dynamically created by searching the repository of hypermedia documents on the WWW. Further, a context sensitive menu provides the facility to invoke external tools (such as a WWW gateway) to retrieve documents

- that have been explicitly linked to an object,
- that are indexed with the keywords defined as attributes of the object, and
- that are considered ‘similar’ (using a variety of search techniques).

In our example, the RF engineer is interested in reviewing details of FCC regulations on RF communication while finalizing the RF features. He may readily retrieve all the documentation on the relevant government regulations that is located in the departmental or corporate Intranets or the WWW. He invokes an indexing gateway provided by our system that is available as a menu option on the concept RF features. His search for related documents may be guided by the following inputs:

- Keyword(s)
- Boolean search operators (and, or, etc.)
- Date relevancy
- Thesauri to be used
- Substring options (word stemming)
- Document hierarchy

It should be noted that the indexing gateway provided with the system can be substituted with any other search procedure or external tool that can be invoked with a system command. Thus, our system provides interfaces to knowledge chunks not captured within the system.

Finally, recent research [52] recognizes the importance of identifying the people involved in the creation and maintenance of organizational knowledge, when explicit representation of such knowledge is not feasible. In our system, such information is maintained as attributes of concepts.

7.4.2. Informal and formal components

Informal information can be maintained using hypermedia documents. Our system supports formal definitions of hypermedia objects and linking them to the relevant concepts. Several attributes can be defined for each hypermedia object that can help in classifying, indexing and retrieving them. Some examples of such attributes are: its creator, maintainer, the other objects it is linked to, and the project(s) that it is a part of.

In our case study, the marketing department is interested in specifying a variety of information about concepts they proposed. For example, they may provide details on the concept Battery Life (BL) with the following information:

- *What*: The duration for which the PDA functions without requiring a recharge. It does not refer to the physical life of the battery.
- *How*: A chart comparing the battery life of various competing products.
- *Who*: Created by marketing team
- *When*: Automatic time stamp by the system at the time of creation
- *Where*: Reference to a marketing department memo discussing implications of battery life on customer satisfaction
- *Why*: A video clip containing a segment from a focus group discussing the importance of battery life. This can be especially useful where tacit or informalizable components of knowledge such as customer expressions and gestures are involved.

Our case study, in agreement with existing literature [17,53,75], suggests that such a rich definition of concepts is considered invaluable by other participants in the NPD team.

7.5. Knowledge Access

Our system uses the deductive query language provided by ConceptBase to define various types of ad-hoc queries that could be used to retrieve information of interest to different participants. Queries are defined as special classes whose instances are answers to the query. A graphical interface is provided for displaying queries and retrieving desired answers. Recursive queries are very helpful in selectively retrieving contents of the knowledge base.

In our case study, a representative query would retrieve all documents that are related to a specific design issue. The answer is constructed by first identifying all alternatives that address the issue, their attendant justifications and assumptions and then retrieving all documents that are linked to any of these. Our case study suggests that a variety of such ad-hoc queries can be tailored to the specific needs of the various team members.

7.6. Assumption Surfacing

The need to explicitly state the assumptions behind concepts was repeatedly highlighted in our case study. As described in our meta-model, assumptions and their relationships to other assumptions and concepts can be captured using our system.

Fig. 3 provides a snapshot of a discussion during the development of a PDA that illustrates the importance of assumption surfacing. In this figure, the output of a design deliberation about a critical issue, viz., the protocol to be used for cellular network communication, is shown. Initially, the various team members propose two alternatives. They are time division multiple-access (TDMA) and code division multiple-access (CDMA). A designer proposes the use of CDMA. His recommendation is based on the assumption that the system will be marketed only in the US where this technology is used (or that coverage in Europe is unnecessary). Marketing however suggests that upgradability is an important consideration in the target market. That is, a PDA user must be able to upgrade to any other cellular network that may eventually replace CDMA as the primary option even within the U.S. Given the high possibility of a third generation cellular standard based on global system for mobile communication (GSM) emerging within a few years, another designer suggests support for the GSM protocol. This option is objected by the need for backwards compatibility with CDMA based systems in the U.S. More importantly, to make this critical decision, it is important for the designers to understand the implicit assumptions behind the consideration proposed by marketing. Marketing is asked to clarify the total life span of the PDA during which it should be upgradable. If it is a short duration, then the option to support GSM is irrelevant, as the third

