Concept and methods for information enrichment

Written by:

Kary Främling, Helsinki University of Technology

Lutz Rabe, Bremen Institute of Industrial Technology and Applied Work Science

DELIVERABLE NO	DR 7.2
DATE	5th August, 2005
WORK PACKAGE NO	WP R7
VERSION NO.	Final Version 1.0
ELECTRONIC FILE CODE	DR72_v1.0.doc
CONTRACT NO	507100 PROMISE A Project of the 6th Framework Programme Information Society Technologies (IST)
ABSTRACT:	The objective of this document is to propose concepts and methods for information enrichment based on the demands of the PROMISE application scenarios and demonstrators. The work approach utilizes the categorization of tacit and explicit knowledge. Based on the state of the art and pre-existing concepts from the partners a set of concepts and methods for managing and thus, enriching information for each category is presented. Although the provided solutions are answering the particular demands of the application scenarios they still have to be refined and evolved for the further development of the "PROMISE System".

STATUS OF DELIVERABLE		
ACTION	**BY**	**DATE (dd.mm.yyyy)**
SUBMITTED (author(s))	Kary Främling, Lutz Rabe	12.08.2005
VU (WP Leader)	Lutz Rabe	12.08.2005
APPROVED (QIM)	D. Kiritsis	21.08.2005

Revision History

Date (dd.mm.yyyy)	Version	Author	Comments
3.3.2005	0.1	Kary Främling	Initial draft
15-17.3.2005		Kary Främling	
18.5.2005	0.2	Kary Främling	Introduction first version, parts of state-of-the-art
6.6.2005	0.3	Kary Främling	Draft contents for sections 1-3, 6. Section 2.4 written by Gerd Große.
16.6.2005	0.35	Jürgen Anke	Added description of mySAP PLM solution
27.6.2005	0.4	Kary Främling	Wrote contents of section 4, 5 and 6 (except 6.4). These parts should be to 90% finished (except for tacit knowledge in sections 2 and 6 if it is to be included)
1.7.2005	0.5	Kary Främling	Finished section 1, 2 (HUT part), 6 (HUT part) and 8, cleaned up reference list. Included appendices A and B. HUT part estimated to >95% ready
25.7.2005	0.6	Lutz Rabe	Finished section 2.2 (BIBA part) Managing Product Related Knowledge
25.7.2005	0.6	Lutz Rabe	Finished section 2.5 (BIBA part) Preexisting …
30.7.2005	0.7	Altug Metin	Included section 2.4.3 Information enrichment in PDM systems
5.8.2005	0.8	Kary Främling	Draft contents in section 7, other small modifications
5.8.2005	0.9	Lutz Rabe	Finished section 6 (BIBA part) Description of proposed …
8.8.2005	0.9.5	Lutz Rabe	Finished section 7 (BIBA part) Comparison with SOTA … Final edition of section 6 (BIBA part) Description of proposed …
12.8.2005	1.0	Kary Främling	Finished Conclusions section, made small adjustments and corrections.

Author(s)' contact information

Name	Organisation	E-mail	Tel	Fax
Kary Främling	HUT	Kary.Framling@hut.fi	+358505980451	
Olli Pekka Lehtonen	HUT	oplehton@cc.hut.fi		
Andreas Finne	HUT	kfinne@cc.hut.fi		
Lutz Rabe	BIBA	rab@biba.uni-bremen.de	+494212185519	
Gerd Große	Cognidata	Gerd.Grosse@Cognidata.de		
Jürgen Anke	SAP Research	juergen.anke@sap.com	+4935144572304	
Altug Metin	InmediasP	metin@inmediasp.de	+493302559409	

Table of Contents

1 Purpose of this document

This document is a “Main Deliverable” with public dissemination of the PROMISE project. Therefore it provides an overview of concepts and methods for product information enrichment as related to PROMISE application scenarios and demonstrators but without going into details that would reveal confidential information. For the internal needs of PROMISE, it should provide an overview of the state-of-the-art and estimate what requirements can be fulfilled by existing methods. Requirements that cannot be fulfilled by existing methods must be addressed by new concepts and/or methods, for which this document should provide initial functional specifications and indicate potential solutions to them. These specifications and solution proposals are intended as inputs at least for the work packages WPR8 and WPR9.

2 Introduction

The first main objective set out in the PROMISE Description-of-work (DoW), page 8 is the following:

*“To develop new **closed-loop life cycle information flow models** for BOL, MOL and EOL. In order to close the information flow for the entire product life cycle, product and process models will be developed for the BOL, MOL and EOL phases. These models will be integrated into the product life cycle models and will be the foundations of new software modules able to support decision making in various product development related fields such as Product Diagnostics, Preventive Maintenance, End-of-Life, Adaptive Production, Design for X (DFX), where X stands for Use, Manufacturing, Maintenance, Service, Logistics, EOL, etc. These modules will be managed through their integration in a web based Product Lifecycle Management system.*”

In order to achieve this objective, all product-related information generated in the different lifecycle phases has to be stored in such a way that it is easily *accessible* and *usable* in the other lifecycle phases. Different actors (companies, individuals etc.) handle products during their lifetime so, in practice, it is currently a great challenge to make product information accessible when needed. In this work package we mainly focus on the usability of product information. Product information is often grouped under the concepts of *data*, *information* and *knowledge*. For most practical uses, data represents the lowest level of usability and knowledge the highest level of usability. Therefore it is usually an explicit goal to try to refine “raw” data as much as possible towards knowledge as illustrated in Figure 1.

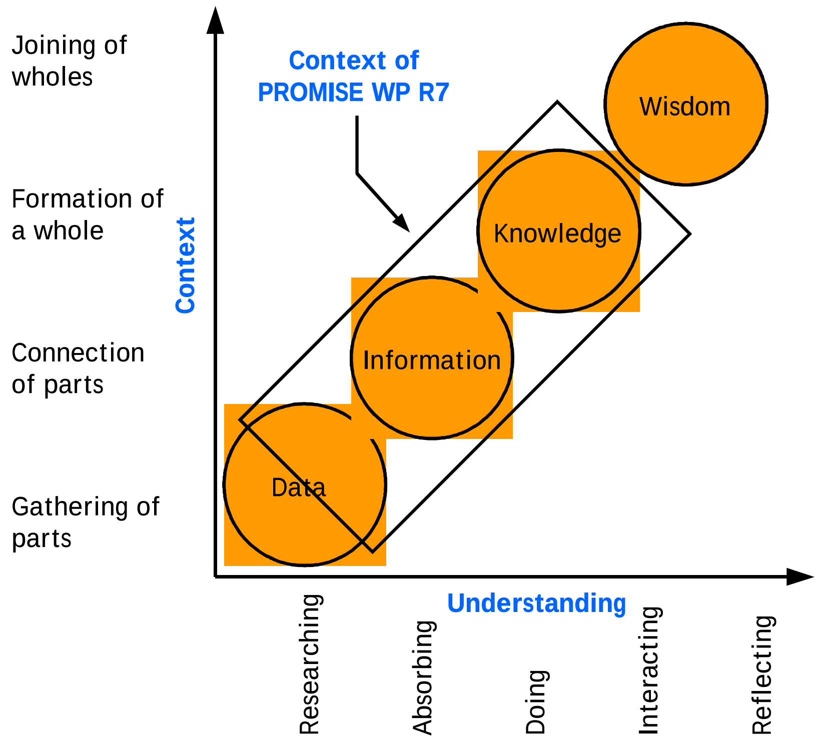


Figure 1. Links between data, information and knowledge in the PROMISE context (adapted from (Clark, 2005))

Wikipedia (http://en.wikipedia.org/) gives the following definitions for the concepts of data, information and knowledge:

"*Data on its own has no meaning, only when interpreted by some kind of data processing system does it take on meaning and become information. People or computers can find patterns in data to perceive information, and information can be used to enhance knowledge. Since knowledge is prerequisite to wisdom, we always want more data and information. But, as modern societies verge on information overload, we especially need better ways to find patterns.*"

The last phrase of this definition concisely expresses one of the major goals of PROMISE work package R7. Today's "intelligent" products are in contact with various stakeholders and gather increasing amounts of data about themselves and their environment from sensors during their lifetime. The information enrichment targeted in task R7.2 consists of

1. capturing tacit knowledge from the stakeholders of a product
2. finding patterns for transforming this sensors data and/or captured tacit knowledge into information and further into knowledge

As mentioned for the first main objective of PROMISE, this information enrichment should enable diagnostics, preventive maintenance etc. In the area of post-sale maintenance, product information is further separated into static data and dynamic data. Static data is typically created during design and manufacturing of the product while dynamic data is created during the use of products. Static data is mainly created by manufacturers during the BOL phase but needs to be used in all phases. Dynamic data is created mainly during the MOL phase but it can be used in all phases for different purposes, e.g. BOL: improving design and manufacturing; MOL: diagnostics, preventive maintenance etc.; and EOL: decisions on reuse/recycling etc. Table 1 attempts to illustrate during what phase static and dynamic data are created or modified.

Table 1. Rough classification of where static and dynamic data is created and improved.

	BOL	MOL	EOL
Static data	Create(1)	Improve based on dynamic data	Improve based on dynamic data
Dynamic data		Create(2)	

The PROMISE work package R7 consists of three subtasks. Task 7.2 mainly focuses on how to represent, store and possibly pre-process product information. This signifies that this task mainly focuses on the "create" cells of Table 1. From Table 1 we can further identify at least the following types of product information:

1. Create(1) refers to representation issues of both tacit and explicit knowledge mainly related to the design and manufacturing of products. Examples: representing relations between different products or organisations, e.g. "part-of", "made-by"; or attributes e.g. "colour", "recommended time of use before replacing".
2. Create(2) refers to the information processing issues related to filtering and pre-processing of dynamic data on different levels. Examples: "if coolant temperature has been somewhat high for over five minutes, then alert the user" or "send usage statistics to backend system every 12 hours".
3. Improving/augmenting static data based on dynamic data. Examples: "vehicle part X is changed after an average of 50000 kms, set recommended replacement interval to 40000 kms" or "engine breakdown was caused by part Y in 80% of cases, improve design of part Y".

Cases one and two are considered to be the main focus areas of Task TR7.2. Case three is not included into the scope of TR7.2, it would rather be considered as a part of Task TR7.1.

2.1 Scope of this document

The PROMISE description of work defines the scope of work package WPR7 as follows: "*With relation to the PROMISE system architecture, information model and related PEID system approaches, concepts and methods for transformation of information to knowledge will be developed in this work package. Furthermore PROMISE specific methodologies, tools and an existing PDM system to foster basic knowledge management processes such as knowledge identification, acquisition, preservation, distribution or assessment, will be selected with particular respect to the exceptional relevance of elusive knowledge for the overall knowledge management process*". This document is the deliverable of subtask WPR7.2, for which the DoW gives the description: "*Based on the PROMISE system architecture, information model (WP R1) and related PEID system (WP R6) approaches appropriate concepts and methods for the Knowledge Management (KM) specific information enrichment will be developed*".

These descriptions are not very explicit. A search on the web for the term "Knowledge Management" (KM) gives a great number of definitions that do not make the scope clearer. Wikipedia gives the following definition: "*KM caters to the critical issues of organizational adaptation, survival, and competence in face of increasingly discontinuous environmental change. Essentially, it embodies organizational processes that seek synergistic combination of data and information processing capacity of information technologies, and the creative and innovative capacity of human beings.*" (http://en.wikipedia.org/wiki/Knowledge_management).

As we can see, there is both an “Information Technology” (IT) and a “human” dimension in KM. Although both dimensions respectively representing branches of KM are dealing partly with similar concepts and methodologies they seem to use different vocabularies and approaches, consequently they will be treated in parallel in different sections.

A formal deviation of this KM approaches will be made by utilizing the concepts of explicit and tacit knowledge.

Explicit knowledge, as the first word in the term implies, is knowledge that has been articulated and, more often than not, captured in the form of text, tables, diagrams, product specifications and so on. In a well-known and frequently cited 1991 Harvard Business Review article titled "The Knowledge Creating Company," Ikujiro Nonaka refers to explicit knowledge as "formal and systematic" and offers product specifications, scientific formulas and computer programs as examples.

Tacit knowledge is a phrase apparently coined by Michael Polanyi in 1962 (Polany, 1962) with his book Personal Knowledge. He described tacit knowledge is a characteristic of experts, who often have no explicit theory of their work: they simply perform skilfully.

In the following tacit knowledge was commonly accepted in the scientific community as an important phenomenon. Nevertheless Gourlay (Gourlay, 2002) stated that there are existing important differences of opinion over many key aspects of tacit knowledge, such as the level at which it is manifested, how it is acquired, what its function is, and whether or not it can be made explicit.

From his review of the literature, Gourlay identifies two issues associated with tacit knowledge. The first is whether tacit knowledge is an individual trait or a trait that can be shared by both individuals and groups, and the second is whether tacit knowledge can be made explicit. For the following work tacit knowledge has been defined as knowledge which has not been made explicit. Although there are more exhaustive concepts in the literature existent this definition is explicitly not differentiating between knowledge which can be articulated and the one which can be not.

The objective of this deliverable is to produce a model for a Product Data and Knowledge Management (PDKM) system that can be used for decision making purposes of the kind identified by the PROMISE application scenarios. This model is here identified under the name PROMISE Decision Support System (PDSS). The PDSS takes into account both the IT and the “human” dimensions of KM in a common framework. The following aspects of the PDSS are in the scope of this document:

- State-of-the-art on KM models and technologies
- Analysis of applicability and suitability of these models and technologies to PROMISE application scenarios
- Proposals for how to use existing models and technologies in PROMISE application scenarios
- Requirements for supplementary functionality
- Proposals for implementing required supplementary functionality

The following are not in the scope of this document:

- Specifications for specific application scenarios
- Selection of tools to use in different application scenarios
- Implementation of software solutions for other than illustration purposes

The main application area of the PDSS is to manage Product Lifecycle Management (PLM) related data, information and knowledge in a way that makes it possible to use all such information as well as possible in all phases of the product lifecycle. The benefits of the PDSS should be improved product quality (design, manufacturing, logistics etc.), improved product maintenance and reliability and improved handling of products at their end-of-life.

2.2 Overview of this document

The document structure is based on PROMISE Quality Manual version 1.0b for documents on "concepts, methods and models". Where applicable, elements from the IEEE Standard 830-1998 entitled "IEEE Recommended Practice for Software Requirements Specifications" have been used.

This introduction is in section 3 followed by a state of the art section that shortly discusses existing PDM systems, followed by a more thorough analysis of how product information can be communicated and processed in a multi-organizational context. Section 4 analyses the PROMISE application scenarios and to what extent existing solutions are applicable to them. Section 5 provides a short analysis of the requirements not covered by existing systems, followed by an overview of how these requirements could be covered in section 6. Section 7 outlines how these requirements could be addressed in a future PROMISE PDKM. Section 8 compares how these solutions compare with existing systems and what are the main novelties in the outlined solution models. Section 9 lists unanswered questions in the current application scenario descriptions, followed by conclusions.

2.3 Definitions, acronyms and abbreviations

ARL: ABLE Rule Language

BOL: Beginning Of Life

CRM: Customer Relation management

Demonstrator:

Application scenario defined by end-user that should be implemented and demonstrated as a result of the project.

DfX: Design for X (where X read: safety, environment, reliability, comfort, cost, manufacturing, assembly, …)

ECU: Engine Control Unit

EEE: Electric and Electronic Equipment

EOL: End Of Life

IDEF0: Diagramming technique for activities modelling (see www.IDEF.com)

IMS: Intelligent Manufacturing Systems

IT: Information Technology

KM: Knowledge Management

LCC: Life Cycle Cost

MOL: Middle Of Life

NC: Numerical Control

OWL: Web Ontology Language

P2P: Peer-to-peer

PDKM: PROMISE Data and Knowledge Management (DoW, page 18). *This acronym will not be used in this document* in order to avoid confusion!

PDKM: Product Data and Knowledge Management (DoW, Glossary page 5, 7, 38 etc.)

PDM: Product Data Management

PDSS: PROMISE Decision Support System

PEID: Product Embedded Information Device

PLM: Product Lifecycle Management

PLMS: PLM System, of which PDSS is a part

RAM: Reliability, Availability, and Maintainability

RDF: Resource Description Framework

WP: Work Package

WS: Web Service

3 State of the art of how to represent, access and enrich product information

This section starts by the classification of product information mainly according to the concepts of static versus dynamic product information, i.e. typically design and manufacturing data versus field data collected during the use of a product and updated knowledge about how the products perform. After this introductory section 3.1, section 3.2 describes the state of the art on how to represent product information and product structures on the levels of tacit and explicit knowledge. Section 3.3 describes existing models for using and accessing product information when it is distributed over different organizations, computers or software. Section 3.4 describes how collected dynamic product information (e.g. field data) can be filtered or otherwise enriched, including the case when field data and other product information are located on different computers with intermittent communication possibilities. Finally, section 3.5 presents existing solutions of different PROMISE partners that address these issues.

3.1 Classification of product information

Different applications have their own characteristics in how product information is created, gathered, maintained, and provided to the ones needing it. In order to understand better the challenges of managing different pieces of information, classifications for product data and product data storage are presented. These classifications explain how the characteristics of information affect the information management practices.

The first classification is by Simon et al. (2001). There are generally two kinds of data that can be stored: static data and dynamic data. Static data means the specification of the product, i.e. data that is once created and stays intact during the product’s life cycle. In practice this kind of data is materials, components, and suppliers used, configuration and options, servicing instructions, and end-of-life information. Contrary to the static data, dynamic data comprises data that is collected during the use of a product. This includes patterns of use, environmental conditions, servicing actions, and part replacements.

Simon et al. present two solutions for storing of static data. The simplest way is to store data to the product itself. This is common for example in motor vehicles. The other possibility is to store data on a remote database and to have a unique identifier for the product. This approach can be used when data cannot be stored to the product for one reason or another. The approach to collect dynamic data Simon et al. present is using life cycle data acquisition (LCDA) technology. In principle this means a system that records automatically data about the use of a product. In addition to using LCDA technologies, dynamic data can, of course, be collected by manually writing down the relevant information.

Marsh and Finch (1998) introduce the concepts of centralized and localized (or decentralized) data storage. The localized solution involves storing data on the component itself. In this way data are available at the point where they are likely to be needed. Data storage can be carried out using number of technologies, including paper prints, barcodes, and RFID tags. The opposite of a localized solution is the centralized model, where data are located in one place. This means a central repository and a means to transmitting information are needed.

In the centralized solution, updating information and making extensive analysis is very easy since the data is located at only one place. But although the centralized solution seems more elegant, Marsh and Finch point out that there are many disadvantages related to it. To begin with, there might be problems with the means to transmitting information, e.g. poor signal strengths with radio-frequency transmissions. Secondly, the central database has to include vast amounts of data and may become unmanageable and inflexible for future changes.

In the localized model there is no need for transmitting information to the point of use. But in the localized system there is still a need to consolidate information to enable planning, scheduling and reporting. That is why Marsh and Finch remark that an entirely localized data storage solution is unlikely to be the answer for maintenance management purposes.

The two presented classification frameworks have some intersections. Simon et al. present two solutions for storing of static data, which can be considered to be a centralized and a localized solution according to the classification by Marsh and Finch. Dynamic data, determined by Simon et al., is created in a decentralized manner for individual product items and service calls. Therefore, it is convenient to manage dynamic data using a decentralized model. On the other hand, Mars and Finch point out that an entirely decentralized data storage solution is unlikely to be the answer for maintenance management purposes because there still is a need to consolidate information in a central system to enable planning, scheduling and reporting.

It should also be pointed out that a centralized solution does not necessarily mean that product information would located in only one place. As enterprises become increasingly global and networked (the “virtual enterprise”), product information tends to become spread on computer systems of multiple companies. This means that accessing product information may require communication between information systems of many different organizations.

3.2 Managing product-related knowledge and information

Corporate knowledge is nowadays well accepted as a decisive asset in most European enterprises. The know-how and expertise of the work-force is an important factor for the success of companies and strongly influences the effectiveness and efficiency of the business processes and their outcome (Thoben et al, 2000).

The concept of Knowledge Management (KM) receives high strategic attention across multiple sectors. In many domains such as the focused area, KM is specifically relevant due to the knowledge intensive character of the related domain such as product development and product related service provision.

According to this the PROMISE project aims to increase the degree of usage of existing knowledge in various kinds of processes, as given by the PROMISE end users in the description of application scenarios.

In order to achieve this objective, appropriate methodologies for the efficient management of product related knowledge have to be selected and if necessary further developed with regards to the particular requirements of the PROMISE application scenarios.

A first step in the sequence of actions for the selection, composition and/or development of KM concepts and methodologies is to agree on a common understanding with reference to the utilized KM framework and specific terms.

3.2.1 Knowledge Management Framework

A KM framework explains the world of KM by naming the major elements, their relationships and the principles of how these elements interact. It provides the reference for decision about the implementation and the application of KM (EKMF, 2002).

In a more abstract sense, a framework is a set of ordered representatives of cooperating objects and their relationships that provide an integrated solution within an application domain. It is directed towards explanation of a domain and making its behaviour understandable and predictable. In contrast to a theory it leaves certain space for interpretation, and in contrast to a method it does not describe complete steps yet, but only gives indications about a direction and a normative message. For practical usage it requires an instantiation.

In a pragmatically sense the group of partners involved in the work of WP R7 were intending to apply an existing KM framework build up on a common consensus with a majority approval aiming at a common agreement within the group about ‘how things shall be named” and how further activities can be structured.

The European KM Forum (EKMF) defines frameworks as a holistic and concise description of the major element, concepts and principles of a domain. It aims to explain a domain and define a standardized schema of its core content as a reference for future design implementations.

Thus, the EKMF framework is not particularly suited to the aims of the PROMISE project, since it includes a lot of issues which are neither addressed in the project objectives nor by the focused domain of product specific information management. Anyhow, the utilization of the framework seems to be useful to underline the particular domain of interest and its interrelations to other KM domains (modules) within the overall KM world.

The EKMF KM framework is based on the contribution of over 60 domain experts and evaluated on a number of KM specific events. Even though it is a draft version the PROMISE consortium decided to take this as an origin for its further work in particularly because of its brought recognition in the European KM community.

