# On Identifying Deficiencies in a Knowledge Management System

**Prashant Baheti[1], Laurie Williams[1], Aldo Dagnino[2], Andrew Cordes[2]**
[1]North Carolina State University, Department of Computer Science
{ppbaheti, lawilli3}@unity.ncsu.edu
[2]ABB, Inc.
{aldo.dagnino, andrew.cordes}@us.abb.com

**Abstract.** The Abou-Zeid Knowledge Management Reference Model provides a structure for the elements necessary for a holistic knowledge management support system. The model underscores the importance for an effective system to consist of three layers: business aspects and associated knowledge nuggets; processes which enable the manipulation and update of these nuggets; and technology that makes this manipulation and access to the nuggets effective and straightforward. This paper supports the Abou-Zeid model through its application at ABB, Inc. in the implementation of the ABB Software Process Initiative (ASPI) Knowledge Base. The ASPI group has strengthened the process and the technology aspects of its previous experience database. The reengineered system demonstrates strength at all three Abou-Zeid layers, leveraging the utility of the captured knowledge nuggets. As structured, initial indications support the reengineered system as significantly more valuable to ABB employees than had been the earlier system.

## 1. Introduction

Organizations need to identify, manage, and exploit their knowledge assets [1]. Often, a large amount of organizational knowledge leaves the organization when an employee and the knowledge he or she possesses leave the company. Additionally, the knowledge within an organization should be available to whoever needs it so that repetitive work can be avoided. By using a knowledge base, organizations can leverage both the positive and negative experiences of its employees.

A software organization's main asset is its intellectual capital, and Knowledge Management (KM) capitalizes on the retention, impact, and distribution of this intellectual capital [2]. Software engineering projects are often challenged by poorly defined requirements, frequent staff turnover, and volatile hardware and software platforms [3]. Software organizations possess knowledge in different areas; each knowledge area is critical for achieving business goals. Some of these areas include documented knowledge about new technology, the organization's domain, local policies and practices, and the knowledge of the employees [2]. Organizations often face problems identifying the content, location, and use of their knowledge because it exists in different forms in the organization.

In the software engineering (SE) domain, KM can be defined as "a set of activities, techniques, and tools supporting the creation and transfer of SE knowledge throughout the organization" [4]. Many organizations realize that KM is an integral part of their process improvement initiative, which involves facilitating access to process-related artifacts that support SE and quality management techniques [4]. For example, software process knowledge can be externalized (converted from tacit to explicit) [5] by having a set of standard templates for different phases of development and support activities, such as a software requirements specification template.

This paper discusses the reengineering of a web-based software and product engineering knowledge base developed at ABB, Inc.[1] The initial version of the knowledge base, the Experience Database (henceforth referred to as ABB-1), contained experiences and training packages. However, the system lacked critical elements necessary for effective update and use by development units. The reengineered knowledge base, called the ASPI Knowledge Base (henceforth referred to as ABB-2), contains process templates, procedures, and documented valuable experiences of the employees in development units. More importantly, the supporting technology has been enhanced and processes have been developed to improve the utility of the system to ABB employees.

Our experiences reengineering System 1 have heightened our awareness that a comprehensive and successful knowledge management support system is multifaceted. Our findings support the framework established by the Knowledge Management Reference Model suggested by Abou-Zeid [6]. The rest of the paper is organized as follows. Section 2 provides background information on knowledge management support systems, particularly those in the SE domain. Section 3 describes the Abou-Zeid KM Reference Model. Sections 4 and 5 describe the initial and reengineered ABB knowledge bases. Section 6 portrays the implementation of this reengineered knowledge base within the context of the Abou-Zeid reference model. Finally, Section 7 provides the conclusion and plans for future work.

[1] http://www.abb.com

## 2. Related Work

This section discusses various industry efforts to support KM within their organizations and discusses different KM models typically used in the software engineering domain.

### 2.1 KM in Software Engineering

Several companies have developed software engineering KM support systems (KMSS), as we now discuss. An objective of the KM initiative at VTT Electronics [4] was that the KM system needed to have minimal impact on the software development and processes. Therefore, the solution could not require new technologies. VTT utilized software process improvement (SPI) experts as knowledge-capturing agents, rather than soliciting knowledge from software developers. The SPI experts gleaned information from project final reports, error databases, discussion forums, and through speaking with people in the organization. This information was provided to customer projects on demand by packaging this knowledge for easy access. The required knowledge for the project was provided just-in-time. This needs-based KM approach adopted by VTT worked well for one of the customer projects inside the company. The customer project could retrieve the needed knowledge from the delivered knowledge package.