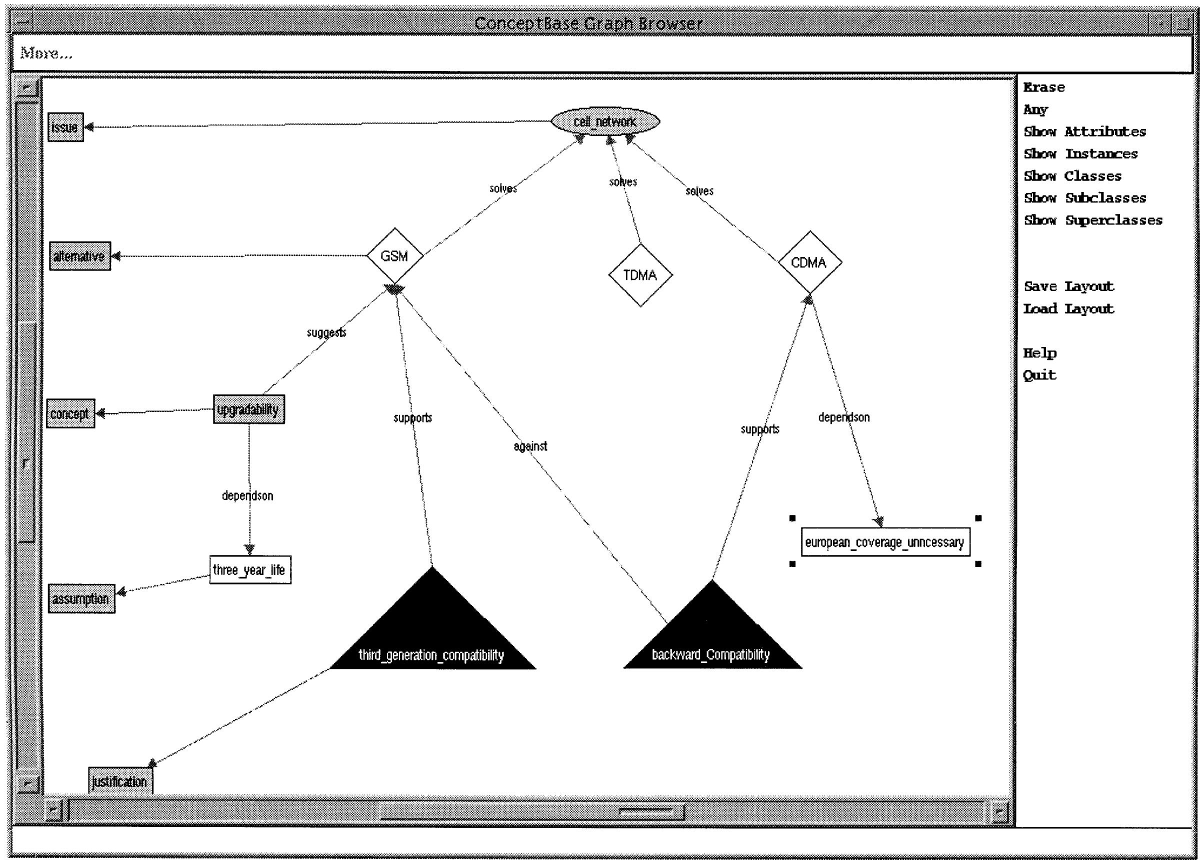

Fig. 3. Managing dependencies and assumptions as demonstrated by our PDA development case study. The figure illustrates how dependencies between assumptions brought into the design decision by functionally diverse participants influences the final design decision at each decision node.

generation standard will not be ready. However, marketing now states that the expected life of PDA is about 3 years, requiring serious consideration of GSM. During such a conversation, critical assumptions behind the various concepts such as issues, alternatives, justifications, etc. are made explicit, thereby helping the NPD team make better-informed decisions.

7.7. Review of Experiential Knowledge

An important feature that was highlighted in our case study is the ability to retrace the various steps that were taken in the NPD process. This would help take corrective action when past mistakes are revealed. Further, such a review of knowledge is also useful to facilitate understanding of decisions, as well as for identifying the choice points where alternative decisions could lead to different solution paths. The team would benefit from a review to understand how the knowledge components were defined chronologically. Finally, the ability to selectively review the history focusing on select aspects of the problem will also be very useful.