The following figure shows the draft version 1.4 of the EKMF KM framework:

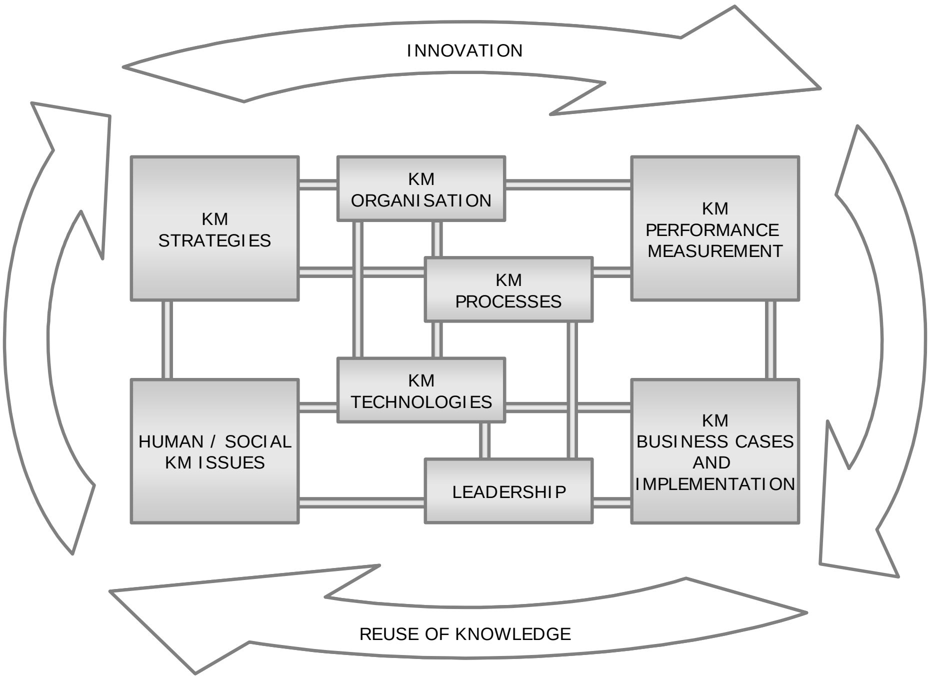


Figure 2: EKMF Knowledge Management Framework

The first draft of a KM framework developed by the European KM Forum consists of seven major modules: KM strategies, Human + Social KM issues, KM organizational aspects, KM processes, KM technologies, KM performance measurement and KM business cases + implementation aspects. These seven modules are closely linked together to support on the one hand side the innovativeness of the whole system, on the other side to secure the aspect of reusing existing knowledge within the system.

With regards to one of the main objectives of the workpackage which is to foster basic knowledge management processes such as knowledge identification, acquisition, preservation, distribution or assessment within the scope of application scenarios it is obvious that the KM processes module is of major interest for further investigations. Nevertheless, due to the fact that each of the framework modules has an interrelation with the process module it is consequently also required to review and assess the relevance of these modules with regards to their relevance for the PROMISE application scenarios.

Specifications of the modules and the assessment of the relevance will be described in the following paragraphs.

3.2.1.1 KM processes

This module deals with the relevance and importance for KM of business and general processes manly in an organisation or an organisational network. It provides guidelines for the whole target group to become more efficient in identifying, acquiring, sharing and maintaining knowledge. A knowledge-oriented organization must understand the internal processes in their organization that support the creation, acquisition, sharing and storing of knowledge. Technology only plays a small role in these activities. The more important issues here are identifying and managing the types of knowledge that need to be acquired and shared and creating a culture that encourages and promotes knowledge sharing. This includes the creation of appropriate systems to enable the efficient and effective use of knowledge in the organization and providing different user groups with access to the information they need at the right time and in the right place.

There are existing a vast number of models for the KM process module, which are dividing the overall process of knowledge management into different phases. Each of these models is focusing on a more or less generic application area.

One of the more comprehensive and generic approaches is provided by Probst/Raub/Romhardt (Probst et al, 1997) with their modules or Building Blocks of KM. Probst differentiates in so called Building Blocks of KM between eight KM processes as shown in the figure below.

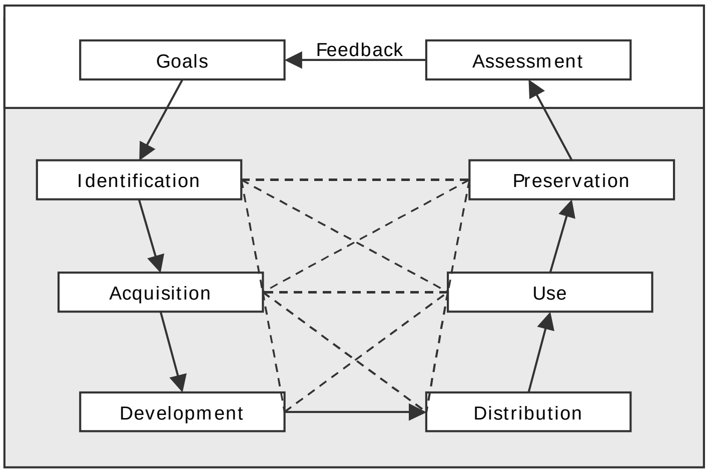


Figure 3: Building Blocks of KM (Probst et al, 1997)

Knowledge Goals

Knowledge goals are directing the activities of knowledge management. They determine on which level which skills should be built up.

- Normative goals are aimed at creating a knowledge-conscious corporate identity.
- Strategic goals define the organisational "core-knowledge" and therefore describe the future requirements of competency of a company.
- Operative goals ensure the implementation of knowledge management

Knowledge identification

Provision of transparency on missing and existing, internal or external knowledge by using knowledge maps, informal and external networks, databases, etc.

Knowledge acquisition

Due to the worldwide knowledge explosion and knowledge fragmentation organisations are decreasingly capable to develop the necessary know-how on its own. Critical skills have to be acquired at various knowledge markets. This implies a calculated procurement strategy. Sources are other companies, stakeholders, external knowledge owners, knowledge products.

Knowledge development

The focus is on the production of new skills, new products, better ideas and efficient processes.

- Processes of individual knowledge development are based on creativity. and skills for systematic problem solving. Creativity can be seen as the chaotic component of the process and problem solving competency as systematic component of the knowledge development process.

- Processes of collective knowledge development are based on collective learning. An atmosphere of openness and trust is necessary. They could be supported by think tanks, learn arenas, design of internal competence centre or product hospitals. Output will be lessons learned.

Knowledge sharing and dissemination

Knowledge sharing and dissemination is mandatory condition in order to provide isolated information or experiences to the whole organisation. Fostering simultaneous knowledge exchange trough IT among many employees. Storing and sharing experiences of the past.

Knowledge Utilisation

Knowledge utilisation meaning the productive effort of organisational knowledge for the organisation's sake is the core purpose of knowledge management. Utilisation of foreign knowledge is restricted by certain barriers. Using foreign skills and knowledge is for many people "unnatural". Sticking on "well-proven" routines is a security mechanism which protects the individual from foreign control and keeps its individuality alive. All efforts are in vain if the potential user is not convinced of the benefit of the new solution.

Knowledge preservation

Once required knowledge is not automatically available for the future. Collective amnesia can often be put down to unconscious demolition of informal networks which control scarcely noticed but important processes. In order to not to carelessly loose valuable expertise the processes of Selection, Storage and Up-dating have to be designed deliberately.

Knowledge evaluation

Methods for assessing the normative, strategic and operative goals are necessary. The quality of the goal setting is proven during the assessment at least.

Brian Newman and Kurt W. Conrad (Newman and Conrad, 1999) provided the approach named the General Knowledge Model. In comparison to the other models this one has the following characteristics relevant for the work of this task:

- It covers the overall way of knowledge flow which means it is not restricted for example to either business processes or human aspects
- Consequently it is also applicable for both tacit as well as explicit knowledge
- It allows an asynchronous creation and application of knowledge

The General Knowledge Model (see Figure 4) sequences the activity areas in a deterministic fashion. In reality, though, all but the most rigorously automated knowledge flows comprise complex systems that are built mostly from asynchronous processes.

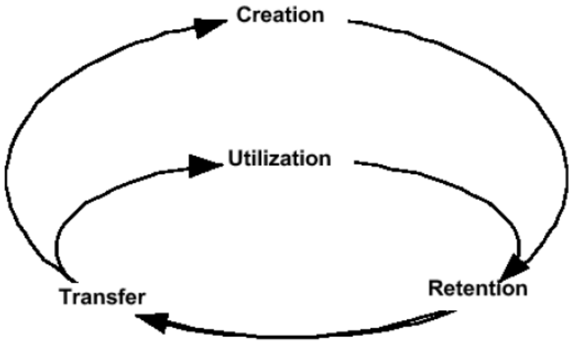


Figure 4: The General Knowledge Model

Within each activity phase exists other, smaller knowledge flows and cycles. These layers span a wide range of macro- and micro-behaviours, ranging from broad organisational and multi-organisational processes to discrete actions and decisions, and include all the various intervening layers: activities, tasks, workflows, systems, interfaces and transformations

The General Knowledge Model is valuable precisely because it relates the individual, highly dynamic behaviours and processes to general activity areas and, by association, to each other. Various theories of learning, problem solving and cognition may imply specific activity patterns, but they are usually not required to organize the key relationships and dependencies among the activity areas. The model allows analysts to trace individual knowledge flows by helping them to examine and understand how knowledge enables specific actions and decisions.

The model consists of the following activity areas:

- **Knowledge Creation:** This comprises activities associated with the entry of new knowledge into the system, and includes knowledge development, discovery and capture.
- **Knowledge Retention:** This includes all activities that preserve knowledge and allow it to remain in the system once introduced. It also includes those activities that maintain the viability of knowledge within the system.
- **Knowledge Transfer:** This refers to activities associated with the flow of knowledge from one party to another. This includes communication, translation, conversion, filtering and rendering.
- **Knowledge utilization:** This includes the activities and events connected with the application of knowledge to business processes.

Although dealing with similar process categories Weggeman (Weggemann, 1998) explicitly underlines the importance of the knowledge identification process for the overall organizational KM in his Knowledge Value Chain model (see Figure 5). He argued that to be able to fulfil the strategy and mission of an organisation, it is important to determine what kind of knowledge is needed, to make an inventory of the available knowledge in the organization, to development or acquire the lacking knowledge, and to share, apply and evaluate the knowledge.

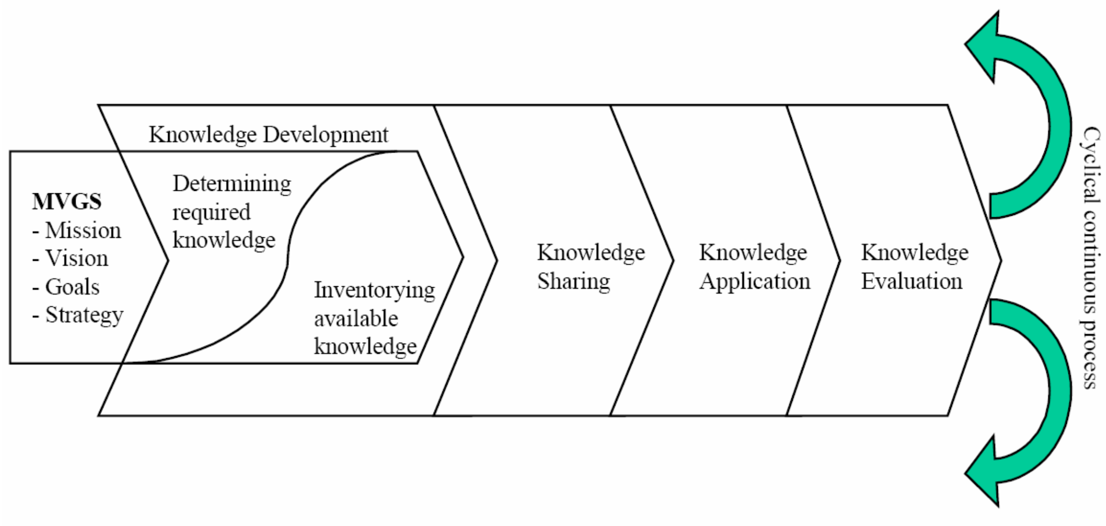


Figure 5: Knowledge value chain (Weggemann, 1998)

The authors Ikujiro Nonaka and Hirotaka Takeuchi (Nonaka and Takeuchi, 1995) are providing a model which is focusing on the aspects two major aspects. One is that this model is mainly dealing with the process of knowledge creation. The process of knowledge creation in an organization often entirely depends on knowledge which is bounded to humans which is from the organizational standpoint somehow elusive knowledge or so called tacit knowledge. Thus, the second major aspect of the model is the issue of tacit knowledge and how it is embedded in the overall knowledge processes

Nonaka and Takeuchi are arguing that knowledge creation depends on tapping into the tacit and often highly subjective insights, intuitions and ideals of employees. The key to this process is personal commitment and the employees' sense of identity with respect to the enterprise and its mission.

The widely discussed model provided by Nonaka and Takeuchi is named the SECI process or Knowledge Spiral. They see the mobilization and conversion of tacit knowledge as the key to knowledge creation. **Tacit knowledge** is highly personal and hard to formalize, it is not easily visible and expressible and consequently difficult to communicate or to share with others. It includes subjective insights, intuitions and hunches. Tacit knowledge is deeply rooted in an individual's action and experience, as well as in ideals, values and emotions. **Explicit knowledge** is formal and systematic. It can be expressed in words and numbers and easily communicated and shared as hard data, scientific formulae, codified procedure or universal principles. The interaction between tacit and explicit knowledge from a lower ontological level to higher levels (individual, group, organizational, inter-organizational) emerges as a spiral (knowledge spiral) that includes four layers of knowledge conversion: **socialization** (from tacit to tacit), **externalization** (from tacit to explicit), **combination** (from explicit to explicit), and **internalization** (from explicit to tacit), or SECI for short (see Figure 6).

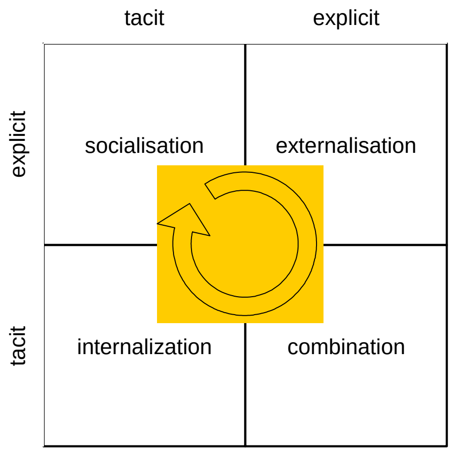


Figure 6: Four steps of knowledge conversion

- **Socialization** is the process of sharing experiences and thereby creating tacit knowledge (e.g. apprentice master relationship, where craftsmanship is learned not through language, but through observation, imitation and practice). The key to acquiring tacit knowledge is shared experience.
- **Externalization** is a process of articulating tacit knowledge into explicit concepts (i.e. documenting knowledge). Making tacit knowledge explicit is a key criterion in the knowledge creation process. Externalization runs into problems when specific skills cannot be documented or processes are purposely based on old traditions and methods.
- **Combination** is a process of systemizing concepts into a knowledge system. New knowledge can be created by combining different forms of explicit knowledge (e.g. files, graphics, databases, paper documents, meetings, telephone conversations, etc.) and reconfiguring existing information through sorting, adding, combining and categorizing.
- **Internalization** is a process of embodying explicit knowledge into tacit knowledge. It is closely related to ”learning by doing”. When socialized, externalized and combined knowledge is internalized into employees tacit knowledge bases (e.g. as shared mental models or technical know how), it becomes a valuable asset.

Due to the significant relevance of tacit knowledge for KM processes in common and it’s particular relevance for the PROMISE objectives the consortium selected this model for structuring parts of the further work of this workpackage.

3.2.1.2 KM strategies

Effective KM needs a clearly defined and well-communicated strategy, goals and direction. These should include:

- **normative** goals (aimed at the creation of organizational knowledge awareness, environment and culture open to effective KM and the sharing and development of individual skills and knowledge),
- **strategic** goals (to provide a definition of organizational core knowledge and a description of the future needs and competencies required by an organization), and
- **operative** goals (to enable the implementation of KM and success of KM initiatives and ensure that normative and strategic goals become definite and can be achieved).

Knowledge goals provide the direction for KM activities and determine which skills need to be developed and on which level. Strategy and goals must be clearly communicated to all concerned. Determining and defining a knowledge strategy requires an understanding of the business needs. The implementation of new technologies is expensive and time-consuming and a bad choice could affect productivity and often block the way for a later, “better” application. Strategy and goal definition are generally top down activities, reaffirming the need for management support in KM activities. Furthermore KM goals should be ultimately closely linked to corporate business goals. However, KM initiatives often grow out of a bottom up approach, with an individual or department realizing the benefits of KM and acting as a catalyst for further KM activities throughout the organization.

The relevance of this module in the context of the PROMISE project is considered as rather small due to the fact that it deals with the implementation issues and more specific with measures supporting the control of KM systems. Contrary to this it is the focus of this workpackage to select and/or develop methodologies supporting the operational processes given in the application scenarios defined by the end user of the project. Thus, KM strategies will be taken under consideration during the process of selection and development of KM methodologies, but methodologies proposed in the strategy module will neither be applied in the scenarios directly nor supported by the software utilized or to be developed by the project.

3.2.1.3 Human + Social KM issues

This module defines the roles of the people involved in the KM process and should result in a clear definition of the specific human-oriented KM issues. The inclusion of this module (and also the Leadership module) is based on the recognition that much work has been done in the field of KM and that an emphasis now needs to be placed also on non-IT is-sues, since it is now widely recognized that knowledge resides in people and resulting from this major barriers are consequential of individual or organizational structures. KM is no longer seen simply as the managing of knowledge, but rather as the introduction of organizational and management methods to ensure knowledge transfer and its contribution to the value creating process. As a result, KM initiatives must now actively look to include human resources activities and methods for personal and organizational development. These include recruitment activities, training measures, mentoring systems, support for formal and informal communities and networks, space management and motivation schemes. The reasons and motivations for creating and sharing knowledge differ from person to person. KM must create a climate of trust for knowledge sharing by addressing fears and barriers and, if appropriate, introducing incentives or rewards for knowledge sharing.

The relevance of the human and social KM issues is considered as potentially high in particularly for the management of so called tacit or implicit knowledge which is a major part of this workpackage, but methodologies or concepts applicable for the management of particular human or social KM issues shall not be considered in the development of the overall PROMISE work.

But although methodologies and concepts covered by this modules will not be taken into account for the further work in the project concepts and formal descriptions of criteria for the selection of KM process supporting methods can be derived.

Selecting adequate methodologies for the acquisition of tacit knowledge for example requires the knowledge on preconditions required for the application of these methods also from the human and social standpoint.

3.2.1.4 KM organisation

This module provides important information on how to create, run and maintain a knowledge friendly organisation. This will include the suggested structure and the roles within a 'KM organisation'. This module should be seen as providing guidelines for aligning existing organisational structures to KM.

Whilst recommendations can be made, it is impossible to define a generic "optimal organisational structure" valid for all organizations, since their needs and requirements are and will all be different.

Organisational culture is extremely important for collaborative working and KM is more likely to succeed if it fits in with the existing corporate culture. Organisational culture is best described as the way things work in an organisation. No two organisational cultures will ever be the same. Unfortunately, KM will rarely (if ever) encounter an organizational culture that is totally open to KM and the real problem starts in ensuring that the necessary cultural changes are made to build an organisation geared to knowledge sharing and exchange.

Another relevant issue in the context of PROMISE is the definition of the organisation itself. The different application scenarios defined by the end user of the project are mostly located in an inter-organisational environment.

This requires on the one hand methodologies supporting the integration of organisations such as supply chain management methods for the KM area but also methodologies to support the management of knowledge held by outside stakeholders such as customers, suppliers, competitors, the regional community, or the government, through social interactions. Tapping these sources is a key activity in the knowledge-creation process.

The overall relevance of this module has to be considered as high for the PROMISE application scenarios. Issues of particularly interest are those which are addressing the application of KM methodologies in the inter-organisational context.

But again, it is rather the formal description of preconditions for the selection of methods which can be derived from this module than methodologies for the development of organisational structures.

3.2.1.5 KM technologies

What technology for what purpose? This fundamental question will be answered with the KM framework module 'KM technologies'. It gives an overall overview over existing and future technologies towards KM and will be helpful for organizations to take the right decision in this 'hard' issue of KM. A more detailed view on the technology issue with respect to the requirements of the PROMISE application scenarios is given in the deliverables DR7.1, DR 7.2 and DR 7.3 of this workpackage where the first two ones are focusing more on the methodology behind particular technologies and the latter on is focusing on the related technology in particular.

3.2.1.6 Leadership

This module looks at the critical success factors in introducing a KM leader within an organization and includes the desirable or assumed characteristics of such a leader, their situation/position in the organizational hierarchy and their activities. Some form of leadership will be necessary for the success of any KM initiative, be it a top-down or bottom/middle-up approach. If nobody takes the lead and promotes the system and its benefits, it will be unlikely to succeed. Furthermore, in the case of IT applications, someone will have to assume a leadership role in system implementation, content and administration. The same applies to communities of practice and formal/informal networks, which form around a key individual or group of players. The use of a pilot project is a good way of gaining support for a system and the members of the pilot team often then take on the role of “KM leaders” in their organization.

Basic issues of this module such as IT implementation leadership and leadership of the KM initiative are not addressed within the scope of the PROMISE project. Leadership is neither required for the implementation of the application scenario nor the demonstrator scenarios.

3.2.1.7 KM performance measurement

This module looks at the importance of measuring the performance of a KM system and the resultant improvements from these measuring methods. It also provides metrics to provide an overview of the maturity of a KM system. In addition, it will include measures for expanding and extending KM activities. Since one of the main reasons for introducing KM is to increase corporate value, organizations need to also introduce methods of measuring the impact and success of KM measures and of normative, strategic and operative knowledge goals. KM measures take up resources and time and must therefore prove their worth. Traditional accounting methods are not suitable for measuring knowledge and there is increasing evidence that they do not capture the full value of an organization. However, there are currently no real tried-and-tested measurement methods available. One of the challenges of effective KM is the establishment of appropriate ways of measuring whether or not goals are being met. One way to do so is to introduce definite, measurable goals for KM that will show the value of the initiative.

PROMISE combines data and information management methodologies with existing KM approaches in a generic way. The application of this mixture of methodologies will be done in the application scenarios. Basically these scenarios are derived from existing real world scenarios. A validation of each scenario as a whole will consequently be realized against the existing real word scenario. Single aspects such as implemented KM methodologies will if necessary also be validated.

3.2.1.8 KM business cases + implementation

This module will provide good and best practices in different areas of KM as well as suggesting a general roadmap to help organizations in implementing and establishing their KM systems. It is obviously advantageous to look at what other organizations are doing, especially partners and competitors, when beginning KM initiatives. There are now a large number of “best practices” stories published in literature and even on the Internet to be looked at and often partner organizations are willing to share their experiences. Although KM projects often simply “grow” over the years, if an organization is looking to implement KM from scratch, then a similar planning approach should be taken to implementation as with other business projects. This is extremely important if the KM approach selected has a strong IT focus, since the introduction of IT tools generally requires a detailed analysis and planning phase prior to installation. However, it is not enough to simply extend project management methodologies to include KM. The reason for this is that KM links both technology and culture and requires KM-specific competencies.

Due to the very specific and detailed definition of application scenarios and demonstrators it would be helpful to apply methodologies such as “best practices” to support the selection of KM methodologies if available.

3.2.2 Overview on related KM methodologies and technologies

This chapter will provide an overview of KM practice, methodologies and technologies based on the structure given by the KM model provided by Nonaka and Takeuchi which is described above. Consequently main criteria for the categorization of methodologies and technologies are the different kinds of knowledge conversations which are related to the four layers of knowledge conversion:

- **socialization:** from tacit to tacit (S),
- **externalization:** from tacit to explicit (E),
- **combination:** from explicit to explicit (C), and
- **internalization:** from explicit to tacit (I)

Additionally to this information will be provided whether the particular KM concepts is be rather driven by a methodology (M) or a technology (T).