Infosys Technologies Ltd. is a software services company that manages organization-wide knowledge using its three centrally-operated knowledge repositories [7]: the Knowledge Shop (K-Shop), Process Asset Database, and People Knowledge Map. The K-shop provides employees access to its resources related to technology, domain, trends, culture, project experiences, internal and external literature through a web interface. The Process Asset Database captures the "as-is" project deliverables, such as project plans, design documents, and test plans. The People Knowledge Map is an intranet-based system for employees to search and locate experts in different fields. The K-shop has helped Infosys to increase its productivity by three percent due to effective reuse and to reduce defect levels by as much as 40 percent.

Similarly, International Semiconductor Technology is an Integrated Circuit assembly and testing company where knowledge intensity, specificity, and volatility are comparable to those in SE and other knowledge-intense fields. This organization utilized a KMSS to assist with knowledge creation, update, sharing, and reuse [8]. Their KMSS consists of a lessons-learned repository, a case repository, and an organizational directory. Soliciting experts for answers to problems that arise supports knowledge creation. The experts give their analyses, comments, and/or recommendations. The experts can then discuss the problems using groupware until a solution is reached. At this time, the exchange becomes "validated" knowledge, which is stored in the lessons-learned repository. By supporting the sharing of critical, task-specific knowledge previously held by individuals, the company aims at developing better systems at a faster pace.

These examples suggest that the structure of software engineering-related KM initiatives depends upon the context of the organization. However, in each case the primary goal was to have easy access to tacit and externalized knowledge of the company. Each KMSS usually involves a centralized repository of knowledge resources with an accessible interface to the users. The other goal was to facilitate socialization [14] by creating an environment conducive to interaction between persons in the company. For example, whenever a person has a problem that can be solved by some other expert in the same organization, he or she should be able to find and contact that expert and/or some documented knowledge provided by that expert.

### 2.2 Knowledge Management Frameworks

Many organizations have built their own KM frameworks to oversee KM processes, methodologies, tools and techniques within the company. Most of the frameworks are prescriptive in nature [9], implying that these frameworks are not holistic. Instead, the frameworks are task-based because they just consider the tasks that facilitate movement of knowledge within the organization. These frameworks do not consider factors such as linking knowledge management to business objectives, feedback loops within the organization, and cultural factors. Examples of organizations who have authored such frameworks include Ernst and Young, Knowledge Associates, and The Knowledge Research Institute, Inc. [9]

Even the descriptive frameworks that characterize knowledge management have been unable to view KM holistically. For example, "the National Technical Institute of Greece, Andersen Consulting, and The Delphi Group include cultural factors in their framework but not learning or linkages with strategic business objectives." [9]

Binney [10] developed a KM spectrum which acts as framework that helps organizations in understanding the range of KM options, applications and technologies available to them. This framework groups KM applications according to six groupings: transactional, analytical, asset management based, process based, developmental and innovation/creation-based. Although this framework transcends most of the KM applications available and their enabling technologies, it is again task-based rather than considering multiple facets of the organization.

The Knowledge Management Maturity Model (KMMM®) [11], developed at Siemens AG, does provide a holistic approach to knowledge management. It defines five maturity levels of Knowledge Management, similar to those of the SEI's Capability Maturity Model [12] and defines the various key areas associated with the model. The phases involved in a typical KMMM® project for assessing an organization's maturity level are orientation and planning; motivation and data collection; consolidation and preparation; feedback and consensus; ideas for solutions and action proposals; and report and presentation. We considered guiding our reengineering work with the KMMM®. However, ABB places a higher priority on business value than on achieving a maturity level.

Abou-Zeid [6] composed a knowledge management reference model that is holistic and focuses on business value. We discuss this model more in detail in the following section because we present our research in relation to this model.