The functionalities of our system to support such a feature is based on the premise that a NPD team may be interested in revisiting a decision process, including the *dead ends*, in the same (time) order in which it happened. Such a review is used in explaining the process and outcomes to new participants or as a training mechanism. In our system, for each assertion in the knowledge base a time stamp is created. This information can be used to trace the steps in the evolution of a concept (say, an issue). Such a replay could also facilitate reevaluation of assumptions and decisions, leading to their revisions. In our case study, when a new design engineer wondered about the final decision to go with a CDMA cellular network instead of a newer technology, the system can present the various steps in the decision process, in the order in which they occurred.

Very often the users may be interested in focusing their search through history to study the context in which a specification or a decision was arrived at [2]. Such 'reasoning backwards from outputs' is important in complex decision making situations and is more useful than the traditional 'what-if' analysis involving 'reasoning forward from data'. In the former, the decision-maker starts with the goal and assesses whether 'satisfying' the goal can endure changes in the inputs, whereas in the later case, the sensitivity of the results to changes in the inputs are assessed. In our case study, a design engineer may just be interested in all the assumptions leading to selection of a particular cellular standard rather than a comprehensive history of the various alternatives considered their justifications and their validity, etc. This can be done by selecting a menu choice from any concept to retrieve all (direct or indirect) assumptions behind it. By selecting this option from the node GSM, the user will retrieve the assumption that the PDA's expected life is 3 years.

7.8. Agents for Dependency Management

The meta model derived from our case study provides ability to explicitly represent dependencies among various concepts such as assumptions. The case study also suggests that this information can be made more valuable if mechanisms for managing such dependencies are available. Our prototype system employs autonomous agents for maintaining dependencies at different levels of automation. For example, when a concept depends on another, the agents maintain the semantics of this dependency by propagating relevant properties of one concept to another. If a concept depends on an assumption, then the belief in that concept is based on the belief in the relevant assumption. Using deductive rules, the autonomous agents propagate these beliefs.

A major concern for the NPD team is that often the repercussions of changes in critical assumptions are not well understood, leading to very costly mistakes and rework. For example, the change in the validity of an assumption, for example, can have serious repercussions throughout the development process. Returning to the scenario discussed in Fig. 3, the critical decision on whether to use GSM or CDMA hinges on, among other things, the life expectancy of the product. When the product life is very short, the option to provide GSM service is not justified. The marketing team was asked to carefully evaluate their assumption that customers indeed will use the same PDA for 3 years. When their market studies revealed that with rapid changes in the industry, a typical consumer would not use the PDA more than 2 years, the belief in this assumption was revised. With such a revision in our system, the support for the concept on upgradability and therefore, the option of a GSM based protocol will be automatically withdrawn.

Similarly, the dependencies in the network of issues that get discussed, the various alternative solutions considered and their attendant justifications are captured and maintained by autonomous agents with different levels of formality. In the simple case, the system warns the user about changes in relevant concepts on which the concept of interest to him/her depends. In the other extreme, changes to concepts are automatically propagated. A network of dependencies can be set up among any concept described using our system. In our example above, instead of automated propagation of the beliefs due to changes in the assumption, the user may be notified to highlight the decision about using GSM needs to be reexamined.

8. Related Work

Our research has focused on providing support for a collaborative task with emphasis on capturing process knowledge in collaborative systems. Systems such as gIBIS [13] and others that are inspired by this work (e.g., IBE [39]) have a similar goal. They advocate the representation of informal information as a part of a comprehensive project knowledge. These systems provide a hypertext interface to the IBIS model and are constrained by the limitations of IBIS such as the lack of representation of inputs and outcomes of argumentation. Synview [43] is an example of a system that uses Toulmin's argumentation framework [71,72]. This system provides facilities for indexing, evaluating and synthesizing information related to justifications for assertions. In contrast to these systems that are limited to a specific model, we provide a generic framework with which any model (such as the IBIS and its extensions) can be easily developed to represent knowledge components. Further, unlike these systems that focus primarily on informal knowledge, we provide a much tighter integration of formal and informal components of process knowledge. Systems such as Constellations [57] facilitate access to informal knowledge by chunking multimedia information. However, the need to integrate informal information in hypertext with formal models has also been suggested by Bhargava et al. [7]. A primary reason for providing a formal representation of the argumentation process (whenever feasible) is to facilitate automated reasoning with the captured knowledge. Our work, therefore, takes a comprehensive approach to dealing with informal information advocating the creation of formal models to facilitate better integration. As a major objective of our research is to provide automated reasoning to support various stakeholder needs, we provide functionalities similar to those offered by SIBYL [40,41]. Our framework is generic enough to support the DRL model proposed in SIBYL as well as more comprehensive models of process knowledge components.