Table 2: Categorization of KM concepts

Name	Type		Type of conversion				Description
	M	T	S	E	C	I	
After Action Review (AAR)	x			x	x		A systematic process to extract the learning from an event or activity. The process addresses the questions: What should have happened? What actually happened? What lessons are there for the future?
Answernet	x				x		A service provided by a network of experts who answer questions posed online.
Artificial Intelligence (AI)		x	x		x		A set of computer techniques that make the computer appear to behave with a degree of human intelligence. Rather than the procedural way of programming, it draws on inferences and rules to guide its actions. Expert systems, intelligent agents and natural language search are examples of the use of AI techniques in knowledge management.
Balanced Scorecard	x			x	x		A performance measurement system that incorporates a balanced set of measures, both financial and non-financial. It adds customer, internal processes and innovation and learning indicators to financial ones to provide a more balanced view. Contrast with the more specific intellectual capital measurement methods.
Benchlearning	x			x	x		A structured approach whose focus is on learning from others to create distinctive improvements. Developed by Bengt Karlof and colleagues, it overcomes the often narrow focus of benchmarking on quantitative comparisons, which downplays the key role of knowledge transfer.
Benchmarking	x			x	x		A systematic process for comparing the performance of an activity or process across a range of organizations or departments. Identifying gaps in performance leads to on to *benchlearning* and learning *good practice* from high performers.

Benefits Tree	x				x	x	A diagrammatic depiction of cause-effect relationships from knowledge processes to business outcomes. Helpful in making the business case for knowledge management.
Best Practice	x			x	x		The distillation of accumulated wisdom about the most effective way to carry out a business activity or process. Since 'best' is highly subjective and context dependent, as well as implying that no further improvements are possible, many people now prefer the term good practice.
Blog (originally Web log)		x		x			A string of thoughts of an individual shown in chronological sequence on a Web page, often with hyperlinks to sources that have stimulated his or her thinking. Although often dismissed as a gimmick some people see Blogging as grass-roots KM, alongside *storytelling*. Others suggest that it perpetuates knowledge silos and that a *Wiki* is more appropriate. See also *K-log* and *weBlog*
Brainstorming	x		x	x			A creativity and problem-solving technique that involves the spontaneous generation of as wide a spectrum of ideas as possible. It is an organized approach for producing ideas. It can be done either individually or in a group.
Case Based Reasoning (CBR)		x			x		An application of AI techniques, where solutions to a given problem are sought through a reasoning process that draws analogies with similar problems whose solution is already known.
Caves and Commons	x		x				Denotes two main types of physical working area: a cave is a private area for concentrated thinking; commons are open areas for socialization and meeting rooms for team discussions. Design of working space can significantly enhance the productivity of knowledge workers.
Chatting	x		x	x			Any system that allows any number of logged-in users to have a typed, real-time, on-line conversation, either by all users logging into the same computer, or more commonly nowadays, via a network. Some well known chat systems to date include IRC and ICQ.
Classification	x			x	x		A key process in the knowledge sharing cycle. Documents are classified and indexed according to their core terms and concepts. Increasingly computer systems provide a level of automation of this process, using natural language or statistical methods.
Collaboration Software		x		x		x	Software that promotes immediate communication between individuals. Features of collaboration software include instant text messages, chat sessions, and shared documents. Portal Server includes portlets that provide access to Sametime, an instant-messaging servers, and QuickPlace, a shared team room application. See *groupware*
Community	x		x				A community of *interest* or *practice*. The focus of a community is usually part of a website that typically provides message boards and other conversational facilities (such as *discussion lists* and *instant messaging* as well as a library of online resources. Some people also refer to communities of purpose or communities of commitment.
Community of Interest (CoI)	x		x				A group of people who share knowledge and experience around a common interest. Driven more by learning and less on outcomes than a Community of Practice.

Community of Practice (CoP)	x		x				A group of people who share and develop their knowledge in pursuit of a common purpose or task, even though they do not necessarily work in the same department or organization. John Seely Brown of Xerox calls them "peers in the execution of real work".
Concept Mapping	x					x	A visual representation of core concepts showing the relationships between them. A typical concept map comprises a set of nodes or bubbles (the concepts) with arrowed links between them (the causal relationships). One of the several types of knowledge mapping.
Conferences	x		x	x		x	A meeting for consultation, discussion, or an interchange of opinions. Conferences are usually pre-arranged and often have a formal agenda.
Content Analysis	x					x	Analysis of a body of content (text) into its key concepts. As well as a method of discerning trends, this technique is used to generate keywords and thesaurus terms to improve subsequent text search and retrieval. The latter result is increasingly achieved through the use of automated classification systems.
Content Management System (CMS)		x			x		A computer system supporting the management of content separated from its presentation. Blocks of content are tagged with metadata and other attributes and held in a content database. Web pages are generated (often 'on-the-fly') by accessing content from the database and inserting it into the relevant 'placeholders' on Web page templates. Since a single block of content may appear on many Web pages, the task of maintenance and updating is simplified.
Customer Relationship Management (CRM)	x		x	x			An approach that gathers and uses knowledge of customers' buying habits and preferences in order to strengthen the ongoing relationship for mutual benefit. Customer knowledge comes out as the most important knowledge to manage in many KM surveys.
Data Mining	x				x	x	A computer technique for extracting meaningful knowledge from masses of data. Using *artificial intelligence* methods it identifies unanticipated patterns by considering the interaction of many more variables than is achievable by humans. Contrast with *text mining*.
Data Visualization	x				x	x	Data represented by visually recognized patterns and trends. This includes 2D and 3D visualization, animation, images and charts.
Data Warehouse		x			x	x	A data warehouse is, primarily, a record of an enterprise's past transactional and operational information, stored in a database designed to favour efficient data analysis and reporting (especially Online Analytic). Data warehousing is not meant for current, "live" data. Data warehouses often hold large amounts of information which are sometimes subdivided into smaller logical units called dependent data marts.
Decision Diary	x			x		x	A diary in which decisions are recorded, together with the assumptions and reasoning behind them. They are used to derive lessons and record knowledge that will help future decision-making.
Desktop Conferencing		x	x				Videoconferencing using a desktop PC. A small camera (webcam) is usually mounted on top of the user's display screen. Evidence suggests that this often transfers expertise better than simply using email or documents.

Discussion List		x		x	x		A mechanism used by to share information and knowledge using a single email address to communicate to all members of a given list. Typically all messages generated during one day are grouped together and sent as a single email in a 'digest'.
Document Management System		x		x	x		A computer-based system for storing and retrieving documents held in a variety of formats, including scanned images of paper documents. Many provide version control and audit trails of changes and usage. The distinctions between document management, content management and records management systems are increasingly blurring.
E-Learning		x				x	E-learning is an approach to facilitate and enhance learning through the use of devices based on both computer and communications technology. Such devices can include personal computers, CDROMs, Digital Television, P.D.A.s and Mobile Phones. Communications technology enables the use of the Internet, email, discussion forums, and collaborative software.
Enterprise Information Portal (EIP)		x			x	x	Strictly, an entry point (home page) into an organization's intranet, although the term now often refers to the intranet itself and its content. Users have a personalized starting page that gives them a single point of access to enterprise information, wherever it is held.
Expert System		x			x		A common class of AI computer system that applies the logic and domain knowledge it has acquired from a human 'expert'. A typical expert system has three main parts - a knowledge base (that contains the rules), an inference engine (that interprets the situation against the rules) and a human interface.
Expertise Directory		x			x		A database of personnel and their skills that allows users to search for people with specific skills or relevant project experience. Often referred to as 'Yellow Pages'.
Expertise Profiling	x			x			The identification and classification of personal knowledge and skills. This may be done through manual completion of data forms or by computer systems that infer people expertise according to what they write in emails and documents. The output of the process may be an expertise directory or a database that is used in automated question and answer systems.
Extranet		x			x		A portion of an organization's intranet that is opened up for external Internet access on a selective basis e.g. for customers to access specific areas following input of a password.
Frequently Asked Questions (FAQs)	x				x		A list of questions that are most frequently asked or are anticipated by website or intranet users, together with their answers. Information providers use this technique to minimize the number of recurring queries and calls. Some organizations use the term AAQs - actually asked questions - since many writers of FAQs anticipate what might be asked or what questions their content answers.
Groupware		x			x		Computer software tools that support collaborative working. Lotus Notes was the archetypal groupware software, but many groupware facilities are now provided on the Internet e.g. bulletin boards, discussion forums, instant messaging. The term is generally falling into disuse compared to 'collaboration software'.

Information Resources Management (IRM)	x				x		The techniques of managing information as an organizational resources. They include the identification of information, its classification and ways of valuing and exploiting it.
Instant Messaging	x			x		x	An Internet or intranet facility in which users type messages into a window that is simultaneously viewed by other participants in that chat room or area. While commonly associated with informal social groups, the tool is a useful adjunct for synchronous knowledge exchange in a corporate context, for example as a way of interaction during a 'webinar'.
Intangible Assets Monitor (IAM)	x			x			A method of *IC Measurement* developed by Karl Erik Sveiby for recording intangible assets. It divides intangible assets into three main categories - competencies, external structure and internal structure. Indicators are divided into four distinctive groups - growth, renewal, efficiency and stability.
Intelligent Agents		x			x		Intelligent agents or bots are software elements that work without the assistance of users by making choices. Choices are based on rules that software developers have identified and built into the software.
Interview	x		x	x			An interview is a conversation between two or more people where questions are asked to obtain information about the interviewee.
Intranet		x			x		An internal internet. In other words an internal computer network that runs the Internet protocol (TCP/IP). Most intranets have a computer 'gateway' to the wider (external) Internet and deploy a 'firewall' to prevent unauthorized access to a company's information. See also *Portal* and Extranet.
Just-in-time Knowledge	x				x		The concept of delivering knowledge to an individual just at the time that they need it to carry out a task. This overcomes the problem of information overload, where knowledge not immediately needed may be forgotten or ignored. Mechanisms that help are alerting systems linked to computerized procedures or what a knowledge worker is typing into their computer and natural language retrieval.
K-Log (Knowledge Log)		x		x			A Blog (weBlog) whose subject is knowledge.
Know-bot (Knowledge robot)		x			x		An intelligent agent that gathers or exchanges knowledge from other agents or computer systems.
Knowledge Audit	x			x	x		The systematic analysis of an organization's information and knowledge entities and their key attributes, such as ownership, usage and flows, mapped against user and organizational knowledge needs. The terms information audit, knowledge audit, knowledge inventory and knowledge mapping are often used synonymously.
Knowledge Base		x			x		A computer held database that record knowledge in an appropriate format for later extraction. It may take various forms depending on whether it supports an expert system or contains documents and textual information for human retrieval.
Knowledge Based System (KBS)		x			x		A computer system that draws on AI techniques or knowledge bases for its operation. Examples include expert systems and neural networks.

Knowledge Broker	x				x		An intermediary that connects knowledge seekers to knowledge providers. It may involve brokering a deal and retaining anonymity between buyer and seller until a suitable stage of negotiation. Some overlap with a knowledge analyst.
Knowledge Café	x		x				Informal meeting area for the exchange of knowledge. Cafe´s can be virtual meeting rooms as well as real ones.
Knowledge Centre	x				x		A central function for managing knowledge resources. Often developed around a corporate library, a typical knowledge centre will manage both physical and virtual resources - documents, databases, intranet content, expertise directories etc.
Knowledge Inventory		x			x		A list or database of knowledge entities - their sources, users and uses. It may be the output of a knowledge audit.
Knowledge Mapping	x			x	x		The process of identifying core knowledge and the relationship between knowledge elements. A map may be portrayed in many visual formats, such as a hierarchical tree or a node and link diagram. It is typically a task carried out as part of a *knowledge audit*. See also *social network mapping*.
Knowledge Market	x				x		A marketplace for the buying and selling of knowledge. Online knowledge markets are sometimes referred to as knowledge e-marketplaces. They commonly allow the posting of knowledge needs and knowledge offers, and may conduct sales by auction. See the Insight: Online Knowledge Markets.
Knowledge Narrative	x			x	x		The articulation of value of and organization's products and services to customers and how knowledge resources are used to achieve this value. It derives from the organization's vision and strategy and describes its KM ambitions. It often forms part of an IC report.
Knowledge Recipe	x				x		The transformation processes that uses existing knowledge assets as inputs and combines them in distinctive ways to create useful outputs and outcomes.
Knowledge Refining	x						The process of filtering, aggregating and summarizing knowledge drawn from a wide range of resources.
Knowledge Repository		x	x		x		A store of knowledge. While the term typically refers to explicit forms of knowledge, such as documents and databases, it can also refer to human-held knowledge.
Knowledge Wrapper	x				x		Information associated with a knowledge object that accurately describes the contents within. It holds metadata in a standard format and may hold encrypted digital rights information.
Learning Network	x		x		x		A network of individuals who share knowledge for the primary purpose of personal development and learning. A specific example of a Community of Interest.
Machine Learning		x			x		The ability of a machine to improve its performance based on previous results. ML is a subfield of artificial intelligence dealing with the computer supported derivation of domain models which are based on knowledge representation paradigms of Artificial Intelligence.

Message Board		x			x		An area on a website where messages can be exchanged and viewed by a workgroup or community. Sometimes referred to as a bulletin board. The conversational interaction via the Web is sometimes called Web conferencing. See also discussion lists.
Metadata	x				x		Data about data. A structured piece of data that describes the contents of a database record. One common metadata format is that of the Dublin core that defines metadata fields for bibliographic databases. See also knowledge wrapper.
Mind Mapping	x			x		x	A visual method of organizing ideas. In most mind mapping systems the ideas branch out from a central point. In turn, each branch can have additional branches or links to other mind maps. A specific form of concept mapping.
Natural Language Processing (NLP)		x			x		The ability of a computer application, such as a search engine to accept ordinary language input rather than highly specified instructions. It processes text through analysis of syntax and semantics.
Neural Networks		x	x				An artificial intelligence technique that mimics the operation of the human brain. It consists of a network of individual neurons that are triggered according to the intensity of various inputs and their relative 'weights'. It adjusts these weights according to the quality of the outcome for a given set of inputs. In other words, a neural network learns from experience.
News Aggregators (RSS-Feeds)		x			x		A news aggregator is a piece of software that periodically reads a set of news sources, in one of several XML-based formats (primarily RSS), finds the new bits, and displays them in reverse-chronological order on a single page.
Online Analytical Processing		x			x		An approach to quickly provide the answer to complex analytical queries. It is part of broader business intelligence category which also includes ETL, relational reporting and data mining. The typical applications of OLAP are in business reporting for sales, marketing, management reporting, BPM (business performance management), budgeting and forecasting, financial reporting and similar areas. See also Data Warehouse.
Online Community	x		x		x		A *community of interest* or practice that uses computer-based collaboration facilities (such as *message boards*, *discussion lists* and *chat*, to share knowledge.
Ontology		x			x		An extension to a taxonomy that adds specifications of relationships between entities plus a set of automatic inference rules and associated actions. Typical relationships include "instance of" and "made of".
Portal		x		x	x	x	The common term for Enterprise Information Portal. A portal is a single point of entry on the Web or an intranet to a wide range of information and knowledge resources and personal tools. c.f. common definition of "gateway".
Project History	x			x	x		The main activities and decisions taken during a project, recorded in a way that aids knowledge sharing and derives lessons for similar projects in the future.
Reading	x					x	Reading is the process of retrieving and comprehending some form of stored information or ideas.

Research	x		x	x		x	An active, diligent and systematic process of inquiry in order to discover, interpret or revise facts, events, behaviours, or theories, or to make practical applications with the help of such facts, laws or theories.
Resource Description Framework (RDF)	x				x		A framework developed by W3C for developing *metadata* standards for WWW resources. It brings together in one place metadata activities for resources such as site maps, content ratings, search engine data collection and digital library collections. The resource descriptions use *XML* as the interchange language.
Schema	x				x		A taxonomy (classification) of knowledge or information. Common terms are used to describe an organization's knowledge domains which are categorized into hierarchies and related terms. See also *Taxonomy* and *Thesaurus*.
Search Engine		x			x		A piece of software or a service that indexes pages from the Web and lists those that match or closely match a user's search terms. Results are ranked by relevance or other factors and include items from sources all over the Web. One of the growing problems is the 'hidden' Web, content that is not indexed because it is generated on the fly or held in databases. It is estimated that over four fifths of Internet content is now hidden.
Semantic Network	x					x	A method of representing structured knowledge. It consists of nodes and links, where the nodes are concepts or entities and the links represent relationships and associations among the concepts. An ontology can be viewed as domain knowledge represented in the form of a semantic network.
Semantic Web		x		x	x		The addition of semantic constructs (ontological elements) to World Wide Web resources to create semantic networks accessible via the Internet. The Semantic Web is seen by some as the next evolution of the World Wide Web (the 'intelligent' Web).
Speech recognition		x		x			Or Voice Recognition describes a group of special technologies for the identification of spoken words by a machine or software. The system may take the spoken language and translate it into written text.
Storytelling	x		x	x		x	The use of stories in the organizational context, as a way of sharing knowledge and helping the process of learning.
Study	x					x	To study means to acquire knowledge, often by memorization or reading.
Taxonomy	x			x	x		A system of classification. A typical taxonomy is a hierarchy of terms (nodes), where lower level terms are more specific instances of higher level ones. Taxonomies in which a term can appear in more than one branch are called 'poly-hierarchical'. Contrast with *Thesaurus* and *Ontology*.
Teaching	x		x			x	The activities of educating or instructing; activities that impart knowledge or skill
Text Mining		x			x		Extracting the essential concepts and meaning from large amounts of textual information. See also text summarizing.

Text summarizing		x			x		The result of text mining a single document and producing a summary which includes some of its key sentences. Typically, all the main concepts of a large document can be summarized in less than twenty per cent of its original size.
Thesaurus		x			x		A controlled vocabulary of terms for a corpus of information. An extension of a taxonomy that includes rules on vocabulary usage for document classification e.g. "preferred terms", "synonym of", "belongs to", "used for" etc.
Topic Map	x				x	x	An ISO standard (ISO 13250) for describing relationships of nodes in an ontology independent of its underlying resources. Associations and Occurrences are key constructs in the XTM (XML Topic Map) standard.
Videoconferencing		x	x				Communications over an electronic network using video. Systems range from desktop units on PCs (desktop conferencing) to dedicated systems that use cameras and monitors in a conference room setting.
Webinar (Web seminar)	x		x				A presentation delivered over the Web using videoconferencing.
White Board Software		x		x		x	Allows people to draw together on a virtual whiteboard over the internet. Each user connects to the whiteboard and they can see what other users are drawing in real-time on their computer screen.
Wiki		x			x		A collaboration tool that allows multiple authors to create and update Web pages. KMWiki is an example of a Wiki devoted to KM. Contrast with a *Blog* which is authored by an individual.
Yellow Pages	x				x		A colloquial term for an expertise directory, since entries are organized by category rather than by name.

3.2.3 Management of tacit knowledge

The breakthrough contribution of PROMISE, in the long term, is to allow information flow management to go beyond the customer, to close the product lifecycle information loops, and to enable the seamless e-Transformation of Product Lifecycle Information to Knowledge. Closing the information loop requires among others the co-operation or collaboration between companies, organisations and individual parties such as product owners or freelancers.

Thus, one major issue of the knowledge management between various parties is the co-operations between people who are acting under specific social, organisational, economic, strategic and legal conditions. Looking at inter-organisational co-operations from this perspective puts the individual employee and the knowledge which is bounded to them in the centre of the observations.

Nowadays there is a widespread agreement that considering knowledge requires the differentiation of knowledge which is already expressed or codified and knowledge which resides in the head of an individual. Along with this differentiation the concepts explicit and tacit knowledge where developed in the scientific community.

Explicit knowledge, as the first word in the term implies, is knowledge that has been articulated and, more often than not, captured in the form of text, tables, diagrams, product specifications and so on. In a well-known and frequently cited 1991 Harvard Business Review article titled "The Knowledge Creating Company," Ikujiro Nonaka refers to explicit knowledge as "formal and systematic" and offers product specifications, scientific formulas and computer programs as examples.

Tacit knowledge is a phrase apparently coined by Michael Polanyi in 1962 (Polany, 1962) with his book Personal Knowledge. He described tacit knowledge as a characteristic of experts, who often have no explicit theory of their work: they simply perform skilfully.

In the following tacit knowledge was commonly accepted in the scientific community as an important phenomenon. Nevertheless Gourlay (Gourlay, 2002) stated that there are existing important differences of opinion over many key aspects of tacit knowledge, such as the level at which it is manifested, how it is acquired, what its function is, and whether or not it can be made explicit.

Gourlay summarized that the concept of tacit knowledge has been used in the literature in at least six different ways as regards individuals.

1. someone can do something, but apparently cannot give an account
2. someone claims they feel something of which they cannot give an account, but it is not clear if subsequent events validate the claim
3. someone can do something, but not give an account at that moment, but can, if pressed, recall the explicit knowledge that was used tacitly when acting
4. knowledge existing prior to the situation in which it is effective, and due to innate (biological) characteristics
5. knowledge existing prior to the situation in which it is effective, and due to cultural factors
6. situations where A knows something that B does not, but where it could be argued A and B share the same practice

From his review of the literature, Gourlay identifies two issues associated with tacit knowledge. The first is whether tacit knowledge is an individual trait or a trait that can be shared by both individuals and groups, and the second is whether tacit knowledge can be made explicit.

For the following work tacit knowledge has been defined as knowledge which has not been made explicit. Although there are more exhaustive concepts in the literature existent this definition is explicitly not differentiating between knowledge which can be articulated and the one which can be not.

Methodologies and tools adequate for the management of knowledge are listed, described and categorized in Table 2 with respect to the conversion levels proposed by Nonaka and Takeuchi. Three of the four conversation layers, namely **socialization** (from tacit to tacit), **externalization** (from tacit to explicit), and **internalization** (from explicit to tacit) are dealing with the subject of tacit knowledge.

With respect to the particularly conditions and requirements of the application area some of the most relevant concepts will be described in the following.

3.2.3.1 Instant Messaging

While communication via e-mail is one of the most popular ways to conduct a conversation with a remote user, it does not allow a real-time communication. Online chatting supports the real-time conversation between two or more people.

Today the most popular means of chatting online are instant messaging applications. Most of these support email communication and deferred delivery of messages. Some also support voice and video chat. Besides being used for Consumer 2 Consumer (C2C) interactions more and more examples appear were it is used for Business 2 Consumer (B2C) interaction. In this case chat substitutes the use of phone for contacting a call center and email.

More and more Web retail companies have started providing real-time customer help lines. Software producers like Microsoft use online chats to collect user feedback about their products, and even politicians use this possibility to get in touch with the public.

Sources:

- http://www.vrd.org/conferences/VRD2000/proceedings/Yue11-20.shtml
- http://en.wikipedia.org/wiki/Chat

Tools:

- **IRCXpro /** http://www.ircxpro.com/
- **ICQ** / http://www.icq.com

3.2.3.2 Customer Relationship Management (CRM)

The generally accepted purpose of Customer Relationship Management (CRM) is to enable organizations to better serve their customers through the introduction of reliable processes and procedures for interacting with those customers.

In today's competitive business environment, a successful CRM strategy cannot be implemented by only installing and integrating a software package designed to support CRM processes. A holistic approach to CRM is vital for an effective and efficient CRM policy. This approach includes training of employees, a modification of business processes based on customers' needs and an adoption of relevant IT-systems (including soft- and maybe hardware) and/or usage of IT-Services that enable the organization or company to follow its CRM strategy. CRM-Services can even redundantize the acquisition of additional hardware or CRM software-licences.