## 3. The Abou-Zeid Knowledge Management Reference Model

Abou-Zeid [6] has devised a Knowledge Management Reference Model (KMRM) that aims to answer the question *"How can a KMSS be developed, operated, and assessed?"* The model identifies the processes to be supported by any KMSS. These processes change the state of current knowledge within the organization to a state where knowledge exists in an updated and desirable form for everyone to access. For example, one such process could be identifying the experts in an organization and documenting their views on specific problems to form new knowledge for future reference. The Abou-Zeid KMRM also models the dynamics of these processes, thereby developing a framework of a KMSS that is specific to a particular business and that is "aware of" the issues of that business.

Abou-Zeid's model identifies the processes that should be supported by KMSS through its three-layer reference model. The three layers, cognitive domain layer, functional layers, and resources layer, are each discussed below.

*Cognitive Domain Layer:* The model's first layer, called the cognitive domain layer, addresses entities, called business things (or *B-things*), that relate to business issues or organizational goals. A B-thing may be a concrete or abstract entity. An example of a **B-thing** is **efficient and cost-effective software production**.

Each B-thing is associated with certain knowledge that enables or supports it. These are called knowledge things (or *K-thing*s, akin to knowledge nuggets). The "efficient and cost-effective software production" B-thing is supported through the reuse of already-available resources within the organization. For example, product development templates already developed within the organization should be available to any business unit. The **product development templates** are **K-things**.

*Functional Layer:* In an organization, the B-things are relatively stable; however, the associated K-things are in a state of continual change as knowledge evolves. The second layer, known as the functional layer, includes the processes required to change the state of the K-things. "Since knowledge requirements are difficult to determine fully a priori, associated processes have to be dynamic, evolving, and flexible."[6] These are called the *K-manipulating* processes. The processes supporting the K-manipulating processes [13] are called *K-enabling* processes. The K-enabling processes produce cultural and organizational enabling conditions for the K-manipulating processes. A **K-manipulating process** provides the ability for an employee or a business unit to **contribute a template** to the KMSS, keeping these K-things as current as possible. A **general KM awareness** within the organization can be termed as a **K-enabling process**.

*Resources Layer:* The third layer is the resources layer that consists of the enabling information/communication technologies (ICT) to support the K-manipulating and K-enabling processes. For example, the computing environment of an organization is a part of the resources layer.

In summary, the Abou-Zeid KMRM guides in the implementation of an integrative KMSS.

## 4. The Initial Experience Database

In our research, we studied a knowledge base reengineering effort at ABB. ABB-1 consisted of a Lotus Notes database with a custom-built HTML interface. The database was a collection of documented software development experiences and good practices from ABB companies. Experiences and best practices were collected through the submittal of structured experience "packages." These packages underwent a multi-phased review and approval process before being accepted as entries to the database. The database provided a limited search capability of the experience packages and did not offer a mechanism for identifying pertinent external resources and web links.

This experience database was a fairly ineffective K-thing. The problems with this experience database are listed below:

1. The review process for the experiences was too cumbersome. To submit an experience, an employee needed to first submit an abstract to a review board and, if approved, send the actual experience formatted according to a detailed packaging template. The experience packages then underwent a second review before final acceptance and publish. Turnaround times of a month or more

were not uncommon for publishing an experience. (unsuccessful K-manipulating process)

2. The database was not frequently updated because no one was formally responsible for it, further increasing the turnaround time. (unsuccessful K-manipulating process)
3. There was no incentive to submit experiences. (lack of K-enabling process)
4. The experiences were approved only if they were a corporate-level best practice, limiting the number of experiences in the knowledge base. (restrictive K-manipulating policy)
5. The experience base was not well publicized. (unsuccessful K-enabling process)
6. It was not easy to find relevant articles in the database because the structure was not intuitive. (ineffective technology)
7. The database was limited to ABB experiences; there were no external references to supporting material. (restrictive K-manipulating policy)

In summary, the function and resources layers of the initial system were lacking, reducing the utility of the system.

## 5. The Reengineering Effort

Due to the inefficacy of ABB-1, a KMSS reengineering effort was launched. The main author implemented a new knowledge base in the Lotus Notes/Domino environment on a Windows platform. The knowledge base is a Lotus Notes database with a template-driven Web interface for easy access to the knowledge artifacts within the database. The knowledge base was developed using the Common ABB Web Platform (CAWP), an application owned by the ABB Corporate Communication division; CAWP was also used to develop the ABB intranet. The use of CAWP provided visual consistency for employees. The contents of the knowledge base are available to all the employees via a well-known link on the company intranet.