Our approach for assigning attributes to multimedia objects for indexing and ease of retrieval is similar to the scheme proposed in DEDAL [5]. Our work, however, is much broader in scope in that multimedia objects are used as adjuncts to and in support of more formally defined knowledge components. Unlike DEDAL whose functionality is limited to multimedia information, we provide a wide range of services to address the needs of NPD teams. Such active support beyond passive capture of knowledge, distinguish our system from several organizational memory systems such as gIBIS [13].

Quality function deployment (QFD) (as defined by Hauser and Clausing [34]) is commonly used in NPD [16,34,45,63]. Our approach to capturing process knowledge is highly complementary to this approach. QFD can be used to represent decisions made by a team along dimensions considered important to the customer and the product development team. We contrast our approach in the following ways.

- The output of a QFD process leading to a house of quality (HoQ) involves the consideration of a variety of viewpoints and expertise. Only the final outcomes of these deliberations are represented in

QFD, say as a QFD correlation matrix. Our approach can complement the QFD process by capturing these rich deliberations as well. Thus, the output of deliberations in our system can be HoQ.

- Capture of process knowledge about the QFD process itself can be very valuable in explaining as well as evolving the HoQ necessitated by changes in the alternatives considered in a QFD activity. For example, a design characteristic used in an HoQ may be based on an assumption about the availability or implementability of the technology in the production process. If this assumption is invalidated later in the product development process, the repercussions of this can be easily ascertained by our system. More fundamentally, the explicit identification of knowledge components such as assumptions (which are not directly represented in a QFD process), can be extremely valuable.

Finally, in contrast to group decision support systems that focus on the capture of informal information, we advocate semistructured capture of information. These semistructured knowledge components can contain informal information captured using such systems. In this respect, our work is complementary to such systems in that we provide mechanisms for capturing and managing knowledge represented using richer media.

9. Discussion and Future Work

A field study [74] demonstrates the feasibility and usefulness of capturing semistructured knowledge in a complex problem solving situation, viz., design decision making in large system development projects. The business value of codification and capturing semistructured knowledge was also found in a recent study of 120 projects across a cross section of firms [33]. However, some components of process knowledge do not easily lend themselves to structuring. Also, an organization may use a variety of means (such as groupware systems) to facilitate informal and formal interactions. Integration of our system with outputs from such sources of knowledge is essential for a comprehensive representation of process knowledge. As an initial step, we provide facilities for linking structured and unstructured knowledge components. Extending this approach, providing links to textual outputs of deliberations from traditional groupware systems is the subject of future work. We are currently investigating integration with a system designed to capture unstructured discussions. Further, we are also developing mechanisms to integrate the tool with document management tools. Users will be able to highlight segments of documents and create concepts directly from them into the knowledge base. Such a non-intrusive capture and maintenance of process knowledge can mitigate the overhead involved. Another area of current exploration is the use of concept classification techniques for identifying candidate concepts from text documents [11]. Using these techniques, text documents ranging from e-mail exchanges to group support system outputs to project documentation can be used to identify potential concepts. Using the interfaces to document management tools, these concepts can be imported into our knowledge base. We are currently investigating the capture of process knowledge from electronic mail exchanges. Future work will include the use of concept learning techniques to learn concept descriptions from past experiences of commonly occurring design situations.

Our approach aids in sharing of knowledge and experience to aid organizational learning. The knowledge base of *semistructured* experiential knowledge lends itself well to the use of induction and case based reasoning approaches to learning. As a first step, we are using concept learning techniques to learn concept descriptions from past experiences of commonly occurring design situations.

The empirical evaluation of the usefulness of our approach is being carried out by incorporating our models and mechanisms in several business environments. Our evaluation would focus on the feasibility of capturing and maintaining product development knowledge in information systems product development activities.