The term CRM is used to describe either the software or the whole business strategy oriented on customer needs. The second one is the description which is correct. The main misconception of CRM is that it is only software, instead of whole business strategy.

Major areas of CRM focus on service automated processes, personal information gathering and processing, and self-service. It attempts to integrate and automate the various customer serving processes within a company.

In this day and age the use of internet sites and specifically e-mail, in particular, are touted as less expensive communication methods, compared to traditional methods like telephone calls. This revolutionary type of service can be very helpful, but it is completely useless if you are having trouble reaching your customers. It has been determined by some major companies that the majority of clients trust other means of communication, like telephone, more than they trust e-mail. Clients, however, are not the ones to blame because it is often the manner of connecting with consumers on a personal level making them feel as though they are cherished as customers. It is up to the companies to focus on reaching every customer and developing a relationship.

CRM software can run an entire business. From prospect and client contact tools to billing history and bulk email management. The CRM system allows it to maintain all customer records in one centralized location that is accessible to the entire organization through password administration. Front office systems are set up to collect data from the customers for processing into the data warehouse. The data warehouse is a back office system used to fulfill and support customer orders. All customer information is stored in the data warehouse. Back office CRM makes it possible for a company to follow sales, orders, and cancellations. Special regressions of this data can be very beneficial for the marketing division of a firm.

Sources:

- http://en.wikipedia.org/wiki/Customer_relationship_management

Tools:

- **Salesforce** / http://www.salesforce.com
- **Wice** / http://www.wice.de/
- **SugarCRM** / http://www.sugarcrm.com/crm/

3.2.3.3 Communities

A virtual community is a group whose members are connected by means of information technologies, typically the Internet.

Today, "virtual community" is loosely used and interpreted to indicate a variety of social groups connected in some ways by the Internet. It does not necessarily mean that there is a strong bond among the members.

A Community of Interest is a group of people connected by a common interest in a specific subject or endeavour. The level of interest may range from passing to intense, and over time develop into expertise on a subject.

The concept of a Community of Practice (often abbreviated as CoP) refers to the process of social learning that occurs when people who have a common interest in some subject or problem collaborate over an extended period to share ideas, find solutions, and build innovations.

Sources:

- http://en.wikipedia.org/wiki/Virtual_community
- http://www.navfac.navy.mil/cheng/coi.htm
- http://en.wikipedia.org/wiki/Community_of_practice

Tools:

- **BAC Instant Community** / http://buildacommunity.com/
- **worldwebCommunity** / http://www.worldweb.de

3.2.3.4 Videoconferencing

A videoconference (also known as a *video teleconference*) is a meeting among persons where both telephony and closed circuit television technologies are utilized simultaneously. Video teleconference communication is multi-way and synchronous, as it would be if all parties were in the same room.

Video conferencing can be used as an imperfect substitute for many face-to-face meeting requirements where the cost or time required for physical transportation of parties is undesirable. For instance, interviews with job applicants, business meetings, and lectures can be conducted effectively via video conference.

For business purposes, the current preference is for the use of private or virtual private network connections for video teleconferencing which guarantees better performance and transmission security. Local area network infrastructure provides for high-bandwidth communication suitable to the application of video conferencing technology.

Video Conferencing provides students with the opportunity to learn by participating in a 2-way communication platform. Video conferencing can provide students with information from all over the world. Students from diverse communities and backgrounds can come together to learn about one another. Students are able to explore, communicate, analyze and share information and ideas with one another. Through video conferencing students can visit another part of the world to speak with others, visit a zoo, a museum and so much more. Video Conferencing can enhance any classroom environment and motivate students to learn.

Sources:

- http://en.wikipedia.org/wiki/Video_conferencing

Tools:

- **Webex Meeting Center** / http://www.webex.com/
- **Virtual Rooms** / http://www.vrvs.org/
- **NetMeeting** / http://www.microsoft.com/windows/netmeeting/
- **SeeNx** / http://www.seenx.com/
- **iVisit** / http://www.ivisit.com/

3.2.3.5 Collaboration Software

Collaborative software, also known as groupware, is application software that integrates work on a single project by several concurrent users at separated workstations. In its modern form, it was pioneered by Lotus Software with the popular Lotus Notes application running in connection with a Lotus Domino server; some historians argue that groupware was anticipated by earlier monolithic systems like NLS.

Groupware can be divided into three categories depending on the level of collaboration:

- communication tools
- conferencing tools
- collaborative management tools

Collaborative software can be either web, or desktop systems.

Sources:

- http://www.groove.net/
- http://en.wikipedia.org/wiki/Collaborative_software

Tools:

- **Groove Virtual Office** / http://www.groove.net/
- **WorkZone** / http://www.trichys.com/collaboration/collaboration-software.vm
- **BSCW** / http://bscw.fit.fraunhofer.de/

3.2.3.6 Mind Map

The mind map has been used for centuries, for learning, brainstorming, memory, visual thinking, and problem solving by educators, engineers, psychologists and people in general. Some of the earliest examples of the mind map were developed by Porphyry, a noted thinker of the 3rd century as he graphically visualized the concept categories of Aristotle. Ramon Lull also used these structures of the mind map form. More recently the semantic network was developed as a theory to understand human learning, and developed into mind maps by the renaissance man Dr Allan Collins, and the noted researcher M. Ross Quillian during the early 1960s. As such, due to his commitment and published research, and his work with learning, creativity, and graphical thinking, Dr Allan Collins can be considered the father of the modern mind map.

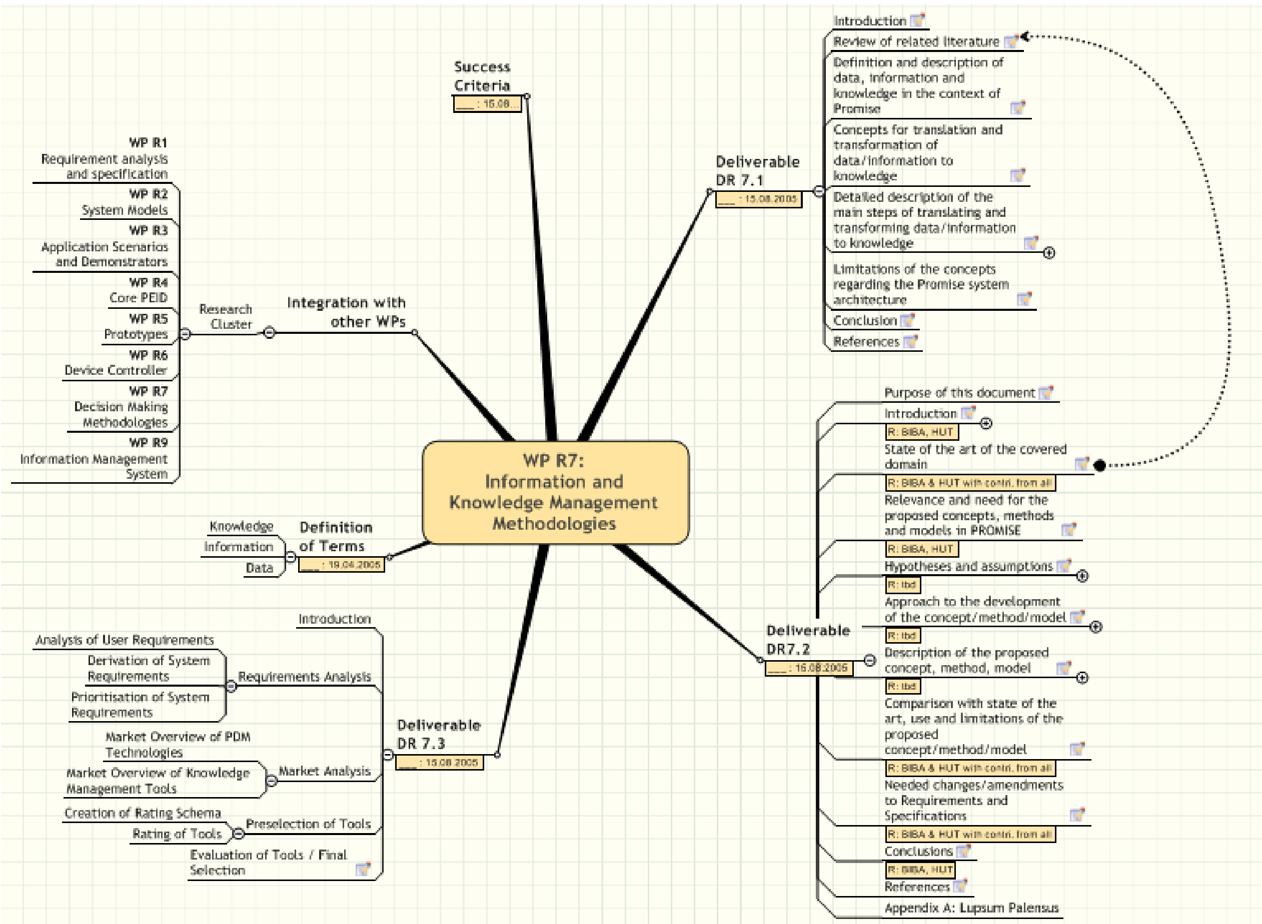
The mind map continues to be used in various semantic network forms, and for various applications including computer learning, learning and education, and in engineering diagramming.

Sources:

- http://en.wikipedia.org/wiki/Mind_map

Tools:

- **MindManager** / http://www.mindjet.com
- **FreeMind** / http://freemind.sourceforge.net
- **MindMapper** / http://www.mindmapperusa.com



3.2.3.7 Best Practice

The term best practice generally refers to the best possible way of doing something; it is commonly used in the fields of business management, software engineering, and medicine, and increasingly in government.

In business management, a best practice is a generally accepted “best way of doing a thing”. A best practice is formulated after the study of specific business or organizational case studies to determine the most broadly effective and efficient means of organizing a system or performing a function. Best practices are disseminated through academic studies, popular business management books and through "comparison of notes" between corporations.

The idea behind best practices is to create a specification for what the best methodology is for any given situation. Then, one can compare job performance to these best practices and determine if either the job performance was lacking in quality somehow, or if the specification for best practices needs updating to include the job performance being graded.

The management movement of Best practices might imply that many if not most situations are repeatable and that if we can sufficiently distill a set of experiences, we can predict all or most of the possible scenarios and the way to best handle them.

There is some momentum behind good practice as a preferred term, since it does not imply that no further innovation or revision is required.

Sources:

- http://en.wikipedia.org/wiki/Best_practice
- http://www.oit.doe.gov/bestpractices/software_tools.shtml

Tools:

- **Best Practices Database** / http://www.bestpractices.org/
- **DOE Industry Tools** / http://www.oit.doe.gov/bestpractices/software_tools.shtml

3.2.3.8 Knowledge Log

A knowledge log is a web-based publication (weBlog) consisting primarily of periodic articles (normally in reverse chronological order), that focus on knowledge.

The format of weBlogs varies, from simple bullet lists of hyperlinks, to article summaries or complete articles with user-provided comments and ratings. Individual weBlog entries are almost always date and time-stamped, with the newest post at the top of the page, and reader comments often appearing below it. Because incoming links to specific entries are important to many knowledge logs, most have a way of archiving older entries and generating a static address for them; this static link is referred to as a permalink. The latest headlines, with hyperlinks and summaries, are frequently offered in weBlogs in the RSS or Atom XML format, to be read with a feed reader.

There are no special Tools for Knowledge Logs, but due to the fact, that Knowledge Logs are simply weBlogs, all weBlog tools can be used.

Sources:

- http://en.wikipedia.org/wiki/WeBlog

Tools :

- **b2Evolution** / http://www.b2evolution.net/
- **bBlog** / http://www.bBlog.com/
- **BLOG:CMS** / http://Blogcms.com/
- **Blosxom** / http://www.blosxom.com/

3.2.3.9 Speech recognition

Speech recognition technologies allow computers equipped with a source of sound input, such as a microphone, to interpret human speech, e.g. for transcription or as an alternative method of interacting with a computer.

Such systems can be classified as to

- whether they require the user to "train" the system to recognize their own particular speech patterns or not
- whether the system is trained for one user only or is speaker independent
- whether the system can recognize continuous speech or requires users to break up their speech into discrete words
- whether the system is intended for clear speech material, or is designed to operate on distorted transfer channels (e.g. cellular phones) and possibly background noise or other speaker talking simultaneously
- whether the vocabulary the system recognizes is small (in the order of tens or at most hundreds of words), or large (thousands of words)

Speech recognition technology is used more and more for telephone applications like travel booking and information, financial account information, customer service call routing, and directory assistance. Using constrained grammar recognition (described below), such applications can achieve remarkably high accuracy. Research and development in speech recognition technology has continued to grow as the cost for implementing such voice-activated systems has dropped and the usefulness and efficacy of these systems has improved. For example, recognition systems optimized for telephone applications can often supply information about the confidence of a particular recognition, and if the confidence is low, it can trigger the application to prompt callers to confirm or repeat their request (for example "I heard you say 'billing', is that right?"). Furthermore, speech recognition has enabled the automation of certain applications that are not automatable using push-button interactive voice response (IVR) systems, like directory assistance and systems that allow callers to "dial" by speaking names listed in an electronic phone book. Nevertheless, speech recognition based systems remain the exception because push-button systems are still much cheaper to implement and operate.

Sources:

- http://en.wikipedia.org/wiki/Speech_recognition

Tools:

- **CMU Sphinx** / http://sourceforge.net/projects/cmusphinx/
- **Dragon NaturallySpeaking** / http://www.scansoft.com/naturallyspeaking/
- **Speech Server** / http://www.microsoft.com/speech/default.mspx
- **LumenVox** / http://www.lumenvox.com/

3.2.3.10 Taxonomy

Taxonomy may refer to either the classification of things, or the principles underlying the classification. Almost anything, animate objects, inanimate objects, places, and events, may be classified according to some taxonomic scheme.

Taxonomies are frequently hierarchical in structure. However taxonomy may also refer to relationship schemes other than hierarchies, such as network structures. Other taxonomies may include single children with multi-parents, for example, "Car" might appear with both parents "Vehicle" and "Steel Mechanisms". A taxonomy might also be a simple organization of objects into groups, or even an alphabetical list.

Sources:

- http://en.wikipedia.org/wiki/Taxonomy

Tools:

- **a.k.a. Classification Software** / http://www.a-k-a.com.au/aka_classification/
- **NeuroXL Classifier** / http://www.neuroxl.com/white_paper_neural_networks.htm
- **Autonomy** / http://www.autonomy.com/content/home/

3.2.3.11 Topic Map

Topic Maps are an ISO standard for the representation and interchange of knowledge, with an emphasis on the ability to find information. The standard is formally known as ISO/IEC 13250:2003.

A topic map can represent information using topics (representing any concept, from people, countries, and organizations to software modules, individual files, and events), associations (which represent the relationships between them), and occurrences (which represent relationships between topics and information resources relevant to them).

Topics, associations, and occurrences can be typed, but the types must be defined by the creator of the topic maps, and is known as the ontology of the topic map. There are also additional features, such as merging and scope. The concept of merging and identity allows automated integration of topic maps from diverse sources into a coherent new topic map.

Topic maps have a standard XML-based interchange syntax called XML Topic Maps (XTM), as well as a de facto standard API called Common Topic Map Application Programming Interface (TMAPI), and query and schema languages are being developed within ISO.

Sources:

- http://en.wikipedia.org/wiki/Topic_map

Tools:

- **Ontopia Topic Map Engine** / http://www.ontopia.net/solutions/engine.html
- **Intelligent Topic Manager** / http://www.mondeca.com/technologie.htm

3.2.4 Management of explicit knowledge

Product design and manufacturing information is usually stored in CAD/CAE and PDM systems. Such systems represent and maintain physical properties of products and product structures as well as product versions and variants. The “mySAP PLM” system described in section 3.5.4 is an example of such a system. Current PLM systems provide most PLM functionality needed for the BOL phase as long as design and manufacturing data are created and used within a single organisation or within a limited and stable group of organisations. When considering MOL and EOL phases it becomes necessary to agree upon compatible representations of product information. The STEP standard currently seems to be the most used for exchanging design and manufacturing data (especially 3-dimensional product models) between different systems and organisations. After the section on STEP the semantic net concept is studied as a way of representing relations between different data objects, followed by XML-based solutions for representing such semantic nets. The reason for this transition from STEP to XML-based and semantic web-based solutions is that there also seem to be movements in the STEP standards towards XML-based solutions.

3.2.4.1 STEP, the Standard for the Exchange of Product Model Data

STEP is a comprehensive ISO standard (ISO 10303) that describes how to represent and exchange digital product information. It has been operational for 3D design data since the 1990's and has also been extended for exchanging numerical control (NC) data. The following description of STEP is has been adapted from http://www.steptools.com/library/standard/:

The ultimate goal is for STEP to cover the entire life cycle, from conceptual design to final disposal, for all kinds of products. However, it will be a number of years before this goal is reached. The most tangible advantage of STEP to users today is the ability to exchange design data as solid models and assemblies of solid models. Despite the many successes of STEP there is still a question in users minds about the speed of its development and deployment. Many critics point out that the XML standards for e-commerce are being developed much more quickly (Hardwick, 2004).

Fundamentally, product model data is different to other kinds of e-commerce data such as invoices, receipts, etc. The traditional method for communicating product model information is to make a drawing and the traditional method to communicate an invoice is to make a form. When you make a drawing or 3D model you need to define information with many subtle and complex relationships and this makes the STEP data exchange problem more difficult.

An XML data format is being developed for STEP but the STEP architecture requires the information requirements of an Application Protocol to be mapped into the common set of Integrated Resources. This allows all of the protocols to share the same information and is essential if all of the interfaces shown in Figure 7 are going to share and reuse the same set of data.

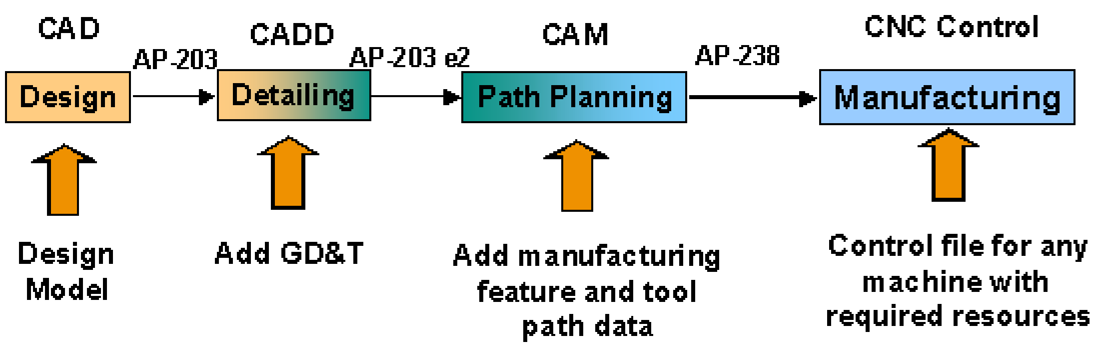


Figure 7. STEP application protocols (AP) and their role in managing BOL product information.

However, the sharing necessarily divides the original data into multiple entities that are not so easy to understand in XML or any other format. This is disappointing because one of the attractions of XML is that is self-documenting (at least for programmers and domain experts). Therefore, a new level of documentation is required in the STEP data to show how the information requirements have been mapped. The required structures are currently in development and it is anticipated that STEP will have a self-documenting XML format in the very near future.

STEP currently its uses own "EXPRESS" definition language for representing product information and product structures. Product structures are typically represented as semantic nets that are constructed from pair-wise named relations. In the web community there are several ongoing standardisation efforts based on XML that could be useful also for representing both product information and product structures. We will first study the concept of semantic nets, followed by a description of developments related to the semantic web and the standards being developed there.

3.2.4.2 Semantic networks

Semantic networks represent sets of named relationships between different nodes (or objects) in a network. Using a collection of pair-wise relations between nodes, where every relation may also have an associated "strength", can represent a semantic net. Relation strengths are particularly useful when semantic nets are used for reasoning, e.g. for diagnostic or prognostic purposes as those needed in many application scenarios of section 4. Such pair-wise relations are also included in the WWAI protocol of the PROMISE partner Stockway as described in section 3.5.5.

A short introduction on semantic networks has been adapted from the source "What Are Semantic Networks? A Little Light History (http://www.informatics.susx.ac.uk/books/computers-and-thought/chap6/node5.html):

The concept of a semantic network is fairly old in the literature of cognitive science and artificial intelligence, and has been developed in so many ways and for so many purposes in its 20-year history. The term dates back to Ross Quillian's (1968), in which he first introduced it as a way of talking about the organization of human semantic memory, or memory for word concepts. The idea of a semantic network - that is, of a network of associatively linked concepts - is very much older: Anderson and Bower (1973, p. 9), for example, claim to be able to trace it all the way back to Aristotle. Specifically, semantic networks were conceived as a representational format that would permit the "meanings" of words to be stored, so that humanlike use of these meanings is possible.

John Sowa (http://www.jfsowa.com/pubs/semnet.htm) gives a very good description of the use of semantic networks in different contexts. He defines a semantic network as follows:

A semantic network or net is a graphic notation for representing knowledge in patterns of interconnected nodes and arcs. Computer implementations of semantic networks were first developed for artificial intelligence and machine translation, but earlier versions have long been used in philosophy, psychology, and linguistics.

What is common to all semantic networks is a declarative graphic representation that can be used either to represent knowledge or to support automated systems for reasoning about knowledge. Some versions are highly informal, but other versions are formally defined systems of logic.

The use of semantic networks for diagnostic purposes is illustrated by the example in

Figure 8 (Främling, 1996). The semantic net only represents the connections. Calculation of probabilities of different diagnosis can be performed by Bayesian reasoning or similar techniques.

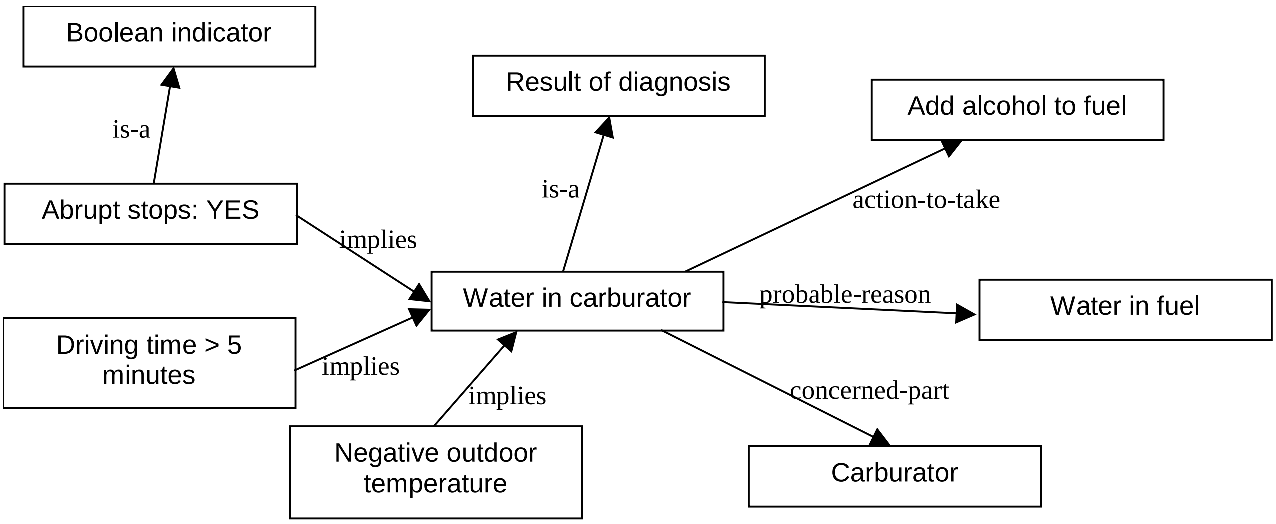


Figure 8. Semantic network for determining probability of the diagnosis "water in the carburator" of a car. Connection strengths are not indicated. Adapted from (Främling, 1996).