The knowledge base consists of product development information for various ABB business units. This includes the various experiences submitted by the employees of ABB and the procedures, guidelines, templates and training materials that have been successfully used by ABB product development organizations. Change agents like developers and project managers can retrieve examples of processes, templates, and guidelines from the knowledge base. Contributions to the knowledge base can be submitted by anyone within the organization. The structure of the knowledge base ties most entries to two important reference points for the company: the ABB Gate Model [14, 15] and the SEI Capability Maturity Model Integration (CMMI) process areas [16].

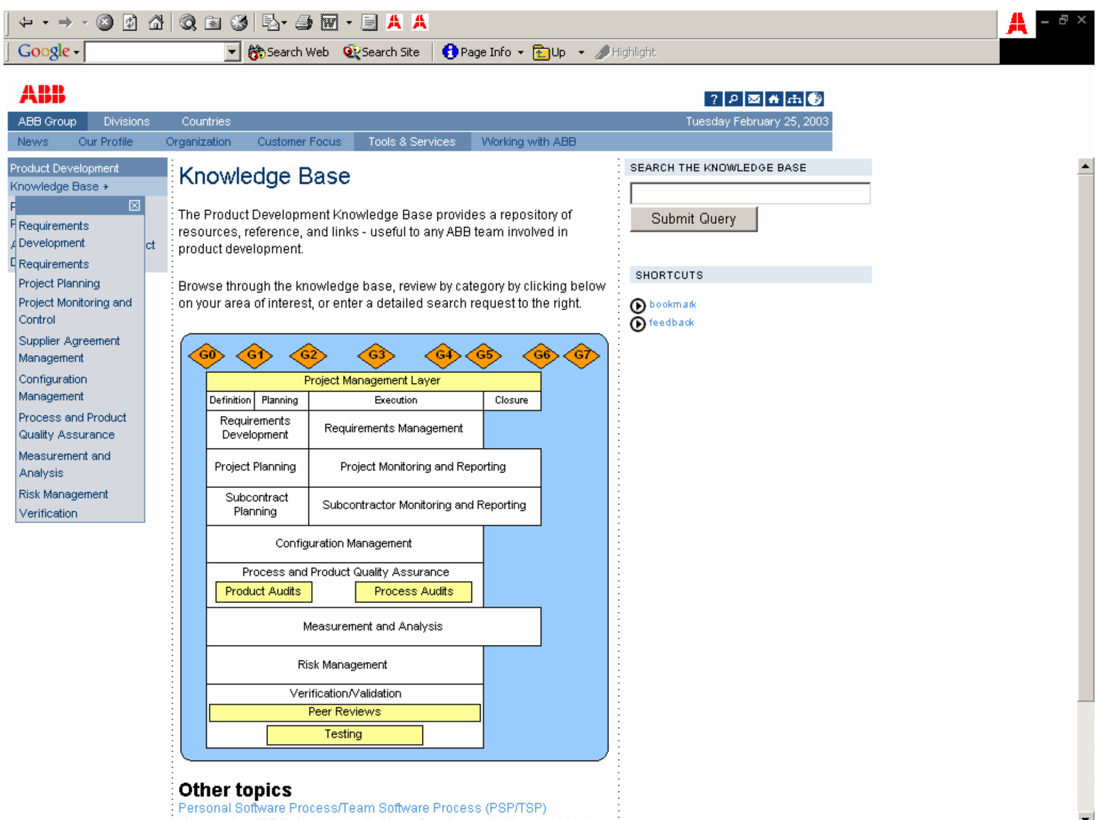


**Figure 1: Main Page of ASPI Knowledge Base (ABB-2)**

Figure 1 provides a screenshot of the main page of ABB-2. The left menu contains links to various process areas within the knowledge base, such as requirements management. The main page displays all the gates of the ABB Gate Model as diamonds that link to resources about each gate. Below each gate is the set of process areas that are related to that gate in the development cycle. Each of these process areas leads to a page of resources; each resource page contains links to ABB experiences, procedures, guidelines, templates and training materials, and other external (non-ABB) references. The knowledge base provides an effective search capability to query for particular documents, templates or keywords. A mechanism is provided for collecting feedback from the users so their questions and concerns can be addressed.