The work presented here makes the following contributions:

- We identify problems associated with KM specifically in the context of NPD by cross-functional collaborative teams.
- We map these problems to broad solutions and subsequently translate these into specific KM system characteristics and requirements.

- A prototype KM system that meets these requirements is used to demonstrate the implementation of our proposed solutions.
- We validate the relevance of the proposed approach by exercising the system with scenarios of usage drawn from an industrial case study.
- The implementation illustrates an infrastructure that can be tailored to support various models of collaborative KM.

In conclusion, acquiring comprehensive contextual knowledge about product development process can be very valuable in supporting the needs of the various participants. Such knowledge should integrate both formal as well informal components. Further, it should be able to represent the linkages or relationships among various components. Due to the high 'overhead' involved this process, intelligent support for the capture, use, and maintenance of process knowledge is essential for project success.

References

[1] P.S. Adler, Technology and the Future of Work, Oxford Univ. Press, New York, 1992.
[2] J. Allen, P. Hayes, Moments and points in an interval based temporal logic, Computational Intelligence 5 (1989) .
[3] W. Applehans, A. Globe, G. Laugero, Managing Knowledge: A Practical Web-based Approach, Addison-Wesley, Reading, MA, 1999.
[4] W.R.J. Baets, Organization Learning and Knowledge Technologies in a Dynamic Environment, Kluwer Academic Press, Boston, MA, 1998.
[5] C. Baudin, C. Sivard, M. Zweben, Recovering rationale for design changes: a knowledge-based approach, in: Proceedings of the IEEE Proceedings, Los Angeles, CA.
[6] R. Bettis, M. Hitt, The new competitive landscape, Strategic Management Journal 16 (1995) .
[7] H. Bhargava, R. Krishnan, A. Whinston, On integrating collaboration and decision technologies, Journal of Organizational Computing 3 (4) (1994) .
[8] R.E. Bohn, Measuring and managing technological knowledge, Sloan Management Review 36 (1994) .
[9] U. Borghoff, R. Pareschi, Information Technology for Knowledge Management, Springer, New York, 1998.
[10] K. Carley, Organizational learning and personnel turnover, Organization Science 3 (1) (1992) .
[11] H. Chen, P. Hsu, R. Orwig, L. Hoopes, J. Nunamaker, Automatic concept classification of text from electronic meetings, Communications of the ACM 37 (10) (1994) .
[12] R. Comerford, Pocket Computers Ignite OS Battle, IEEE Spectrum, May 1998.
[13] J. Conklin, M. Begeman, gIBIS: a hypertext tool for exploratory policy discussion, ACM Transactions on Office Information Systems 6 (1988) .
[14] N.R. Council, Improving Engineering Design: Designing for Competitive Advantage, National Academy Press, Washington, DC, 1991.
[15] A.W. Court, The relationship between information and personal knowledge in new product development, International Journal of Information Management 17 (2) (1997) .
[16] D. Daetz, W. Barnard, R. Norman, Customer Integration: The Quality Function Deployment (QFD) Leader's Guide for Decision Making, Wiley, New York, 1995.
[17] P. Darke, G. Shanks, M. Broadbent, Successfully completing case study research: combining rigor, relevance and pragmatism, Information Systems Journal 8 (4) (1998) .
[18] T. Davenport, D. DeLong, M. Beers, Successful knowledge management projects, Sloan Management Review 39 (2) (1998) .
[19] T.H. Davenport, Process Innovation: Reengineering Work through Information Technology, Harvard Business School Press, Boston, 1993.
[20] T.H. Davenport, L. Prusak, Working Knowledge: How Organizations Manage what They Know, Harvard Business School Press, Boston, 1998.
[21] G.I. Doukidis, F.W. Land, G. Miller, Knowledge-based Management Support Systems, Ellis Horwood, Chichester, NY, 1989.
[22] P. Drucker, The Post Capitalist Society, Harper Business Press, New York, NY, 1993.
[23] D. Duffy, Knowledge Champions: What Does it Take to be a Succesful CKO?, CIO Enterprise, 1998.
[24] W.E. Eder, Information systems for designers, in: Proceedings of the Proc. International Conference in Engineering Design, Harrowgate, pp. 1307–1319.
[25] L. Fahey, L. Prusak, The eleven deadliest sins of knowledge management, California Management Review 40 (3) (1998) .
[26] J. Favela, C.J. Jerome, Accessing corporate memory in networked organizations, in: Proceedings of the Hawaii International Conference on System Sciences, Hawaii, pp. 181–190.
[27] G. Fisher et al., Making argumentation serve design, Human Computer Interaction 6 (1991) .
[28] G. Fisher et al., Supporting indirect collaborative design with integrated knowledge based design environments, Human Computer Interaction 7 (1992) .
[29] J. Galegher, R.E. Kraut, C. Egido, Intellectual Teamwork: Social and Technological Foundations of Cooperative Work, Lawrence/Erlbaum Associates, Hillsdale, NJ, 1990.
[30] L.H. Glickenhaus, Knowledge transfer: breaking boundaries with web-based networks, Law Practice Management 25 (2) (1999) .
[31] F. Golshani et al., Proceedings of the Sixth International Conference on Information and Knowledge Management: CIKM'97, November 10–14, 1997, Las Vegas, NV, ACM Press, New York, 1997.
[32] J. Grudin, Evaluating opportunities for design capture, in: M. Carroll (Ed.), Evaluating Opportunities for Design Capture, Lawrence Erlbaum Associates, Mahwah, NJ, 1996.