3.2.4.3 Semantic web

The Internet plays an essential role in order to close the PLM information loop. The semantic web is a concept that attempts to add meta-data to the contents of the Internet so that search functions, validation of the information etc. would become more efficient. The formatting standards for this metadata could also provide a possibility to representing and accessing distributed information about products and product structures (Berners-Lee et al., 2001).

The WWW consortium (W3C) defines the Semantic Web as follows (http://www.w3.org/2001/sw/):

*The **Semantic Web** provides a common framework that allows **data** to be shared and reused across application, enterprise, and community boundaries. It is a collaborative effort led by W3C with participation from a large number of researchers and industrial partners. It is based on the Resource Description Framework (RDF), which integrates a variety of applications using XML for syntax and URIs for naming.*

W3C describes the Resource Description Framework (RDF) as follows (http://www.w3.org/RDF/):

The Resource Description Framework (RDF) integrates a variety of applications from library catalogs and world-wide directories to syndication and aggregation of news, software, and content to personal collections of music, photos, and events using XML as an interchange syntax. The RDF specifications provide a lightweight ontology system to support the exchange of knowledge on the Web.

What might be of even greater interest for the PROMISE is the Web Ontology Language OWL (for a good overview of OWL, see http://www.w3.org/TR/owl-features/). The OWL reference pages give the following short description of OWL (http://www.w3.org/TR/owl-ref/):

The Web Ontology Language OWL is a semantic markup language for publishing and sharing ontologies on the World Wide Web. OWL is developed as a vocabulary extension of RDF (the Resource Description Framework) and is derived from the DAML+OIL Web Ontology Language. This document contains a structured informal description of the full set of OWL language *constructs and is meant to serve as a reference for OWL users who want to construct OWL ontologies.*

This section has discussed representation standards for product information. However, representing product information is not as such useful unless that information can be made accessible and communicated between all concerned parties. This is the subject of the next section.

3.3 Managing product-related knowledge and information in a multi-organisational context

As long as PDM can be performed in a centralized manner it is relatively easy because decisions on data formats and information processing methods can be taken locally. When PDM is distributed standards become essential both concerning the format of the data and how to exchange it. This section starts by an overview of EDI, which is currently the most used standard for exchanging information between organizations. Portal systems are covered in the same section. Then we give an overview of XML-based communication protocols that tend to replace EDI in inter-organizational information exchange.

3.3.1 Data models for accessing product information

The chronologically first possibility of conveying product information in electronic format was using EDI messages or some file-transfer protocol (Angeles and Nath, 2001). The logic of using EDI messages to convey product information is similar to the paper-based transfer, the information is sent to the next downstream supply chain partner (i.e. the primary customer).

Handling product information with electronic or paper-based messaging is problematic due to the following reasons:

- Many companies in the supply chain may not need the information for their own activities but they still have to be able both to receive and transmit the information to all their partners.
- All companies have to be able to communicate with each other. If one of the companies of the supply chain is unable to receive and transmit the information, then the information flow is interrupted.
- The product information that is sent and stored at various downstream companies is difficult to keep up-to-date. The producer of new information may not know what parties to inform about updates and thus companies with outdated information risk making decisions based on wrong information (Kärkkäinen et al., 2003).
- Transmitting all product information downstream may cause information overflow downstream in the supply chain (Beulens et al., 1999). This is especially true with complex products, in which the information related to the components of the product has also to be managed.
- EDI communication is usually expensive and takes long to set up between two partners, so it makes collaboration links more rigid than they would otherwise need to be (Johnston and Yap, 1998).
- Integrating current EDI-based solutions is too expensive for most small companies (Timm et al., 2001), therefore limiting the participation of small companies in the supply chain information exchange (Kärkkäinen and Ala-Risku, 2003).

The most popular way of overcoming these problems has been the use of Internet and Extranet technologies for transferring product information. In practice, forerunner companies have developed portal applications through which their customers can download the product information. Portals are Internet based services that can represent information from various systems in a single place, through a browser (Linthicum, 2001). Portals have proved to be efficient in making the product information available to supply chain participants and have thus grown increasingly common. However, portals are just the user interface representing the information to the consumer of the information. The issues related to the management of information, e.g. linking information to specific product items and handling the bill-of-materials of products, demand additional solutions.

3.3.2 XML-based communication protocols

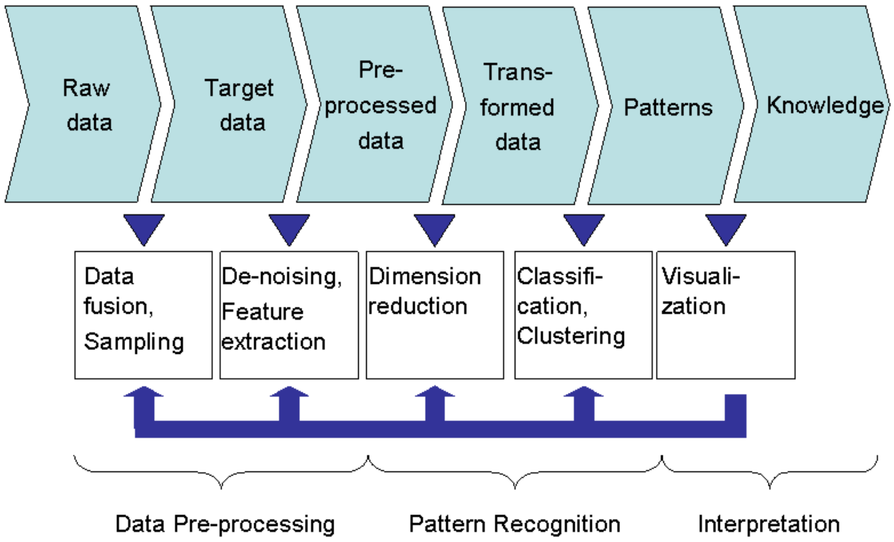
Web Services (WS, see http://www.w3.org/2002/ws/) seems to be the most universally accepted concept for XML-based communication between software components or applications over the Internet. The WS activities include several protocols for different tasks, notably the Simple Object Access Protocol (SOAP, http://www.w3.org/TR/soap/) for XML-based messaging and the Web Services Description Language (WSDL, http://www.w3.org/TR/wsdl) for describing services and their interfaces. Other activities concern messaging security, message persistence service composition etc. (Weerawarana et al., 2005). Especially SOAP and WSDL seem to gain in popularity for defining application interfaces (e.g. the Dialog platform described in 3.5.3 includes a SOAP/WSDL interface). The main advantage of WS is that they are relatively easy to configure (especially compared to EDI-based solutions) and could even allow for dynamic discovery of appropriate services for a given task.

Other XML-based communication protocols are ebXML (http://www.ebxml.org/) and RosettaNet (http://www.rosettanet.org/). The use of WS as the base level for these protocols has been studied. WS are also the cornerstone of the EPC-IS (EPC Information Services, see http://www.epcglobalus.org/Network/EPC_Information_Services.html) under development. For EPC-IS could become relevant for PROMISE but is still in a preliminary stage.

If we assume that applications can communicate over the Internet, the next step would be to react to incoming new information. In the context of PROMISE this could signify the filtering of dynamic field data, detecting anomalies, making decisions on how to react on anomalies etc. This is the subject of next section.

3.4 Methods for information enrichment of dynamic data

By realizing the vision underlying the PROMISE approach to Product Lifecycle Management, data are collected at an ever-increasing amount and pace. In order to extract useful information from it, sophisticated techniques, developed in the field of data mining, are being applied. We consider data mining as the end-to-end process of finding patterns starting with raw data.



It consists of two main steps, data pre-processing, during which relevant high-level features or attributes are extracted from the low level data, and pattern recognition, in which a pattern in the data is recognized using these features.

The pre-processing step mostly consists of the following sub-steps:

- In case the raw data is very large, a sampling will be used in order to work with fewer instances.
- Noise in the data is removed to the extent possible, and relevant features are extracted. When data is taken from different sources or sensors, data fusion may be required.
- Finally, a feature vector for each data instance is built.

In the second main step, the detection of patterns through the use of algorithms such as classification, clustering, regression, etc is carried out. These patterns are then displayed to the user for validation. Data mining is an iterative and interactive process. The output of any step, or feedback from the domain experts, could result in an iterative refinement of any, or all, of the sub-tasks.

In the subsequent sections the most popular algorithms are presented and how they are applied in feature extraction, feature selection, classification, clustering and visualization.

3.4.1 Information enrichment in PDM systems

A PDM system is defined as a technical database and communication system that enables to store and manage information about products and their development process in a consistent way. A system containing all product-related information automatically bears a great potential of enriching the available information with data and information from other sources in order to present the information in a more valuable way. This section deals with possibilities of information enrichment in PDM resulting from integration with other systems.

3.4.1.1 PDM-CAD integration

Managing computer-aided design documents played a great role in the historical development of PDM systems. The earlier integration concepts included data integration at the non-geometric level, e.g. titles, design notes and design attributes. Even this simple kind of integration enabled

PDM users to access to a more comprehensive information source and improved their work productivity.

Nowadays almost all modern PDM systems offer a product structure level of data integration. At the product structure level, the CAD design file is interrogated by the PDM system to not only extract standard non-geometric data, but to also sift through and extract out any information at assembly/component/part level.

Product structure information extracted from the CAD design file is then populated into the PDM system's data structure – enabling PDM users to actually visualize the assembly tree or product structure details that were only visible through the CAD products. This assembly tree or product structure, can include relationship information, as well as assembly/part level attributes, e.g. center-of-gravity, weight or material. This level of data integration opens up many doors to information enrichment options that span the lifecycle of the products.

Lifecycle visualization solutions through direct links to leading CAD applications enable the immediate integration of the work output of a design department into the complete business flow and is available to the extended business community. Lifecycle participants are able to visualize and leverage product data in 2D and 3D formats even when this data is created using different authoring software.

CAD integration facilitates more fully informed lifecycle decisions by allowing product teams to exchange, share and mark up universally accessible visualizations that represent the product in its entirety. Furthermore extended lifecycle participants (e.g. sales, marketing, service and maintenance, suppliers) can take part in the concept design process early and access to graphical product data in later lifecycle phases.

Collaborative design review tools in PDM systems have 2D/3D analysis capabilities for 3D investigation, including measurement, dynamic cross sectioning, comparison and user-defined coordinate system capabilities for analyzing CAD assemblies. Digital mock-up utilities enable design teams to assemble a complete digital prototype, perform advanced analysis – including static and dynamic clearance analysis on the whole product – and detect issues early in the product lifecycle.

3.4.1.2 PDM-ERP integration

Historically, PDM and ERP have been developed as separate systems under the control of different parts of the organization. ERP systems have generally been considered a manufacturing or even an overall business operations responsibility. Many organizations that use ERP for its business processes and PDM for product development has realized the importance of having a strong integration between these two systems. The on-going integration of product data management (PDM) with enterprise resources planning (ERP) will improve communications between engineering, manufacturing, purchasing, sales, etc.

A PDM-ERP integration enables the effective synchronization between the engineering bill of material (EBOM) and the manufacturing bill of material (MBOM). The goal is that the MBOM is modified to meet the requirements of the EBOM (where the EBOM is master).

A great information enrichment aspect is the MBOM visibility in PDM. It is often desirable for engineers to be able to use a PDM-ERP integration to browse through items, MBOMs and Engineering Changes located within the ERP system. Furthermore, the integration allows for engineers to retrieve what-if reports based upon information located within ERP, such as a cost-report for a given prototype design that the engineer has created. Any information that is related to the logistics chain could also be interesting for engineers, e.g. supplier and raw material information.

3.4.1.3 PDM-Project Management integration

Many PDM systems offer interfaces to project management tools that are designed to effectively control project structures, schedules, costs, and resources. Integrating project management tools with PDM enables the association of product data with project management related issues, e.g. tasks and milestones. This allows cross-discipline teams to create and share project information in a real-time environment that facilitates simultaneous project viewing. Benefits of information enrichment are e.g. listing of all documents that are related to a certain task or viewing of parts that have to be designed until a given milestone.

3.4.2 Gaining knowledge from raw data

3.4.2.1 Feature extraction

The process of extracting features that are relevant to the problem is very problem- and data-dependent. In PROMISE there are applications from various domains, i.e. automotive, telecommunication, building machines, etc.

In some cases, the features are relatively easy to identify, e.g. a sensor sending data representing the oil temperature. In other asset management applications it might be more challenging, e.g. when analyzing the sound of an engine for possible break downs. In this latter case, the first task is to understand the data stream as being the overall sound of the engine and to identify from that the sound of the different parts of the engine. Then, a set of features has to be extracted that can be used to decide whether one of the engine parts creates an unexpected sound.

The identification is usually the more difficult of these two tasks, as it involves the conversion of the low-level representation (i.e., stream of sound data) into a higher-level representation (i.e., engine parts). Alone the separation of noise and errors from incoming sensor data is a field of extensive research. For each industrial application very specific filters will be developed in order to extract the “real” data out of the incoming stream.

3.4.2.2 Feature selection

Once the relevant features representing the data items have been extracted, it is often helpful to reduce this set of features. There are several reasons for this. In many situations, it is not possible to know a priori which features extracted from the data will be relevant to the problem at hand. Including features that are irrelevant not only increases the time complexity of many algorithms, but also increases the time needed to extract the features. Further, as the number of examples needed for learning a concept is proportional to the dimension of the feature space, fewer training examples will be required if the number of features is reduced. This leads to the problem of feature subset selection which is the task of identifying and selecting a useful subset of features to be used to represent patterns from a larger set of often mutually redundant, possible irrelevant, features.

The simplest way to remove irrelevant features is to apply domain knowledge. For example, if we are interested in a specific part of an engine, it is has to be checked whether data related to other parts are needed. However, this approach is feasible only when a domain scientist can easily identify irrelevant attributes, which is rarely the case. More complex techniques such as principal component analysis can also be used to obtain linear combinations of attributes by projecting them along the directions of the greatest variance.

3.4.2.3 Statistics

Statistics have long been used to create a model of data sets.

Discriminant analysis is the study of finding a set of coefficients or weights that describe a Linear Classification Function (LCF), which maximally separates groups of variables. It is popular when attempting to find common groupings of variables.

A **regression equation** is one that estimates a dependent variable using a set of independent variables and a set of constants. Classification studies can be constructed with traditional statistic regression techniques. Linear regression models attempt many of the same things that other data mining approaches do, i.e., making predictions or classifying data.

3.4.2.4 Classification

The classification or supervised learning approach is the most popular one. For example, the behavior of a truck can be analyzed by classifying the age of the truck into categories less than one year old, between one and 10 years old and older than 10 years. Once data is classified, the traits of these specific groups can be summarized and corresponding rules can be identified, e.g., "if the truck is older than 10 years, check the oil filter". Thus, classification is finding the best segmentation of the features with respect to a defined problem. For each problem the segmentation is different and has to be computed by applying adequate algorithms.

In contrast to unsupervised learning, this way of building a model requires knowledge about the behavior of the object in previous cases, i.e., a training set is needed. In this training set the examples are labeled with the actual class and the correct answer. For instance, a set of data is needed, in which the age of the trucks are related with the time when their oil filter had to be changed. On this basis, the algorithms compute the best time to change the oil filter.

3.4.2.5 Clustering

Clustering or unsupervised learning is a method of grouping rows of data that share similar trends and patterns. Clustering, or segmentation, is the process of dividing a data set into distinctive groups. For example, someone may want to cluster all cars in certain groups depending on their usage. In contrast to supervised learning, clustering studies have no dependent variable. There is no specific trait that has to be profiled as in classification studies.

3.4.2.6 Visualization

Visualization is simply the graphical presentation of data. Data can sometimes be best understood by graphing it. For example, visualization techniques can easily show outliers. Outliers are values that are clearly not in the range of what is expected. The process of representing data graphically is used today in most query tools.

3.4.3 Machine Learning algorithms

We next discuss the ways in which algorithms can be used to address the above-described problems. Greedy top-down construction is the most commonly used method for tree growing today. A hierarchical model can be constructed top-down, starting from the entire data, somehow partitioning it into subsets, and recursing the partitioning procedure. A description of tree growing then reduces to a description of techniques for splitting data into meaningful subsets. This splitting problem can be tackled with many mathematical approaches from which the most common are presented in the subsequent sections.

3.4.3.1 Decision Trees

Creating a tree-like structure to describe a data set has been used for quite some time in computer science. Herein, all decision-tree algorithms undergo a similar type of process, they employ different mathematical algorithms to determine how to group and rank the importance of different features. At the beginning, features are chosen from a data source and a dependent feature, e.g. engine break down, is chosen by the user.

Afterwards, each feature affecting the dependent feature is examined. An iterative process of grouping values together is performed on the values contained within each of these features. For example, the age of the truck is evaluated as being a good indicator for identifying the best time to change the oil filter. This process will be repeated with all features and the resulting sub-tree will position the most expressive feature as its root node.

3.4.3.2 Genetic Algorithms

Genetic algorithms are a method of combinatorial optimization based on processes in biological evolution. The basic idea is that over time, evolution has selected the "fittest species." Applying this idea to data mining usually involves optimizing a model of the data using genetic methods to obtain "fittest" models. Genetic algorithms are good at clustering data.

First, a random grouping of data is chosen. Think of each of the clusters to be created as an organism. The genetic algorithm will have what is called a fitness function that determines if a data set is a match for one of the clusters. This fitness function could be anything that identifies some data sets as "better fits" than others. As data sets are read, they can be evaluated by the fitness function to see how well they relate to the other data elements in a cluster.

Second, genetic algorithms have operators that allow for copying and altering of the descriptions of groups of data. These operators mimic the function found in nature where life reproduces, mates, and mutates. If a row of data in a data set is found to be a good fit by the fitness function, then it survives and is copied into a cluster. If a row of data is not a good fit, it can be crossed over to another set, or, in other words, it can be mated.

3.4.3.3 Neural Networks

Neural networks are used extensively as predictive models. They attempt to mimic a neuron in a human brain, with each link described as a processing element (PE), Neural networks learn from experience and are useful in detecting unknown relationships between a set of input data and an outcome. Like other approaches, neural networks detect patterns in data, generalize relationships found in the data, and predict outcomes.

A PE processes data by summarizing and transforming it using a series of mathematical functions. One PE is limited in ability, but when connected to form a system, the PEs create an intelligent model. PEs are interconnected in any number of ways and they can be retrained over several, hundreds, or thousands of iterations to more closely fit the data they are trying to model.

The process of training the network involves modifying the strength, or weight, of the connections from the inputs to the output. Training repeatedly, or iteratively, exposes a neural network to examples of historical data. PEs summarize and transform data, and the connections between PEs receive different weights. That is, a network tries various formulas for predicting the output variable for each example. Training continues until a neural network produces outcome values that match the known outcome values within a specified accuracy level, or until it satisfies some other stopping criterion.

3.4.3.4 Bayesian Belief Networks

While Bayesian networks are powerful tools for knowledge representation and inference under conditions of uncertainty, they were not considered as classifiers until the discovery that Naïve-Bayes, a very simple kind of Bayesian networks that assumes the attributes are independent given the class node, are surprisingly effective.

A Bayesian classifier is a directed graph in which there are nodes that represent each of the features and a node that represents the class. The node that represents the class is called the class node and the other nodes are called evidence nodes. Each node in a Bayesian classifier has a state for each possible value the associated feature may take on. Saying that the node is in a particular state is equivalent to saying the feature has the specific value associated with that state of the node. Each node is labelled by a probability distribution over its states. That probability distribution represents the probability that the associated feature will take on each of its possible values. These are initialized to the probabilities that the attribute will take on each of its possible values in the population being modelled.

Once the Bayesian classifier has been built, classifications of new entities can be predicted by specifying the state of each node associated with a feature whose value is known for the new entity. This is called instantiating the evidence nodes and is accomplished by changing the probabilities stored at these nodes so that the current state has a probability of 1 and all of the other states have probability of 0. The process of instantiating the known evidence nodes and re-calculating the probability distribution of the class node is called inference. The result of inference is a new probability distribution for the class node representing the probabilities that the test case is in each class.

A major advantage of Bayesian networks over many other types of predictive models, such as neural networks, is that the Bayesian network structure represents the inter-relationships among the dataset attributes. Human experts can easily understand the network structures and if necessary modify them to obtain better predictive models. By adding decision nodes and utility nodes, Bayesian network models can also be extended to decision networks for decision analysis. The following classifiers are the most popular ones: Naïve-Bayes, Tree augmented Naïve-Bayes (TANs), Bayesian network augmented Naïve-Bayes (BANs), Bayesian multi-nets and general Bayesian networks (GBNs).

3.4.3.5 Case-based reasoning

Case-Based Reasoning (CBR) has emerged from research in cognitive psychology as a model of human memory and remembering. It has been embraced by researchers of AI applications as a methodology that avoids some of the knowledge acquisition and reasoning problems that occur with other methods for developing knowledge-based systems.

Often CBR is applied when the application domain is not well-understood. By having access to a huge database of historic cases problem solving is reformulated to retrieving very similar cases and analyzing their solutions by the human engineer. This suggests that developing CBR systems may require less knowledge engineering than, say, rule-based or model-based approaches. It is generally accepted among CBR researchers that this is only true to a limited extent. A CBR system that is not built on the type of domain analysis that knowledge engineering involves will probably not work very effectively.

3.5 Pre-existing concepts, methods and models of involved partners

The previous sub-sections have attempted to give an overview mainly on existing concepts, methods and models that could be used for augmenting PLM according to the needs in PROMISE. In this section we study existing concepts, methods and models mainly of PROMISE partners involved in this WP R7 and to what extent they already integrate the concepts presented in the previous sub-sections.

3.5.1 BIBA

With the increased use of flexible team structures in the agile enterprise where teams rapidly form and disband, the risk of loss of crucial knowledge is high. Also the risk of duplication efforts in different parts of such agile enterprises increases due to a lack of adequate communication.

A considerable amount of industries have started to explore the possibilities of the creation of virtual network organisations dedicated to exploit the competencies residing within these regions (e.g. Prato Italy, Virtual Factory Bodensee). Currently available solutions follow the logic of the industrial production process, where chains of actions are treated as workflows. Also the sheer amount of information and the need to identify and isolate relevant information is rapidly becoming a crucial factor.

Various technologies are currently available for these purposes, such as Lotus Notes, but they do not easily combine personal information management facilities with collaborative working support nor do they easily cater for the creation of environments that specifically support innovation and creativity. The shortcomings are mainly due to the fact that these solutions are predicated on standard structures and procedures where knowledge is gathered during workflows at specified points and do not leave space for creativity that goes beyond the specified framework.