User-friendliness was a main priority in the knowledge base design. Based upon user feedback, the user interface design was modified several times to ensure that the knowledge base was easy to access and intuitive in the way it was organized. Another key objective was the provision of an employee incentive program to encourage contributions to this knowledge base; a list of contributors is posted on the front page of the knowledge base.

Knowledge base contents are regularly updated and new experiences submitted by the employees are incorporated after an efficient yet careful review. A review board, comprised of experts from the ASPI group, is responsible for this review process. The turnaround time is much shorter than that for the prior experience database, because the review board is formally responsible for the knowledge base. The approved submissions are posted on the knowledge base within a week. All ABB business units were asked to submit development process/practice artifacts that they have developed and found to be useful. Specifically, software development process templates were solicited and incorporated into the knowledge base. Subsequently, the business units can readily obtain these artifacts from one central repository. ABB-2 is now fully functional within the organization.

## 6. Mapping to the Abou-Zeid Model

ABB started its KM initiative in 2000, gaining more attention recently. In 2002-2003, a product development knowledge base was created and implemented for more effective utilization of the experience database K-thing. The reengineered ABB-2 system is now discussed in the context of the three layers of the Abou-Zeid Model.

### 6.1 Cognitive Domain Layer

The model of ABB's cognitive domains is shown in a UML diagram in Figure 2. The Product Development Organizations within ABB form the enterprise of the KM initiative; they are the primary targets for ABB-2. The external B-things (generalizations of the generic B-thing) that this enterprise interacts with are its customers, suppliers, competitors and the market. For example, customers are a primary source of requirements to design and develop products. It is essential to have knowledge on how to elicit and manage requirements from the customer.

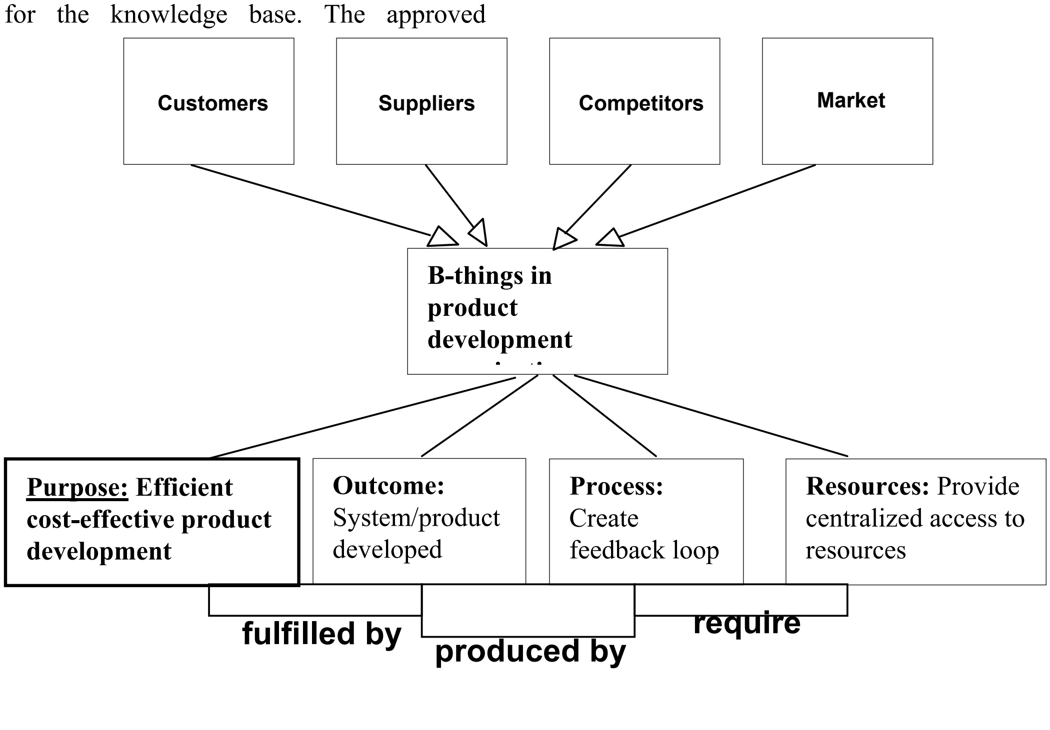


**Figure 2. Model of ABB's cognitive domains**

Often, sub-contractors are used to assist the development group and specific knowledge is used to manage subcontractors. ABB-2 provides knowledge about how to investigate competitors' intellectual property and products. Similarly, knowledge on how to incorporate market information into the development organization is presented in ABB-2.