[33] M. Hansen, N. Nohria, T. Tierney, What's your strategy for managing knowledge?, Harvard Business Review, March–April 1999.

[34] Hauser, Clausing, The house of quality, Harvard Business Review 3 (1988) .

[35] M. Iansiti, A. MacCormack, Developing products on internet time, Harvard Business Review, September–October 1997.

[36] A.R. Jassawalla, H.C. Sashittal, An examination of collaboration in high technology new product development process, Journal of Product Innovation Management 15 (1998) .

[37] A. Khurana, Managing complex production processes, Sloan Management Review, Winter, 1999.

[38] N. Krishnakumar, R. Jain, Mobility support for sales and inventory applications, in: T. Imeluski, H. Korth, (Eds.), Mobility Support for Sales and Inventory Applications, Kluwer Academic Publishers, 1996, pp. 571–594.

[39] M. Lease, M. Lively, J. Leggett, Using an issue based hypermedia system to capture the software life cycle process, Hypermedia 2 (1) (1990) .

[40] J. Lee, Sybil: A qualitative decision management system, in: Whinston, Shellard (Eds.), Sybil: A Qualitative Decision Management System, MIT Press, Cambridge, MA, 1990.

[41] J. Lee, Design rationale capture and use, AI Magazine 14 (1993) .

[42] D. Leonard-Barton, S. Sensiper, The role of tacit knowledge in group innovation, California Management Review 40 (3) (1998) .

[43] D.G. Lowe, Cooperative structuring of information: the representation of reasoning and debate, International Journal of Man Machine Studies 23 (1985) .

[44] J.G. March, Exploration and exploitation in organizational learning, Organization Science 2 (1) (1991) .

[45] S. Mizuno, Y.o. Akao, K. Ishihara, QFD, the Customer-driven Approach to Quality Planning and Deployment, Asian Productivity Organization, Tokyo, Japan, 1994.

[46] K. Morik et al., Knowledge Representation and Organization in Machine Learning, Springer, Berlin, 1989.

[47] R. Morrison, J. Kennedy, Advances in databases, in: 14th British National Conference on Databases, BNCOD 14, Edinburgh, Scotland, United Kingdom, July 3–5, 1996 Proceedings, Springer, Berlin, 1996.

[48] J. Mylopoulos, Telos: representing knowledge about information systems, ACM Transactions on Information Systems 8 (1990) .

[49] C.K. Nicholas, J. Mayfield, Intelligent Hypertext: Advanced Techniques for the World Wide Web, Springer, Berlin, 1997.

[50] I. Nonaka, The knowledge creating, Harvard Business Review, November–December 1991.

[51] I. Nonaka, N. Konno, The concept of 'Ba': building a foundation for knowledge creation, California Management Review 40 (3) (1998) .

[52] I. Nonaka, H. Takeuchi, The Knowledge-creating Company: How Japanese Companies Create the Dynamics of Innovation, Oxford Univ. Press, New York, 1995.