The problem is to develop processes of knowledge creation and reuse in the organisation (management, teams) and systems that support them. Thus the need for an individual information and knowledge management approach has to be combined and integrated with a virtual information and knowledge management solution.

Whilst preserving the need for a solution that is flexible enough to allow for creative freedom, an approach is needed that uses a common core ontology or reference model to allow for adequate communication capabilities between individuals in the virtual enterprise. There is a need for a mechanism that turns the creativity of European firms, especially SMEs into the source of sustainable competitive advantage.

In order to create a European culture of innovation and improve the efficiency of turning scientific and technical knowledge into a successful product, there is a need for a system that effectively records knowledge in a form that is useful to the individual and can easily be accessed by others in virtual enterprises.

The NOPIK (Personal Information and Knowledge Organiser Network, IST-2001-33487) Suite is a solution for the information and knowledge management needs of users that can be immediately exploited by them and further developed into a commercial product. Aim of the NOPIK suite is to overcome the shortcomings of available solutions (process-orientation, lack of intuitive organization principles, improved support of the individual worker). Basic design principles is to utilize extensible ontologies which can be created individually with a ontology management module as an organizing principle for classification, storage and retrieval of information (documents etc.) and associated knowledge (e.g. logic database systems) which allows thematic and context sensitive browsing. Additionally to this the Case Based Reasoning (CBR) methodology is implemented to increase the reusability of existing knowledge. Problem and solution descriptions are based on the ontologies which are created by the ontology management module. Exchange of ontologies is realized by utilizing the RDF Ontology language for data exchange. The distributed information system is fully web-based and supporting both on-line and off-line working.

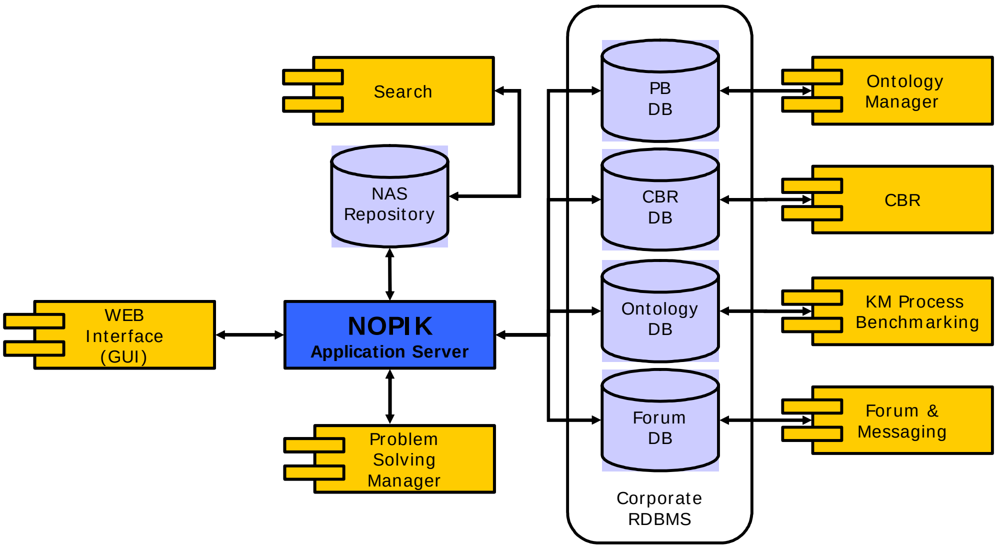


Figure 9:NOPIK system component architecture

The main components of the NOPIK system (which are fundamentally the same in the PIKO client as in the IKON server) may be briefly described as follows:

- a web Graphical User Interface, serving as a personal desktop organizer and as a unifying environment for the management of the system functionalities at corporate level;
- a Problem Solving or Workflow Manager (PS tool) acting as a process support and messaging facility;
- a Knowledge & Information Repository, where personal and corporate relevant data (Cases, Ontologies) are stored and manageable in a user friendly way;
- an interface to Legacy Systems and Users databases;
- an Ontology Suite, acting as a company information and knowledge management system which allows users to create graphical document maps, navigate and store them in the Ontology Repository which resides on the IKON server;
- a CBR Module, supporting the Problem Solving activities inside the (extended) Company, which enables a user to formulate a new problem, retrieve a list of possible solutions, revise the solution selected and finally store it in a database of "solved cases";
- a Performance Indicators Module, enabling users to "rate" the solutions provided by the PS / CBR activities and to store and maintain previously created indicators in a specific database;
- a Search Engine on all of the above mentioned databases;
- a core module (NAS – NOPIK Application Server) that works as the back office system of both PIKO and IKON, coordinating the transactions to the different repositories of the other system components, and in charge of the interoperability between PIKOs and IKON.

3.5.2 CogniData

Since the year 2002 Cognidata is developing a decision support system in the area of diagnosis of large rotating electrical machines. The system is intended to make use of the historical experience collected over many years by human experts in the field. In this approach we are constructing experience items on the fly in accordance to users' needs.

3.5.2.1 The decision problem

In power plants, large rotating high voltage electrical machines like turbo or hydro generators operate almost uninterruptedly to produce electrical energy. These electrical machines are highly complex and designed to serve a large life span, but unfortunately, one of their most important components, the insulation system of the winding, is in the same time their weakest component, because it is subject to failures and irreversible ageing. To avoid a machine breakdown resulting from insulation system failures, machine operators rely on diagnostic services offered by specialized companies that determine when condition-based maintenance is necessary. To perform a diagnosis, an experienced engineer equipped with special measuring devices has to go to the site of the operator (customer) and carry out a series of measurements and visual inspections on the machines. Based on the collected data, the history of the machine and human experience, the engineer decides whether faults are present, what source they might have and what kind of maintenance work is necessary to eliminate them.

3.5.2.2 The computer-based decision support system

In cooperation with its customer Cognidata implemented an Internet-based decision support system that assists the experienced and inexperienced engineers of our project partner in performing their tasks of diagnosing electrical machines situated all over the world. Since the act of diagnosis on a specific machine can be thought of as an experience item (case) or sometimes a kind of reinforcement of a previously learned lesson, it is reasonable to use case-based reasoning (CBR) technology at the heart of this decision support system. However, although the mainstream of research on CBR is founded on the premise of having a case base were the cases are stored, recent research directions indicated Cognidata that this is a promising solution. Therefore, the present approach is that of constructing experience items (cases) on the fly whenever the necessity will arise, instead of the usual way of building a case base.

In practice that means constructing cases, because the data and information that will constitute a case are found in different sources and need to be brought together in the known form of case representation (problem description; problem solution). There are several reasons why Cognidata took this approach and did not proceed as usual by creating a case base. First, it was intended to build a decision support system, i.e., in every moment only the human expert will be making decisions, and what was needed is support in the form of similar previous situations. In this way, from the entire cycle of CBR methodology only the steps of case representation and case retrieval were needed. Second, in a previous effort to extract knowledge from the database of historical measurements with data mining methods it was found out that it was hardly possible to extract static classification rules for the domain data, or formulated differently, in the multi-dimensional space populated by the data instances different classes overlapped. This made us consider the possibility of avoiding the use of classification rules (which create classes that are equivalent to the concept of generalized cases in CBR), and instead implement a context-sensitive dynamic retrieval of instances from the database. And last but not least, the existing information that should be taken into consideration to create the cases is large and no domain expert is available and willing to take the effort building it. For these reasons, the reverse approach was undertaken: constructing cases on the fly, and evaluating the achieved performance.

3.5.3 Helsinki University of Technology

The importance of solid product information management practices and systems is increasing due to the intensifying technical sophistication of products, as well as the governmental regulations demanding efficient product lifecycle management (Töyrylä, 1999; Hamilton, 2001; Kärkkäinen et al., 2003). Recent developments in information transfer and data storage capacities enable distributing vast amounts of product and component information forward in the supply chain and thus overcoming the problems associated with paper-based data transfer. However, in complex products this can lead to information overflow in the downstream supply chain, when the amount and sophistication of product components increases. Another challenge is to maintain the information up-to-date for the relevant supply chain members during the products' lifecycle.

"Product centric information management" in which information regarding a product is retrieved over information networks when needed using unique product identities has been proposed as a solution to this challenge (McFarlane et al., 2002; Wong et al., 2002; Kärkkäinen et al., 2003). During the review of current practices it was noticed that it is a formidable challenge to link the product related information to the products themselves (Kärkkäinen et al., 2003). Making the information of all the product components easily achievable without the risk of downstream information overflow proved to be especially challenging. Software agents were seen as one possible answer to these challenges. A fundamental issue in this approach is how to associate the software agents with their physical counterparts. For creating this connection each physical product has to have a unique identity, which can be used as a reference to locate the product's agent on the Internet. At HUT an item coding of the format ID@URI has been used (Främling, 2002; Huvio et al., 2002), where the URI is an Internet address of the server where the product agent is located and the ID part is a product identity that is unique inside that server. This identification is, by definition, globally unique. The uniqueness of the URI part is guaranteed by the Domain Name System (DNS) used on the Internet, while the manufacturing company should be able to guarantee the uniqueness of the ID part. This makes the allocation of the codes simple and, as current product codes can be used, the coding is potentially very easy to integrate to current information systems. Existing standards, such as the Global Trade Item Numbers (GTIN) (EAN International, 2001) can also be used for the ID part, as well as EPC numbers. The Global Location Number (GLN) (UCC, 2002) is also globally unique and provides a standard means to identify legal entities, trading parties and locations. GLN cannot directly be used as a reference to a product agent, but it is useful for identifying physical locations in tracking applications, for instance.

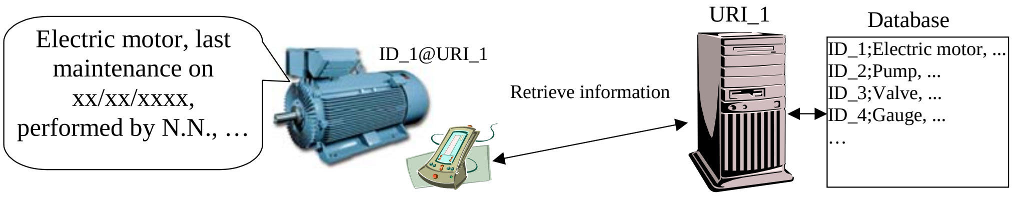


Figure 10. Accessing product information through ID@URI.

In agent-based solutions utilising the ID@URI notation, the product information can be made available everywhere where Internet access is available, without developing additional information registries. The URI part of the product identity directly tells where to find the information, while the ID part tells what physical product item the information is asked for (illustrated in Figure 10). The software component at the given URI can therefore act as the product agent of the particular product item and maintain the product data and links towards other sources of information on the product item.

In the agent model, product related information is retrieved and/or updated using the product-specific reference and only when needed as in Figure 11. The retrieved information depends on the party asking for the information and the specifications of the request. Therefore only data that is useful and for which access and usage rights exist for the inquirer is transmitted. It is important to note that some data are related to groups of similar physical products (such as user instructions for products of the same model), while other data are specific to an individual physical unit (such as maintenance records). Both of these can be handled efficiently with the agent model, and in particular, a piece of information that is common for several products is easier to keep up-to-date as it needs to be stored in only one place.

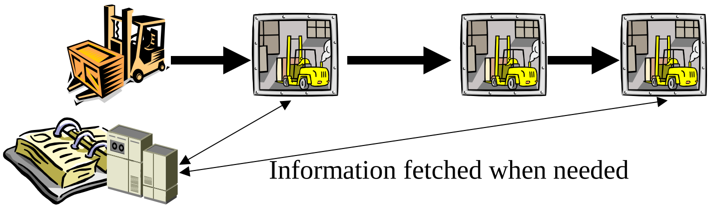


Figure 11. The "agent model" for real-time access to product information.

In product information management, information access can be split into two main functions:

1. *Accessing product data*, i.e. the ability to read and review specific information regarding a product. Examples of product data that need to be accessed are user instructions, maintenance records, assembly instructions etc.
2. *Updating product data*, i.e. the ability to append or amend the information regarding a product. Typical updates concern maintenance records, status monitoring of machines etc.

These functions can be handled by traditional means in a single-company setting but become challenging in a multi-company setting (Luckham, 2002). With the agent model, there is no difference whether it is applied to a single- or multi-company setting. All information requests for a given physical product item are performed with equal methods over the Internet and the information availability depends on the security class of the information and the authorisation codes of the inquirer. Each product's agent can manage the access rights as required.

The pilot middleware system called "Dialog" has been developed at HUT for accessing and updating product-related information in the ways described. The software is available at "http://dialog.hut.fi" and is distributed under the terms of the GNU Lesser General Public License. And has mainly been used for testing and verifying new information management concepts and models. The software has also been used in two industrial asset-tracking pilots in a multi-enterprise setting (in 2002 and in 2004). In addition to these basic functionalities of accessing and updating product information, the system has since 2004 been developed in the direction of supporting named relations between products and product items. At the lowest level these named relations should enable the construction of multi-organizational semantic networks between products, product items or other information entities. Such semantic networks can be easily used for implementing well-known data structures and algorithms such as object-oriented design patterns (Gamma et al, 1995). Design patterns offer tested solution models for organizing and using information in a flexible and standard way.

3.5.4 SAP

SAP's software solution for product lifecycle management is called "mySAP PLM". It can manage documents and information of the whole product lifecycle. It consists of several components, which address various issues in PLM. Here we will focus only on the functionality that is relevant for the application scenarios in PROMISE.

3.5.4.1 Lifecycle Data Management

Lifecycle data management is the foundation of the PLM system. It contains a document management with versioning, status management, and change management. The structure of a product can be stored in various stages. These structures are labelled accordingly “as required”, “as designed”, “as planned”, “as built” and “as maintained” (Figure 12).

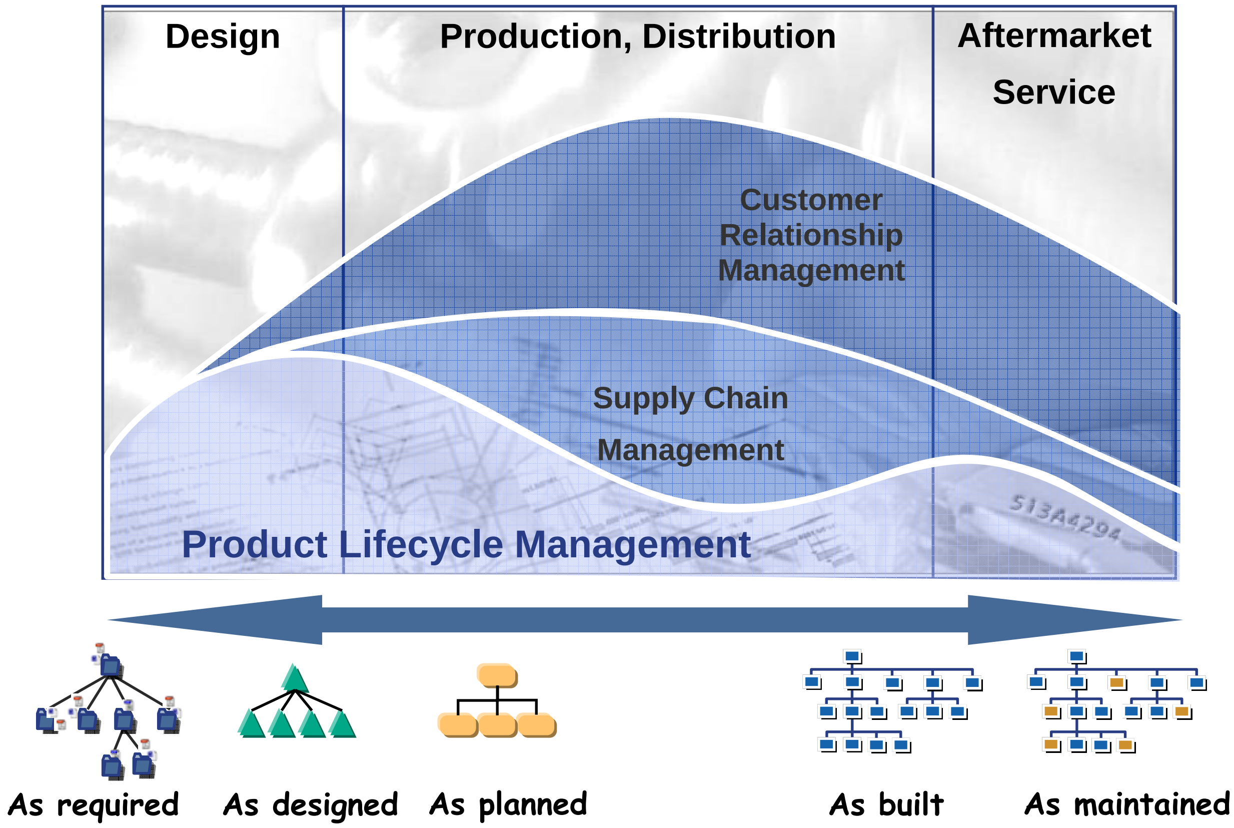


Figure 12: mySAP PLM supports the entire lifecycle

To keep track of all changes in the product lifecycle, the Configuration Management allows to setup folders for all phases. Within these folders, an arbitrary number of product versions can be defined. The Change Managements ensures that product data remains consistent during the change processes. The system also supports product variants as rules for allowed product variants can be stored in a knowledge base.

3.5.4.2 Enterprise Asset Management

Enterprise Asset Management focuses on the operation of a product or machine. It provides functionality to support maintenance and service. To do that, it has a data base with technical facilities and their properties. Based on that, maintenance tasks, including required tools and personnel can be scheduled. Upon completion the maintenance staff can store a report in the system to document their activities and results.

3.5.4.3 Lifecycle Collaboration

The mySAP PLM system offers capabilities to enable internal and external collaboration. Internal collaboration is used when the product is prepared for production. In this case engineering and production departments have to work together. External collaboration can be used to work with other companies that support the engineering phase. In either case, information can be shared using collaborative Folders (*cFolders*).

3.5.4.4 Environment, Health & Safety

This component handles issues of product safety, dangerous goods management, safety at work, and waste management. For PROMISE the functionality regarding recycling will be of interest. Waste management allows supporting waste disposal and recycling processes, including related reports and distribution of cost on respective cost centres. Product safety manages the used materials, waste information, classification for dangerous goods, and others.

3.5.5 Stockway

The Finnish company Stockway Oy has defined a protocol called the World Wide Article Information (WWAI) protocol, implemented in their Trackway™ software product. WWAI offers a P2P solution to defining and accessing any piece of data (product information in databases, design documents etc.). The protocol also gives a general-purpose possibility to define arbitrary named relations between these pieces of data as well as to construct queries that use these relations. Therefore WWAI offers a general-purpose framework for constructing distributed semantic nets of product information. The P2P-based architecture also makes it easy to verify the identity of requesting parties and locally (e.g. within a company) control the access and use of this information.

A WWAI relation element contains information about the relation between two WWAI Objects. If the WWAI Event object (see http://www.wwai.org/tech_specifications.html) was like the whole sentence, the WWAI Relation is kind of a verb in your WWAI sentence.

```
<wwai-relation relation-type="" relation-tense="" not-before="" not-after="">
  <parent-wwai-object-id></parent-wwai-object-id>
  <child-wwai-object-id></child-wwai-object-id>
</wwai-relation>
```

The wwai-relation element has got attributes relation-type describing the type of the relation and relation-tense describing the tense of the relation. Attributes not-before and not-after may be used to limit the relation's lifetime.

As a conclusion, WWAI and software products based on it can provide the necessary infrastructure for distributed representation of and access to general-purpose product information, including information represented as any kind of semantic net.

4 Relevance and need for the proposed concepts, methods and models in PROMISE

Closing the information loop in PLM signifies using product information in different ways to improve the design, use or recycling of products. Sections 4.1 to 4.3 give an introduction to the use of product information in these three phases of the product lifecycle. Section 4.4 analyses the application scenarios specified by PROMISE end-user companies both from the perspective of the phases of the product lifecycle and from the perspective of static and dynamic product information. The result from this analysis serves as a base for section 5 and the following sections that outline potential solution designs that are applicable.

4.1 Improving product design (BOL)

The concept of improving product design to facilitate maintenance has been addressed in different contexts. Many authors have claimed that the collected data on the product's technical heath can be used for the development of new generation of products. One of the oldest references is from Ives and Vitale in 1988. They stated that improving product design is one of the three approaches to meeting customer's maintenance requirements. They emphasized that electronic monitoring devices can provide the manufacturer with valuable data to be used in the design of future products. A more recent study by Simon et al. (2001) suggested that data collected during the use of a product can be used to change the product design to attain better reliability and conformance to customers' needs. Markeset and Kumar (2003) also have pointed out the same prospect. They suggested that the collected data could be used for developing future generations of products, but most importantly to remove or reduce critical weaknesses in the product design.

4.2 Equipment maintenance (MOL)

Tsang (1995) stated that "the primary objective of equipment maintenance programs is to preserve system functions in a cost-effective manner". There are different ways or approaches how a company can arrange its maintenance function in order to reach this goal, and usually the main issue is when the maintenance tasks should be carried out. In this chapter, different approaches, or maintenance methodologies, are examined based on two classifications by Tsang (2002) and Hill (2000).

The simplest approach in Tsang's classification is *run-to-failure* (RTF). It means maintenance tasks are carried out after the failure has occurred. Hill identifies this as *reactive maintenance* where equipment is repaired as needed. They both point out that this approach is justified only when the impact of breakdown is inconsequential or the benefits of some other maintenance methodology would not outweigh its costs.

The second approach by Tsang is *scheduled preventive maintenance*, which means equipments are replaced or returned to good condition before failure occurs. This is achieved when maintenance tasks are performed following a time or usage based schedule. Optimal schedules can be determined through the use of quantitative decision models, but they are often drawn up only on the supplier's recommendations. Hill's definition for *scheduled preventive maintenance* is congruent with Tsang's definition; Hill only calls this methodology as *preventive or scheduled maintenance* (PM). He points out that the objective in *preventive maintenance* is to reduce the probability of breakdown by replacing worn components at set intervals. He continues that the replacement interval is usually based on the mean failure time of certain components.

The third approach by both Hill's and Tsang's classification is called *condition-based maintenance* (CBM), which is a more sophisticated way to preventive maintenance. Tsang states that in CBM the condition of the item is monitored continuously or intermittently to carry out PM actions only when failure is judged to be imminent. Thus replacing or servicing equipments prematurely can be avoided. Decision when the maintenance task is carried out is made based on condition monitoring techniques, as vibration monitoring, process-parameter monitoring, thermography, and tribology (Tsang, 1995). There are models available for optimizing replacement decisions. Hill as well emphasizes the need to acquire information about the condition of equipment under CBM. This information is used to identify instances where maintenance could be performed either earlier or later than the regular preventive maintenance schedule dictates. He adds that CBM can also be used to enable ongoing checks to be made on the effectiveness of a preventive maintenance schedule and thus enable it to be fine-tuned.