An internal ABB B-thing is to develop new products or systems in an **efficient and cost effective manner**. The outcome of a product development organization is the developed product or system that is ready to be sold to a customer or market or that is ready for production. Knowledge is essential to the people in the development organizations to perform their work and a feedback loop is needed to incorporate experiences and knowledge into the knowledge repository that is accessed by the developers. As shown in the diagram, the B-thing consists of a purpose, outcome, process, and resources.

Figure 3 displays a simplified version of the ABB-2 in the context of Abou-Zeid KMRM. In this paper, we examine one internal B-thing, "efficient and cost effective product development." The set of actions required to address this B-thing are:

1. Providing centralized and easy access to all the experiences submitted by the employees within the organization. These experiences are integrated with the product-development related resources which were classified according to both the ABB Gate Model [14, 15] and SEI Capability Maturity Model Integration (CMMI) process areas [16].
2. Avoiding unnecessary development by facilitating reuse.
3. Making training materials developed by the ASPI group available to all business units.
4. Providing a mechanism for feedback and answering queries of the business units with respect to any of the three items listed above.

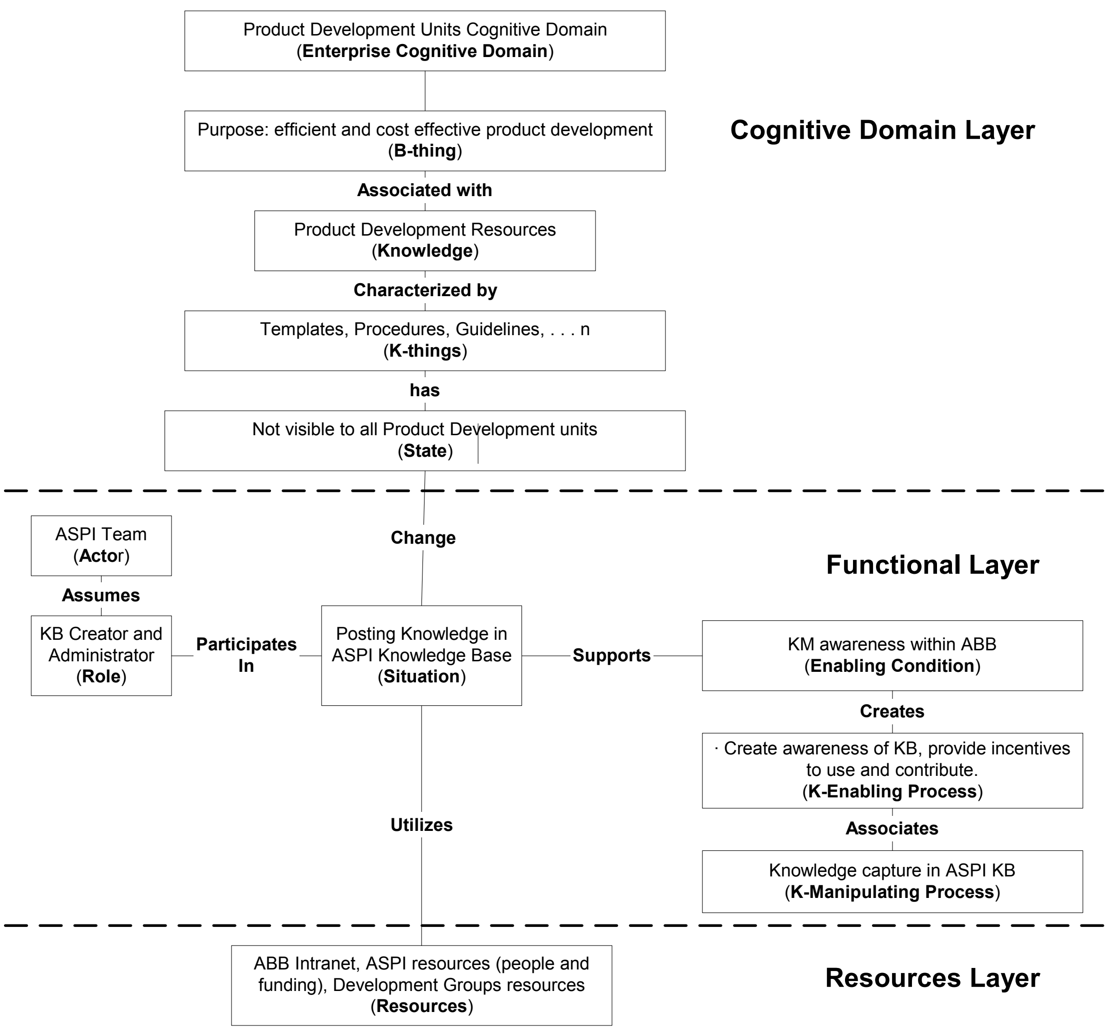


**Figure 3: ABB Knowledge Management Model**

The K-things that are associated with the B-thing are:

- Experiences in ABB-1;
- Templates for the different phases of product development;
- Training materials; and
- Tacit knowledge of the experts.

### 6.2 Functional Layer

The ASPI group, which is responsible for the KMSS reengineering effort, owns the knowledge associated with the various B-things for the product development organizations. The group assumes the role of the administrator for ABB-2 and is responsible for eliciting knowledge from various sources, including databases, employees, and reports and posting the knowledge to ABB-2 after a careful review. This layer is responsible for converting the K-things to a visible state so that any employee on the intranet of ABB can easily access them.

The K-manipulating processes are:

- Identifying product-development related resources to be made available through ABB-2;
- Making training materials available through ABB-2;
- Setting up a review board, which controls new additions to ABB-2; and
- Publicizing ABB-2 once within the organization.

The K-enabling processes can be identified as:

- Spreading awareness about the KM initiative taken up by the ASPI group among other business units; and
- Providing incentives to the employees to contribute to the knowledge base.