[53] W. Orlikowski, J. Baroudi, Studying information technology in organizations: research approaches and assumptions, Information Systems Research 2 (1991) .

[54] J.B. Quinn, P. Anderson, S. Finkelstein, Managing professional intellect: making the most of the best, Harvard Business Review, March–April 1996.

[55] B. Ramesh, V. Dhar, Supporting systems development using knowledge captured during requirements engineering, IEEE Transactions on Software Engineering, 1992.

[56] B. Ramesh, K. Sengupta, Multimedia in a design rationale decision support system, Decision Support Systems 15 (1995) .

[57] V. Rao, R. Goldman-Segall, Capturing stories in organizational memory systems: the role of multimedia, in: Proceedings of the Hawaii International Conference on System Sciences, Hawaii, pp. 333–341.

[58] Robillard, The role of knowledge in software development, Communications of the ACM 42 (1) (1999) .

[59] R. Ruggles, The state of the notion: knowledge management in practice, California Management Review 40 (3) (1998) .

[60] R. Sanchez, A. Heene, Strategic Learning and Knowledge Management, Wiley, Chichester, 1997.

[61] P.M. Senge, The Fifth Discipline: The Art and Practice of the Learning Organization, Doubleday/Currency, New York, 1990.

[62] M. Shaw, M. Fox, Distributed artificial intelligence for group decision support, Decision Support Systems 9 (1993) .

[63] M.L. Shillito, Advanced QFD: Linking Technology to Market and Company Needs, Wiley, New York, 1994.

[64] M. Song, M. Montoya-Weiss, Critical development activities for really new versus incremental products, Journal of Product Innovation Management 15 (1998) .

[65] S. Srivastva, The Executive Mind, Jossey-Bass, San Francisco, 1983.

[66] E.W. Stein, Z. Vladimir, Actualizing organizational memory with information systems, Information Systems Research 6 (1) (1995) .

[67] G. Szulanski, Exploring internal stickiness: impediments to transfer of best practice within the firm, Strategic Management Journal 17 (1996) .

[68] D. Teece, Research directions for knowledge management, California Management Review 40 (3) (1998) .

[69] R.E. Teigland, C. Fey, J. Birkinshaw, Knowledge Dissemination in Global R&D Operations: Case Studies in Three Multinationals in the High Technology Electronics Industry, Stockholm School of Economics, Stolkholm, 1998.

[70] D.R. Tobin, The Knowledge-enabled Organization: Moving from 'Training' to 'Learning' to Meet Business Goals, Amacom, New York, 1998.

[71] S. Toulmin, The Uses of Arguments, Cambridge Univ. Press, Cambridge, UK, 1958.

[72] S. Toulmin, R. Rieke, A. Janik, An Introduction to Reasoning, MacMillan, New York, 1984.

[73] M.A. Von Glinow, S.A. Mohrman, Managing Complexity in High Technology Organizations, Oxford Univ. Press, New York, 1990.

[74] K.B. Yakemovic, E.J. Conklin, Report on a development project use of issue based information system, in: Proceedings of the Computer Supported Cooperative Work, pp. 105–118.

[75] R. Yin, Case Study Research: Design and Methods, Sage Publications, Thousand Oaks, CA, 1994.
[76] M. Zack, An architecture for managing explicated knowledge, Sloan Management Review, Spring, 1999.

Balasubramaniam Ramesh is an Associate Professor of Computer Information Systems at Georgia State University. Dr. Ramesh received his PhD degree in Information Systems from New York University. His research work has appeared in several leading conferences and journals including the IEEE Transactions on Software Engineering, IEEE Expert, Annals of Software Engineering, Annals of Operations Research, Concurrent Engineering — Research and Applications, Decision Support Systems and Journal of Systems Integration. His research interests include supporting collaborative work with artificial intelligence, decision support and multimedia technologies in the areas of requirements engineering and traceability in systems development, concurrent engineering, NPD, knowledge management, and business process redesign.

Amrit Tiwana is currently a doctoral student in the Department of Computer Information Systems at J. Mack Robinson College of Business, Georgia State University, Atlanta. His research interests include knowledge management, organizational learning, and new product development. He is also the author of The Knowledge Management Toolkit: *Techniques and Strategies for Building Knowledge Management Systems* (Prentice Hall, 1999).