4.3 Managing product end-of-life

Product items for which it becomes more expensive to perform maintenance than to replace them with new ones enter the EOL phase of their lifecycle. No matter for what reason a product is taken out of use, it needs to be taken care of by a remanufacturing, refurbishing, recycling, reuse, or disposal process (Parlikad et al., 2003). These processes should be seen as an opportunity to enhance competitiveness and ultimately, create competitive advantage (Stock et al., 2002). Information can also be collected from used products (Krikke et al., 2003b), and remanufacturing end-of-use products has vital implications for the image of a company. In fact, goodwill building and the creation of a good corporate reputation are important sources of competitive advantage (Dowling, 2004; Hartshorn and Wheeler, 2002; Porter and Kramer, 2002).

4.4 Analysis of PROMISE application scenarios

The PROMISE application scenarios have initially been classified according to how they relate to the PLM phases (BOL/MOL/EOL) as indicated in Table 3. The fourth column of Table 3 indicates what product information needs to be accessed by whom in the scenario, as well as where the product information is assumed to be located as defined by the application scenario.

The fifth column indicates what on-line (or real-time) information enrichment procedures are needed in the scenario. An additional aspect has been added to the analysis in the fifth column, i.e. whether all processing (analysis, information enrichment etc.) is performed in one single backend system or if parts of the processing should occur locally in the PEID or on different computers of different organizations. The reason for analyzing this aspect is that when data processing is performed in one single backend system (*centralized*), product information and the data processing logic is often proprietary and held inaccessible to external actors. Therefore the representation issue for product information is often less relevant for the scope of this document in such scenarios than in scenarios that involve more *distributed* handling of dynamic product information.

Table 3: PROMISE application scenarios analysed according to needs for representation of static product information and on-line (i.e. real-time) enrichment of dynamic product information.

Demonstrator	WP	B/M/ EOL	Static product information	Online (i.e. not backend) processing of dynamic product information
CRF (EOL)	A1	EOL	Needs to be accessed by dismantler, on-line decision support processes are performed in backend system.	NA, dynamic product information is just retrieved at EOL and transmitted to backend system for decision-making purposes.
Caterpillar	A2	EOL	Needs to be accessed by dismantler, on-line decision support processes are performed in backend system.	NA, dynamic product information is just retrieved at EOL and transmitted to backend system for decision-making purposes.
INDYON	A3	EOL	Tacit knowledge of human operators (?), explicit knowledge coded in rule-based systems.	Expert knowledge coded in rule based systems and threshold methods shall support the decision on the milling process and the generation of the bin descriptors.
CRF (MOL)	A4	MOL	Static product model could be stored on the on-board diary. Prediction algorithm may be located both in on-board and in backend system. Current scenario situates it in the onboard system.	The information synthesis will be performed on the truck's on-board diary to display alerts on the dashboard in case of an upcoming failure. Summary statistics are also to be sent to a ground station periodically and will most probably be stored in a backend database as well.
Caterpillar	A5	MOL	Results from analysis of field data, need to be accessible to design, manufacturing and suppliers. Dealers, owners and operators should be informed about needs for preventive maintenance.	NA
FIDIA	A6	MOL	Component history needs to be maintained even though the component moves between different machines. The history contains both static and dynamic product information. Transfer of dynamic information should be periodic (e.g. every three months). Questions: 1) what data is transferred from/to backend/PEID/RFID? 2) is it the PC of the NC or some RFID tag that stores information about the installed components?	Reacting on abnormal values for sensor data (current, velocity) that correspond to degenerating components condition and can be used as indexes for the prediction of faults. Predictive Maintenance decision support software should run locally on the PC of the Numerical Control. This SW should: 1) perform suitable tests on the machine; 2) elaborate data and extract relevant parameters and; 3) make decisions based on these parameters.
MTS	A7	MOL	Predictive maintenance modules perform analysis on backend systems. Measurements and results of analysis modules are stored in database, can be visualized with Web-interface.	NA. Remark: Data flow from PEID to the PLM/PDKM/DSS: When/how often data must be send out will be decided with partners involved in predictive algorithms
Wrap	A8	MOL	Predictive maintenance modules perform analysis and decision support on backend systems.	NA. Remark: Defrosting cycle length can be varied when needed and specifically to production there is the need to make it the most flexible as possible. However, in the demonstrator this is not performed online.
INTRACOM	A9	MOL	Access by various actors (maintenance man, engineers, customer support etc.) to results of analysis, product data etc.	NA. Dynamic data is transferred to backend systems for analysis, no on-line processing defined in demonstrator.
Bombardier	A10	BOL	Both static and dynamic information in many different systems, difficult to	NA, performed by existing systems.

			access/use.	
Polimi	A11	BOL	?	NA. No real-time collecting of dynamic data involved.

For the current application scenarios it is for the moment not clear in what format product information is stored. It seems like most information would be stored in various software systems that mainly use proprietary formats. There seems to be two major possibilities to create access to such product information in a universal way. The first possibility is to re-encode existing product information using more universal standards such as RDF and OWL. In practice this tends to be a cumbersome task that few organizations are willing to undertake. The second possibility is to use a compatibility layer or middleware solution that converts information needed by external systems into universal standards.

As indicated by the analysis in the fifth column of Table 3, most information processing is performed in one single system. However, some application scenarios (e.g. A10) mention that the organization currently uses many different software products for storing product information, so the question how product information is represented and communicated is not only an inter-organizational issue. This is one of the assumptions made in this document, which are further analyzed in the next section.

5 Hypotheses and assumptions

The application scenarios analyzed in section 4.4 define a minimal set of functionality to achieve. The models and software to be developed should not only fulfill the requirements of these scenarios but rather a much more general set of requirements. This is necessary because the next generation of applications will be sources of much further going requirements. Experienced software engineers generally agree that the best way of meeting such changes to initial requirements is to first develop a general-purpose solution that solves a much greater task than the initial one. In addition to providing better possibilities to meet requirement changes, this approach also often leads to better software structure and thereby shorter development times even for the first version.

Our first hypothesis is therefore that we need to define a much more generic information management model than what is required by the current set of application scenarios. Therefore we will target an information management model that could also handle application scenarios with distributed data and processing over several PEIDs, computers and organisations.

5.1 Users

The three **EOL** scenarios (A1-A3) are quite different. A1 seems to be the most “typical” EOL scenario where a passenger car is taken out of use. Decision-making is mainly on what parts to re-use, remanufacture or dispose of. Users identified in A1 are 1) *last owner*; 2) *dismantler*; 3) *crusher*; 4) *shredder* and 5) *metallurgical industry*. The dismantler seems to be the main user of the IT system. The crusher, shredder and the metallurgical industry may also have need for information collected at EOL. The dismantler, crusher, shredder and the metallurgical industry may all be different organisations.

A3 treats the “shredder” part of A1 where plastic parts need to be shredded and routed depending on both BOL and MOL information. Identified users are 1) the *truck driver* and 2) the *decision maker* who decides on what category the shredded material belongs to. In practice, this scenario is quite close to A1. However, in order to be able to design a suitable IT system, the definition of the "decision maker" user would need to be more explicit.

A2 is a scenario where collected MOL data from a heavy vehicle needs to be accessed at EOL and transferred back to the producer in order to improve product design, manufacturing and logistics. There is also a possibility of deciding on remanufacturing parts of the vehicle based on MOL data, which seems to be quite close to the A1 scenario. In practice, the description of this scenario is still too general to make it possible to identify users on the level needed for defining end users and a suitable IT solution.

There are six **MOL** scenarios (A4-A9). The following users seem to be found in all these MOL scenarios: 1) *owners/users of product*; 2) *after sale service companies* and 3) *design department*, especially product designers. All of these can be identified with specific roles/employees in the companies concerned. Scenario A5 is not yet specific enough about these users but it can be assumed that they will be quite similar to those of the other scenarios.

The two **BOL** scenarios (A10, A11) attempt to use data collected during MOL end EOL phases to improve product design and manufacturing. Identified users in these two scenarios are 1) service organisation (inclusive commissioning) (remark: this seems to be more MOL than BOL); 2) product engineers and designers; 3) product operators and 4) information system engineers and maintenance personnel. Both scenarios would still need to be more detailed before a suitable IT system can be defined.

5.2 Environment

All application scenarios involve three levels: PEID, middleware and backend system. Concerning the PEIDs there are great differences between what the physical hardware looks like between the different application scenarios. In all application scenarios PEIDS transmit at least a unique product ID and collected data from sensors and other data sources towards backend systems using a middleware layer. There still seems to be great differences in what data needs to be transmitted and in what format.

Most of the representation and analysis issues addressed in this document will be performed in backend systems, leaving very little "intelligence" to the PEID. However, this cannot be a general assumption for the information management solutions to develop here. Even though only application scenario A4 explicitly states that product information and analysis algorithms may also be located in the PEID, it is easy to see that this could be the case for many other application scenarios at least in the near future. Finally, the application scenarios assume that the backend system just one computer or a set of computers controlled by one company. In practice, product information may be distributed over many different computers and organisations, which means that there may be a need for more communication interfaces than just the PEID to backend interface. For instance in application scenario A4, data is retrieved from the PEID and sent to the DSS system located in the manufacturer's backend system. The manufacturer's backend system then sends back decisions on how different parts should be handled. In the simplest case this information could be sent back as a human-readable web page. Another possibility is to write the result of the DSS into the RFID tag of every component before dismantling. In both cases, the results of the DSS would still normally need to be communicated also to the dismantler's information system because this information might be needed when a component is transferred to the next step after the dismantler.

5.3 Functionality

The provided functionality should be general enough to provide easy adaptability to the requirements of all application scenarios, similar tasks in other organisations and PLM contexts and additional or modified requirements in the future.

5.4 Constraints

All demonstrators for different application scenarios have to be implemented to work on the hardware defined in the application scenarios or at least demonstrated with a very similar set-up on hardware for demonstration purposes. Existing software platforms also have to be taken into consideration, notably existing PDM systems that may contain huge amounts of pre-existing data, information and knowledge.

5.5 Assumptions and dependencies

The analysis of the application scenarios is mainly based on the deliverable DR3.2 from PROMISE WP R3 entitled "*Application scenarios and demonstrators*". PROMISE WP R6 "*Fully Functional Device Controller*" is assumed to provide middleware definitions and infrastructure that supports transmitting data from a PEID to a backend system as well as writing information back from a backend system to a PEID.
This document should serve as input mainly to PROMISE WP R9 "*Development of PROMISE information management system*" but also partially to PROMISE WP R8 "*Methodologies for decision making for BOL, MOL, EOL*"

6 Approach to the development of the concept/method/model

From the state of the art section on existing PLM systems, it appears that there are many commercial products that fulfil all needs for document management (versioning etc.) for the needs of the BOL phase and also partially for the MOL phase. However, the functionality of these systems mainly seems to be guaranteed in the context of one single company or a group of tightly collaborating companies. From application scenario A11 it seems like it can also be challenging to use several PLM software systems together even within one single company. The STEP protocol makes it possible to exchange mainly CAD data and product structures represented in them between different software systems and organisations.

However, STEP is a protocol that is mainly used for information exchange between different BOL software products. It is not currently a protocol that would allow for easy set-up of connections with service companies or EOL actors who only need access to a limited part of the product information. Furthermore, protocols like STEP are not conceived for the transfer and treatment of dynamic field data.

This is why we propose extending existing systems with support for general semantic nets for representing product information using RDF and OWL structures. Access to this information will be provided by agent-based middleware, e.g. web services, WWAI, methods for the semantic web etc. Unique product identifiers are an essential part of the infrastructure, for which several coding schemes exist, e.g. EPC, SSCC, WWAI, ID@URI etc. that are able to complement each other.

We will shortly analyse how these different components interact in a few reduced application scenarios that can be envisaged in PROMISE:

Filtering RFID reader data. In many applications (e.g. warehouse applications), RFID readers continually try to read all tags in their vicinity. This means that they may do multiple reads of the same tag in a very short time. For "composite products" (Främling et al., 2004), i.e. a product that contains subassemblies that also have their own RFID tag, there may also be several reads of subassembly tag. It is usually not useful to send all these readings to backend systems because that would increase network traffic and load on the backend systems. Therefore RFID tag reads should be filtered and aggregated so that only relevant events are sent to backend systems. For composite product, this signifies that there should be a way of accessing the product structure as close to the reader as possible. With current PDM systems this might become very difficult because the RFID tag reader(s) and the PDM system might belong to completely different organisations.

Tracking scenario. In shipment tracking applications it is not usually interesting for the end user (e.g. project manager in construction project) to know where all shipments are located. This information becomes interesting only if an exception has occurred. This means that instead of sending all location updates to the user (for instance in GPS-based tracking this could mean very frequent updates), we only want to send a message if the shipment has some problem (being delayed, temperature too high/low, strong vibrations, …). Deciding if a shipment is late from its schedule requires knowing the shipment schedule so there needs to be a filtering "agent" that checks every incoming location update against the shipment schedule and generates user events if any exception is detected. Remark: this is rather a logistics scenario than a PROMISE scenario but it is used here because of its generality and simplicity.

Diagnostic scenario: The local diagnostic agent in a vehicle detects a problem but the locally stored product information and diagnostic rules are not sufficient to determine the reason for the problem. The agent informs the driver about the situation, who can take the vehicle to an authorized service provider. The vehicle's diagnostic agent then transmits all needed information to the service provider's or the manufacturer's diagnostic agent who returns a diagnostic of the problem. This scenario involves a PEID with an embedded agent, representation of structured product information and middleware for communication between agents.

In the next section we will study in detail the available components for this kind of scenarios as well as the PROMISE application scenarios.

7 Description of the proposed concept and related methods

In this section we will analyze what parts of the needed functionality can be achieved by existing software or other products from different PROMISE partners. However, the main goal of this section is to list and analyze different solutions and software that could be used for obtaining the possibly needed supplementary functionality. Solutions and software studied concern ways of representing and accessing structured product information in a distributed way and integration of decision support methods into a distributed system mainly through agent-based concepts. Section 7.4 shows how these elements could be combined to fulfil the requirements of PROMISE using one of the PROMISE application scenarios as an example.

7.1 Management of tacit knowledge

Following chapters are providing an overview on methodologies and technologies relevant for the management of tacit knowledge from the perspective of the PROMISE end users respectively the application scenarios as they are described in DR 3.1 and DR 3.2.

7.1.1 Product information portal

The success of Internet portals such as My Yahoo! has prompted vendors to market enterprise information portals (EIPs) to business users for accessing corporate business information. The promised benefits of EIPs are the same as those on the Internet - a simple Web interface that helps users rapidly sift through information managed by a large distributed computer network (Collin, 1999).

An Enterprise Information Portal (EIP) is strictly, an entry point (home page) into an organization's intranet, although the term now often refers to the intranet itself and its content. Users have a personalized starting page that gives them a single point of access to enterprise information, wherever it is held.

On the public Internet, an information portal employs a profile of the user's information requirements and the services of a search engine to help consumers quickly find information that matches their needs. An Internet portal provides the consumer with a single interface to the vast network of servers that constitute the Internet. Information portals in the corporate environment have a similar objective: to provide business users with a single interface to information scattered throughout the enterprise (Collin, 1999).

According to Eckerson (1999) there are four generations of portals, see Table 4. They are to be understood as layers built upon one and other. Without the first level it would be impossible to build the second.

Table 4: Generations of corporate portals (Eckerson, 1999).

Generation	Category	Corporate Portals
First	Referential	Search engine plus a hierarchical catalog of Web content. Each catalog entry contains a description of the content object and a link to it.
Second	Personalized	Users create personalized views of the portal contents, known as a “MyPage.” Their views show just the categories and applications they are interested in viewing. They can also publish documents to the corporate repository for viewing by others as well as subscribe to those documents.
Third	Interactive	The portal embeds applications to enhance personal and workgroup productivity. These include e-mail, calendars, workflow, project management, expense reports, travel, monitors of productivity indicators, etc.
Fourth	Specialized	Role -based portals for managing specific corporate functions. This involves integrated enterprise applications with the portal so that users can read/write/update corporate data.

The first generation stress: content management, mass dissemination of corporate information and decision support. The second is about customized distributed content. The third put emphasis on the collaborative features of the portal. The last generation talks about specific roles in the company and how to connect corporate applications with the portal.
In accordance to an EIP a Product Information Portal (PIP) is proposed as an enabling concept to increasing the effectiveness of product related process by implementing KM methodologies. Consequently the PIP shall provide mechanisms and tools for the management of product item related information including the capturing and sharing of product related tacit knowledge from the product centric viewpoint based on the concepts of an EIP as described above. Following graphic outlines the principle of the proposed PIP based on the SECI process provided by Nonaka and Takeuchi as described in chapter 3.2.1.1.

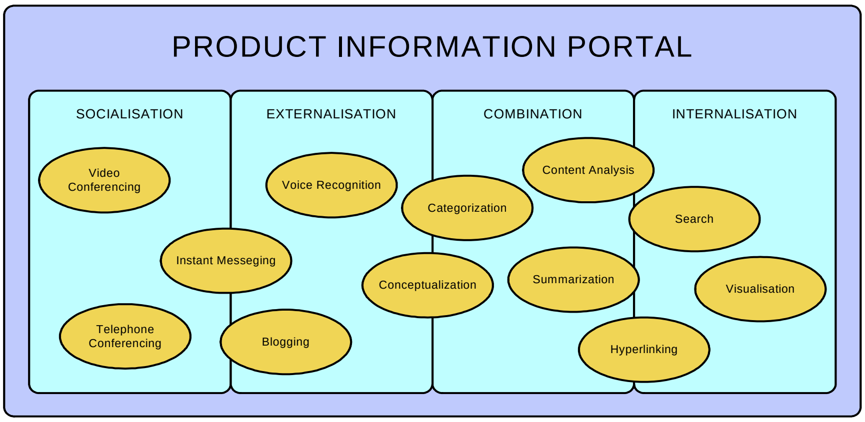


Figure 13: Conceptual framework of the Product Information Server

A successful knowledge management application needs to solve the problem of capturing and exploiting tacit knowledge. Current solutions that focus on structured data and semi-structured documents are limited to the knowledge that users are willing or able to enter in these forms (Brown et al, 2001). Considering the process of the knowledge externalisation where tacit knowledge needs to be articulated into explicit knowledge two major issues are significant. First one is the effort (in correlation with his current situation), which is required for an individual to articulate his knowledge; second one is the expressiveness of the resulting explicit knowledge.

Individual barriers for the externalisation of tacit knowledge can be reduced with decreasing the effort for the externalisation and/or creating sufficient ambient conditions. Due to the fact that PROMISE is not dealing with organisational methodologies to create environments or ambient conditions supporting the sharing of knowledge the latter one is not addressed in the following. The following will mainly address methodologies and technologies that are supporting the knowledge conversation and transportation with respect to the SECI processes by similarly decreasing the individual effort for sharing knowledge. Concerning the individual conversation processes of the SECI model the proposed concept can be described as follows:

Socialization: Provision of communication means such as phone or video conferencing in direct relation to the product relevant processes.

Externalization: Firstly externalization of tacit knowledge on the fly by the means of speech recognition applied to the communication means mentioned above. Secondly the provision of text based communication means allowing the stakeholders of a product to share knowledge in a common speech such as Blogging or forums also in direct relation to the product relevant processes.

Combination: Application of methods for the enrichment of rather unstructured externalized knowledge to increase the degree of reuse such as semantic annotations, ontologies and taxonomies, summarizing and categorization.

Internalization: Provision of “intelligent” and structured access for the stakeholders of a product to the existing knowledge.

Methods and concepts relevant for the concept are described in the following chapter.

7.1.2 Methods and technologies supporting the externalisation of tacit knowledge

7.1.2.1 Speech recognition (Externalisation)

Humans understand speech with ease and use speech to express complex ideas, information, and knowledge. A substantial amount of individual knowledge is conveyed in speech either in meetings, on the telephone, or in casual conversations.

Generally information of the spoken word can be captured and stored which is in a first step not more than data gathering. But in combination with techniques for extracting knowledge from the data these data can provide access to much more of the underlying knowledge that an organization or a team wants to preserve. Thus, the externalization process consists of firstly capturing the spoken word into textual information and then applying KM methodologies for the management of this explicit knowledge such document classification or clustering methodologies to this text.

Main objective for the utilization of the speech recognition technology is to reduce barriers in the externalization (tacit to explicit) processes of tacit knowledge.

Consequently utilizing information conveyed in the spoken word for the knowledge capturing and creation process in the PROMISE application scenario context requires

- the provision of sufficient communication means as mentioned above directly related to individual processes around the product
- the application of speech recognition technology to existing communication means such as telephony (e.g. via fixed telephone network or via VoIP) desktop conferencing, chatting etc. for externalizing the spoken word to textual information
- the application of technologies and concepts for processing resulting textual information to ensure the adequate preservation and effortless reuse of the externalized knowledge

7.1.2.2 K-Logs (Externalisation, Combination and Internalisation)

WeBlogs or in the context of KM so called K-Logs are an instrument for managing tacit knowledge. (Bausch et al., 2002) defines a K-Log as a tool for an expert or employee to publish insight, a point of view, links to resources, important documents and e-mails with annotation, and other thinking to an intranet where it can be archived, searched, and browsed.

K-Logs are essentially supporting the capturing of tacit knowledge mainly due to the rather poor formal structure which is required for entering information to a system. Major effects resulting from this are the following

- The effort for employees to share their knowledge decreases
- Existing barriers for the articulation of individual knowledge are shrinking

Consequently Cayser (Cayser, 2004) recapitulates that Blogging's greatest benefit is social, not technological. First, ease of use makes it likely that more people will publish and publish more often, and that more information will be communicated. The structure of the information is often different from more static home pages, more like online journals, or (at a higher level) a series of information “snippets”.

The technical aspects of Blogging which is the structured nature of RSS technology (Rich Site Summary (RSS 0.9x), RDF Site Summary (RSS 1.0), Really Simple Syndication (RSS 2.0)) allowing Bloggers and readers alike to integrate and search information feeds (such as BBC News).

Cayser summarized that based on the existing technology Blogging can provide the following capabilities:

- *Ease of use and capture.* Capturing knowledge should be easy at a useful level of detail while causing minimal disruption to users' normal activities;
- *Decentralized aggregation.* Though knowledge are likely to be scattered throughout an organization in a variety of locations and stored in a variety of formats, it should be possible to integrate them and perform some global search over the result;
- *Distributed knowledge.* Information consumers can add value by enriching information or knowledge at the point of use by, say, adding ratings, annotations, relationships, and categories;

In addition to this it should be pointed out that Blogging is not just providing a mechanism to share information and/or knowledge but also providing links to experts related to that knowledge.

But concerning the requirements for a decentralized, informal knowledge management (system) Cayser defined the following additional capabilities.

- *Flexible data model.* Knowledge items are polymorphic; depending on the task, people may reasonably want to capture email, Web pages, documents, text fragments, and images;
- *Extensible.* It should be possible (post hoc) to not only enrich knowledge but extend the knowledge data schema to model the changing world; and
- *Inferencing.* It should be possible to infer new metadata from old

In traditional Blogging, metadata is used only for headline syndication. Metadata is not extensible, not linked to a rich, flexible data model, and certainly not capable of supporting vocabulary mixing and inferencing. Metadata can be extended and is what the existing Blog standard RSS1.0 aims to do (web.resource.org/rss/1.0/).

Consequently Cayser proposed the enhancement of traditional Blogging with Semantic Web technology. Semantic Web is a common framework that allows data to be shared and reused across application, enterprise, and community boundaries; information is given well-defined meaning, better enabling computers and people to cooperate (Berners-Lee et al, 2001).