### 6.3 Resources Layer

The resources layer consists of the infrastructure of ABB, including the intranet of ABB, the Lotus Notes/Domino development environment and CAWP, the web development tool. The ASPI resources, including people employed for the project and funding the KM initiative also falls into this layer.

## 7. Conclusion and Future Work

By sharing our experiences, we aim to make organizations aware of the importance of all three of the layers of the Abou-Zeid model. A retrospective analysis of ABB-1 based upon the Abou-Zeid model revealed significant deficiencies in the functional and resources layers. Because of these deficiencies, ABB did not effectively capitalize on the resources invested in this initial KM system. These deficiencies were addressed in the reengineering effort, resulting in the more effective ABB-2. A comparison of the processes and things at each level can be found in Table 1. Note the stability of the B-thing, the addition of K-things and K-processes, and enhancement of the ICT.

Initial feedback on ABB-2 has been positive. ASPI team members and change agents at ABB organizations planning process improvements have found the knowledge base to be easy to find on the intranet, easy to use, and full of information and resources supportive of their tasks.

**Table 1: Comparison Chart**

| **Things and Processes (Level)** | **Experience Database (ABB-1)** | **ASPI Knowledge Base (ABB-2)** |
|---|---|---|
| B-things (1) | Efficient and cost-effective product development | Efficient and cost-effective product development |
| K-things (1) | • Experiences<br>• Training materials | • Experiences<br>• Templates<br>• Training materials<br>• Tacit knowledge |
| K-manipulating processes (2) | • Cumbersome review process<br>• Not well publicized<br>• Difficult to navigate<br>• No frequent updates | • Identify resources<br>• Centralized access to resources<br>• Efficient review<br>• Proper publicity<br>• Frequently updated |
| K-enabling processes (2) | No K-enabling processes identified | Awareness of the KMSS and recognition for contribution |
| ICT (3) | Lotus Notes | Lotus Notes, CAWP |

The structure of the knowledge base, relating development activities to CMMI process areas and ABB Gate Model gates, has simplified the knowledge search process as these two are well known within the organization. New contributions to the knowledge base are reviewed and incorporated within one week and initial web statistics show a broad range of users across multiple country sites. We are continuing the evaluation of ABB's knowledge management initiative through a detailed quantitative assessment. The three aspects being monitored are usage, effectiveness and return on investment.

*Usage:* Usage metrics include the number of people using the knowledge base and the frequency of use gathered via server logs. We will also analyze the profiles of the people accessing the knowledge base, which includes factors such as position in the organization and group to which they belong. This will help us to better tailor the knowledge base to the employees who access it.

*Effectiveness:* Effectiveness is measured through feedback from the users of the knowledge base. This feedback will help us analyze the average percentage of times the employees find what they want from the knowledge base. Effectiveness metrics include ease of use, accessibility of the knowledge base, and average amount of time required to find the required resource.

*Return on investment (ROI):* To examine ROI, we have solicited all business units to provide all the templates they presently use for their development purposes. When we feel this collection is complete, we can find out how many different copies/versions of the same template have been created in the organizations. Feedback from the product development organizations about the time taken to develop their templates will help us find an estimate of the average amount of development time required for each template. The time saved henceforth by avoiding redevelopment of templates can be factored in calculating ROI. The other factor is time saved in the future due to presence of a central repository of resources. This involves feedback from the product development organizations concerning the time required to access resources without the presence of ABB-2, compared to that required with the presence of ABB-2. We are investigating additional ROI metrics for ABB-2.

**Acknowledgements**

We would like to thank Dr. Abou-Zeid for helping us by further explaining his model in our context and for reviewing and improving this paper.

## References:

[1] M. Lindvall, I. Rus, R. Jammalamadaka, and R. Thakker, "Software tools for Knowledge Management," DoD Data Analysis Center for Software, Rome, N.Y. 2001.

[2] I. Rus and M. Lindvall, "Knowledge Management in Software Engineering," *IEEE Software*, vol. 19, no. 3, pp. 26-38, May/June 2002.

[3] T. K. A. Hamid, "The Slippery path to productivity improvement," *IEEE Software*, vol. 13, no. 4, pp. 43-52, 1996.

[4] S. Komi-Sirvio, A. Mantyniemi, and V. Seppanen, "Toward a Practical Solution for Capturing Knowledge for Software Projects," *IEEE Software*, vol. 19, no. 3, pp. 60-62, May/June 2002.

[5] H. Takeuchi and I. Nonaka, *The Knowledge Creating Company:How Japanese Companies Create the Dynamics of Innovation*: New York: Oxford University Press, 1995.

[6] E.-S. Abou-Zeid, "A Knowledge Management Reference Model," *Journal of Knowledge Management*, vol. 6, no. 5, pp. 486-499, 2002.

[7] S. Ramasubramaniam and G. Jagadeesan, "Knowledge Management at Infosys," *IEEE Software*, vol. 19, no. 3, pp. 53-55, May/June 2002.

[8] C.-P. Wei, P. J.-H. Hu, and H.-H. Chen, "Design and Evaluation of a Knowledge Management System," *IEEE Software*, vol. 19, no. 3, pp. 56-59, May/June 2002.

[9] B. Rubenstein-Montano, J. Liebowitz, J. Buchwalter, D. McCaw, and T. K. M. M. Team, "A Systems Thinking Framework for Knowledge Management," *Decision Support Systems*, vol. 31, no. 1, pp. 5-16, May 2001.

[10] D. Binney, "The knowledge management spectrum - understanding the KM landscape," *Journal of Knowledge Management*, vol. 5, no. 1, pp. 33-42, 2001.

[11] K. Ehms and D. M. Langen, "Holistic Development of Knowledge Management with KMMM®." http://www.kmadvantage.com/docs/km_articles/Holistic_Development_of_KM_with_KMMM.pdf, 2002.

[12] M. C. Paulk, B. Curtis, M. B. Chrissis, and C. V. Weber, "Capability Maturity Model for Software, Version 1.1," *IEEE Software*, vol. 10, no. 4, pp. 18-27, July 1993.

[13] V. Krogh, I. Nonaka, and K. Ichijo, *Enabling Knowledge Creation*: Oxford University Press, Oxford, 2000.

[14] R. A. Carter, A. I. Antón, A. Dagnino, and L. Williams, "Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model," presented at IEEE 5th International Symposium on Requirements Engineering (RE'01), Toronto, Canada, 2001.

[15] C. Wallin, F. Ekdahl, and S. Larsson, "Integrating Business and Software Development Models," vol. 19, no. 6, pp. 28-33, November/December 2002.

[16] M. B. Chrissis, M. Konrad, and S. Shrum, *CMMI®: Guidelines for Process Integration and Product Improvement*: Addison Wesley Professional, 2003.