RSS1.0 is a Semantic Web vocabulary that provides ways to express and integrate with rich information models (web.resource.org/rss/1.0). The Semantic Web standard Resource Description Framework (RDF) specifies, in essence, a Web-scale information-modelling format (www.w3.org/RDF/). The key element in RDF is the triple—a simple subject-predicate-object construct—that can be joined to create a graph-like structure, with subjects (or objects) as their links, or arcs.

First demonstrators of the semantic Blogging concept, are implemented by the SWAD-Europe project (SWAD-Europe, 2001) in the test domain for the demonstrator we chose the problem of publishing bibliographic information and sharing it amongst small work-groups

Results of the first implementation are described as follows:

The notion of semantic Blogging builds upon this success and clear network value of Blogging by adding additional semantic structure to items shared over the Blog channels. In this way we add *significant value allowing view, navigation and query along semantic rather than simply chronological or serendipitous connections. Our semantic Blogging demonstrator design emphasises three key behaviours:*

- ***Semantic View**: Semantically enriched Blog metadata enables context sensitive, metadata driven views of the Blog content (over and above fixed templates).*
- ***Semantic Navigation**: Semantically enriched Blog metadata enables new Blog navigation modalities (over and above unlabelled links).*
- ***Semantic Query**: Semantically enriched Blog metadata enables richer query and discovery mechanisms (over and above free text search).*

We should note here that the use of the term 'semantic' emphasises the use of semantic web technology to enable these behaviours. It is true that in the current instantiation of the demonstrator, the new capabilities are enabled primarily by using rich metadata, and require little in the way of actual 'semantic' machinery. However, in each case it is easy to see how the behaviour can be further extended by adding inferencing over a semantic model. A simple example would be subcategory inferencing for semantic query. We prepare our Blog for such possibilities by encoding its metadata in RDF.

In summary then, the rich structure and query properties enabled by the semantic web greatly extends the range of Blogging behaviours, and allows the power of the metaphor to be applied in hitherto unexplored domains. One such domain (and the one explored in the demonstrator) is small-group management of bibliography data.

7.1.3 Methods and technologies supporting the processing of externalized tacit knowledge

Methods and technologies depicted above are mainly addressing the externalisation of tacit knowledge in the PROMISE context. By providing means which are allowing the externalization of tacit knowledge “on the fly” or with alleviated efforts individual barriers for the externalization of tacit knowledge can be reduced.

Due to the immense amount of rather unstructured information produced by this kind of application, the manual research and collection of relevant information is a time consuming challenge.

The chance to analyze all available data is clearly out of scope of human ability, thus smart analyzing tools are needed to bring some structure into the chaos of the immense information. Thus, the implementation of the above describe methodologies seems to be senseless without guiding concepts for analysing and synthesizing these information to enable the seamless reuse.

Adequate solution to get an uncomplicated and effective access to this captured information must be able to collect the data, sometimes from different sources, to classify it, to group it, to file it and to allow users to rummage in the data effectively.

Many corporations have recognized the need for the management of unstructured data. There are dozens of tools that claim to bring a structure into business data. In this chapter we will name and describe some of these approaches.

7.1.3.1 Search Engines

Currently the problem of unstructured data is bypassed by the generous exertion of search engines. This allows finding some information in the whole chaos, comparable to the search in a large pile of books for the right one. Understandable, this approach is not that efficient, in comparison to the search in a well sorted library. The use of search engines requires some well elected keywords to get the wanted results, which is the crucial factor for the quality of the results. People often have problems selecting the proper keywords, especially if they have no clear vision of what they are looking for. Also, keywords often have different meanings regarding of the context, which a search engine is capable to handle.

An advantage of Search Engines is that they are able to provide access to data, that other methods aren't capable to classify. And additionally, they are also good to use in already structured data. So in conclusion, one can say that Search Engines are the easiest way to provide access to unclassified data, but they are not able to classify unstructured data.

7.1.3.2 Hyperlinking

The use of Hyperlinks allows it for users quickly to jump between related data. An automated tool can recognize same keywords in different texts and link them together to show the relation. This method allows the user to get information about related data while reading or writing a text.

This method is comparable to the search engine, with the difference that the user does not need to explicitly search for keywords. Automatic Hyperlinking instead activates the search automatically.

7.1.3.3 Summarization

Automatic Summarization is used to provide a brief overview about a given text. In this approach, a tool parses the text and automatically creates an abstract describing the text and the topic. It helps to decide whether a text is useful or not without having to read the whole text. Although this method does not automatically structure the data, it helps users to classify it. Using this method, it is easier for users to characterize and to order the data. It reduces the time, which is needed to handle the information which speeds up business operations.

7.1.3.4 Ontology

Ontology deals with the being or existence of something. In the domain of Knowledge Management, it is a collection of representations of explicit knowledge, which can be used to recognize incoming information. Ontology can be used to assign a semantic to the components of a free-form text by mapping the components to the ontology-database.

Because some words have different meanings in different contexts, Ontology is often bound to a specific domain, that means, only the representations of a single domain are known. For example, the word "stream" has different meanings in geographical, technical, physical and computer-related domains. Covering only one specific domain helps to minimize the probability of misinterpretations. The use of ontologies helps to increase the quality of the methods mentioned above, namely Search Engines, Summarization and Hyperlinking. This appliance gives the computer some "smartness" to have a meaning of all the used keywords. An ontology-enhanced search engine for instace, could search also for related keywords and automated hyperlinking could link to similar keywords.

7.1.3.5 Web Ontology Language

The Web Ontology Language (shortened OWL) is a markup language for publishing and sharing data using ontologies.

The basic principle of OWL is based on the extension of content with additional information regarding machine readable semantics. While most web contents are smoothly readable for human, computers have some difficulties to recognize the meaning. For example, in the fragment "Miller keeps his home clean", it's not obvious for a computer, if Miller is the name of a person or if it is the profession of a person (or if it's a person at all). In the concept of OWL, the data is enriched by metadata, disclosing the type of data that the content provides.

Obviously, this additional data has to be provided with the content. The problem in this method is that it complicates the writing of a text, because the writer has to specify each significant word to eliminate all possible ambiguities. Using this approach, it is questionable if it would gain productiveness. It would be additional work for writers, especially, if the content is for communication with human partners and is not intended for computer readers. Indeed, every E-Mail and every report could be grouped and classified more efficiently by a computer, but it would not reduce work at all. Additionally, OWL is not wide used in the internet, which means, that most internet resources could not be classified automatically.

In conclusion, one can say, that OWL does not ease the work of classification or categorization of unstructured data; it simply reallocates the work to the writers. Anyway, OWL is a good way to label data with its semantic. For example, an automated semantic analysis algorithm could review the text and save the generated metadata directly to the content using OWL. Having this information right by the content, the other tools like search engines and hyperlink tools could work more efficiently.

7.1.3.6 Content Analysis

Originally, Content Analysis is a methodology that is used in social science. The aim of it is to ask some questions about the content to identify its properties. The advantage of Content Analysis is that it is capable to put the content into a context. It is fundamentally based on the idea that documents using the same vocabulary talk about the same topics.

The method of content analysis enables the researcher to include large amounts of textual information and identify systematically its properties, e.g. the frequencies of most used keywords (KWIC = *Keyword In Context*) by detecting the more important structures of its communication content. Yet such amounts of textual information must be categorized according to a certain theoretical framework, which will inform the data analysis, providing at the end a meaningful reading of content under scrutiny.

Different Content Analysis Tools vary in their analysis approaches. Simple tools just count the recurring words and ascribe the correlative importance to them. More sophisticated systems additionally count the joint frequency for word pairs and calculate standard frequencies. This information provides a way to compare contents, which allows the creation of groups and categories with different contents.

Source: http://en.wikipedia.org/wiki/Content_analysis

7.1.3.7 Categorization

Automatic Categorization is able to sort given data into different categories using pattern matching. Human intervention consists solely of the definition of the categories and patterns. In combination with data retrieval tools like data miners and intelligent agents, this method is able to collect important data and to store it in the appropriate database or to forward it to an interested user.

This approach searches available data, analyses it and stores it in a database for browsing. Additionally users can be notified if relevant information is found. This approach works the other way that a search engine. Instead of waiting for the user to give some keywords, the user waits for the Categorization tool to provide relevant data.

Powerful Categorization tools are able to resort already classified data, when categories are added or removed. In combination with Content Analysis, this method can create a hierarchical structure of the found topics.

7.2 Management of explicit knowledge

A requirements' analysis for software that would make it possible to manage a distributed semantic net is being developed at Helsinki University of Technology. The requirements analysis remains middleware-neutral enough to be applicable for the Dialog framework (section 3.5.3), WWAI and similar protocols. In the rest of this section we study existing software solutions for managing explicit knowledge based on semantic web (RDF, OWL) related concepts. The goal of this list is not to be complete but rather give an overview of the functionality provided by existing software. The list of software is sorted according to alphabetical order. All the software products listed here are distributed under public licenses that would allow them to be used in PROMISE at least for demonstration purposes (and usually also in commercial applications) without license fees. If one of several of these products is used in PROMISE the corresponding licenses should be studied more in detail.

7.2.1 JENA

Jena is a general-purpose Java framework that could be used in PROMISE for managing product information described using RDF/OWL. However, it seems like JENA would not as such provide any middleware layer that would make it easy to link to existing PDM system.

The JENA homepage gives the following description (http://jena.sourceforge.net/):

Jena is a Java framework for building Semantic Web applications. It provides a programmatic environment for RDF, RDFS and OWL, including a rule-based inference engine.

Jena is open source and grown out of work with the HP Labs Semantic Web Programme. The Jena Framework includes:

- *A RDF API*
- *Reading and writing RDF in RDF/XML, N3 and N-Triples*
- *An OWL API*
- *In-memory and persistent storage*
- *RDQL – a query language for RDF*

7.2.2 OpenRDF.org (Sesame)

The same comments apply concerning the usability of OpenRDF for PROMISE as for JENA. The documentation mentions a support for distributed models and access through Java RMI and HTTP protocols but it is not yet clear to what extent this functionality allows connecting with existing PDM systems.

The OpenRDF homepage gives the following description (http://openrdf.org/):

Sesame

Sesame is an open source RDF database with support for RDF Schema inferencing and querying. Originally, it was developed by Aduna (then known as Aidministrator) as a research prototype for the EU research project On-To-Knowledge. Now, it is further developed and maintained by

Aduna in cooperation with NLnet Foundation, developers from OntoText, and a number of volunteer developers who contribute ideas, bug reports and fixes.

Sesame has been designed with flexibility in mind. It can be deployed on top of a variety of storage systems (relational databases, in-memory, filesystems, keyword indexers, etc.), and offers a large scala of tools to developers to leverage the power of RDF and RDF Schema, such as a flexible access API, which supports both local and remote (through HTTP or RMI) access, and several query languages, of which SeRQL is the most powerful one.

Rio

Rio stands for RDF I/O. It is a set of parsers and writers for RDF that has been designed with speed and standards-compliance as the main concerns. Currently it supports reading and writing of RDF/XML and N-Triples, and writing of N3. Rio is part of Sesame, but can also be used as a separate tool, and can be downloaded separately.

7.2.3 Redland RDF Application Framework

The Redland RDF Application Framework is written using the "C" programming language which makes it less platform-independent than JENA and OpenRDF but may also offer some advantages in speed and memory usage. It is unclear to what extent Redland RDF supports distributed implementations. Compared to the Java-based frameworks it could also be more difficult to add such support. In PROMISE, Redland RDF would mainly be an alternative for applications where "C" or C++ programming languages have to be used.

The Redland RDF homepage gives the following description (http://librdf.org/):

Redland is a set of free software packages that provide support for the Resource Description Framework (RDF). Features:

- *Modular, object based libraries written in C.*
- *APIs for manipulating the RDF graph, triples, URIs and Literals.*
- *Triple sequences for efficient streaming.*
- *Parsers and Serializers for reading and writing RDF as RDF/XML, N-Triples and Turtle Terse RDF Triple Language syntaxes via the Raptor RDF Parser Toolkit.*
- *Storage for graphs in memory, with Sleepycat/Berkeley DB, MySQL 3/4, AKT Triplestore, SQLite, files or URIs.*
- *Querying with RDQL and SPARQL using the Rasqal RDF Query Library.*
- *Redland contexts for managing data aggregation and recording provenance.*
- *Language Bindings in C#, Java, Obj-C, Perl, PHP, Python, Ruby and Tcl via the Redland Bindings package.*
- *Command line utility programs rdfproc (RDF), rapper (parsing) and roqet (query).*
- *Portable, fast and with no known memory leaks.*

The packages that form Redland are:

- *Raptor RDF Parser Toolkit for parsing and serializing RDF syntaxes (RDF/XML, N-Triples, Turtle and RSS tag soup)*
- *Rasqal RDF Query Library for executing RDF queries with RDQL and SPARQL. Requires: Raptor.*

- *Redland RDF Application Framework providing the C RDF API. Requires: Raptor and Rasqal*
- *Redland Language Bindings for APIs to Redland in C#, Java, Obj-C, Perl, PHP, Python, Ruby and Tcl. Requires: Redland.*

These are mature RDF packages developed since 2000 used in several projects. Each library has it's own documentation and status information in the form of a list of TODOs/bugs, general news, detailed release notes and file-by-file changes in the ChangeLog.

7.3 Managing product-related knowledge and information in a multi-organisational context

Section 7.1 mainly listed tools for describing RDF/OWL-based product information and product structures. Those tools are not designed to be used as software protocols for distributed applications. Therefore their main use in PROMISE might be providing class frameworks for accessing and managing RDF/OWL information as parts of the PROMISE PDKM system.

In this section we will study two agent frameworks (ABLE and JADE) that integrate inter-organizational communication using several different protocols. As for most agent frameworks, they also provide modules for data analysis and decision support. Other agent frameworks exist but the two presented here seem to be the most used ones or otherwise appropriate for the need of PROMISE. This is the case especially for the decision support modules in ABLE. The last sub-section presents a simple rule base that addresses the MOL application scenario A4 (estimating number of kilometers to run before next oil change).

7.3.1 Agent Building and Learning Environment (ABLE)

ABLE is a collection of Java class libraries and programs created by IBM (Bigus et al., 2002) that is freely accessible at least for development purposes. ABLE supports inter-agent communication by different protocols (e.g. FIPA, web services etc.) natively or through supplementary modules and could therefore be an interesting alternative for inter-organizational communication in PROMISE. The data analysis and decision support classes and agents provide support for practically all the methods described in section 3.4, which is particularly interesting for PROMISE scenarios that include diagnostics/prognostics. These agents can be trained both on- and offline and included in different software components to perform filtering or decision making on different levels.

The ABLE homepage gives the following description (http://www.alphaworks.ibm.com/tech/able):

ABLE is a Java framework, component library, and productivity tool kit for building intelligent agents using machine learning and reasoning. The ABLE research project is made available by the IBM T. J. Watson Research Center.

The ABLE framework provides a set of Java interfaces and base classes used to build a library of JavaBeans called AbleBeans. The library includes AbleBeans for reading and writing text and database data, for data transformation and scaling, for rule-based inferencing using Boolean and fuzzy logic, and for machine learning techniques such as neural networks, Bayesian classifiers, and decision trees. Developers can extend the provided AbleBeans or implement their own custom algorithms. Rule sets created using the ABLE Rule Language can be used by any of the provided inferencing engines, which range from simple if-then scripting to light-weight inferencing to heavy-weight AI algorithms using pattern matching and unification. Java objects can be created *and manipulated using ABLE rules. User-defined functions can be invoked from rules to enable external data to be read and actions to be invoked.*

<u>How does it work?</u>

Core beans may be combined to create function-specific JavaBeans called AbleAgents. Developers can implement their own AbleBeans and AbleAgents and plug them into ABLE's Agent Editor. Graphical and text inspectors are provided in the Agent Editor so that bean input, properties, and output can be viewed as machine learning progresses or as values change in response to methods invoked in the interactive development environment.

Application-level agents can be constructed from AbleBean and AbleAgent components using the ABLE Agent Editor or a commercial bean builder environment. AbleBeans can be called directly from applications or can run autonomously on their own thread. Events can be used to pass data or invoke methods and can be processed in a synchronous or asynchronous manner.

The distributed AbleBeans and AbleAgents are as follows:

Data beans

AbleImport reads data from flat text files.

AbleDBImport reads data from SQL databases.

AbleFilter filters, transforms, and scales data using translate template specifications.

AbleExport and AbleDBExport write data to flat text files and SQL databases.

AbleTimeSeriesFilter collects periods of data for use in predicting future values.

Learning beans

Back Propagation implements enhanced back propagation algorithm used for classification and prediction.

Decision tree creates a decision tree for classification.

Naive Bayes learns a probabalistic model for classification.

Radial Basis Function uses radial basis functions to adjust weights in a single, hidden-layer neural network for prediction.

Self-Organizing Map clusters data using Gaussian neighborhood function.

Temporal Difference Learning uses reinforcement learning for time series forecasting; gradient descent is used to adjust network weights.

Rules beans inferencing engines include

Backward chaining

Forward chaining

Forward chaining with working memory

Forward chaining with working memory and Rete'-based pattern matching

Planning

Predicate logic

Fuzzy logic

Script

Agents

Genetic search manipulates a population of genetic objects which may include AbleBeans.

Neural classifier uses back propagation to classify data.

Neural clustering uses self-organizing maps to segment data.

Neural prediction uses back propagation to build regression models.

Rule agent contains a rule set whose rule blocks define its init, process, and timer actions

Script uses rule sets to define its init, process, and timer actions.

JavaScript names JavaScripts to run when the agent's init, process, or time actions are called.

7.3.2 Java Agent Development Framework (JADE)

The main advantage of the JADE framework is that it is planned for inter-agent negotiations using the FIPA agent communication standard. It also includes possibilities to use web service protocols. It is unclear whether such agent communication will be necessary or useful in the PROMISE project but JADE might provide partial solutions or models for constructing software solutions in PROMISE.

The ABLE homepage gives the following description (http://jade.tilab.com/):

JADE is a software framework fully implemented in Java language. It simplifies the implementation of multi-agent systems through a middle-ware that complies with the FIPA specifications and through a set of graphical tools that supports the debugging and deployment phases. The agent platform can be distributed across machines (which not even need to share the same OS) and the configuration can be controlled via a remote GUI. The configuration can be even changed at run-time by moving agents from one machine to another one, as and when required. JADE is completely implemented in Java language and the minimal system requirement is the version 1.4 of JAVA (the run time environment or the JDK).

The synergy between the JADE platform and the LEAP libraries allows obtaining a FIPA-compliant agent platform with reduced footprint and compatibility with mobile Java environments down to J2ME-CLDC MIDP 1.0. The LEAP libraries have been developed with the collaboration of the LEAP project and can be downloaded as an add-on of JADE from this same Web site.

JADE is free software and is distributed by TILAB, the copyright holder, in open source software under the terms of the LGPL (Lesser General Public License Version 2). Since May 2003, a JADE Board has been created that supervisions the management of the JADE Project. Currently the JADE Board lists 5 members: TILAB, Motorola, Whitestein Technologies AG., Profactor GmbH, and France Telecom R&D.

7.4 Simple example developed for PROMISE application scenario A4

PROMISE application scenario A4 is a MOL scenario defined by CRF. One of the functionalities to develop in A4 is to estimate the remaining number of kilometers before it becomes necessary to change the engine oil. We will here use this scenario A4 as a basis for a developing a very reduced demonstrator that will most probably differ from the final demonstrator in many aspects.

In A4, the *engine control unit* (ECU) receives from sensors and processes relevant information on the vehicle to estimate the remaining number of kilometers before oil substitution. The ECU then transmits this information to the manufacturer's or the service company's PDKM that integrates it with existing knowledge to decide whether to change the engine oil immediately, schedule single or fleet maintenance, or what mission the vehicle should be assigned to as a function of the length of the mission.

Traditionally, the connection with the ECU is normally established by a special terminal that knows how to communicate with the ECU (e.g. by the ODB-II protocol). The terminal can then establish a connection over the Internet (wireless or not) with the PDKM. In A4 it is assumed that this connection can be taken wirelessly directly by the ECU but this is irrelevant for the demonstration developed here. The vehicle information can be sent using standard messaging protocols such as those supported by the Dialog middleware developed at HUT (see section 3.5.3). The same can be done for the response that indicates the remaining number of kilometers before it becomes necessary to change the engine oil. An extension to the existing Dialog system is under development that would make it easy to plug in different handlers for such application-specific messages. One such plug-in could be an ABLE agent.

A small rule base that uses fuzzy logic was developed as an initial demonstration of how the reasoning could be performed in this case (Figure 14). Most modern ECUs measure the load on the engine at every moment for control purposes. By integrating this measurement over time it could be possible to obtain an indication of the engine load and, implicitly, also engine oil wear. The rule base only uses this indicator of the engine load and the number of kilometers since the last engine oil change for estimating the remaining number of kilometers. In practice, this estimation uses much more information (e.g. RPM, fuel consumption, trip duration, number of engine start-ups, oil temperature, water temperature, engine working hours, engine age, boost pressure etc). During the creation of this rule base it also became clear that it is difficult to produce good fuzzy sets and rules even for this simple case. Therefore some kind of learning mechanism, e.g. neural networks, would probably be more appropriate for the real application. When using ABLE agents it is relatively easy to change the DSS being used just by changing the agent.

```
/**
 * This ABLE ruleset determines, using fuzzy logic, the remaining number
 * of kilometers to go before the next oil change for an IVECO truck.
 *
 * These rules are created only for demonstration purposes and do not
 * correspond to real-life decision rules. A default oil-change interval
 * of 20000 kms has been used here, which is valid for most cars. For
 * trucks the oil change interval could even range from 80000 to 300000
 * kilometers depending on the driving conditions of the truck.
 */
ruleset KmsToOilChange {

  variables {

    /**
     * NbrKilometersSinceOilChange is an input variable.
     *
     * This is the most important variable for determining the need
     * for oil change.
     */
    Fuzzy NbrKilometersSinceOilChange = new Fuzzy(0, 1000000) {
      Linear   Low    = new Linear  ( 0, 15000, ARL.Down);
      Triangle Medium = new Triangle( 0, 10000, 20000 );
      Shoulder High   = new Shoulder( 10000, 20000, ARL.Right  );
    };

    /**
     * LoadLevel is an input variable.
     *
```

```
     * It is indicated as a percentage in the range 0-100 where 0
     * signifies very low load since last oil change and 100 a
     * continuously high load since last oil change.
     */
    Fuzzy LoadLevel = new Fuzzy(0, 100) {
      Linear   Low    = new Linear  ( 0, 50, ARL.Down);
      Triangle Medium = new Triangle( 0, 50, 100      );
      Linear   High   = new Linear  (50, 100, ARL.Up  );
    };

    /**
     * KilometersToGo is an output variable.
     *
     * Many kilometers and high load means quick oil change, few
     * kilometers and low load means lots of kilometers to go.
     * We assume 20000 kms is the longest possible number of kms to
     * go.
     */
    Fuzzy KilometersToGo = new Fuzzy(0, 20000) {
      Linear   VeryLow= new Linear  (0, 1000, ARL.Down);
      Triangle Low    = new Triangle(0, 1000, 3000);
      Triangle Average= new Triangle(1000, 3000, 10000    );
      Linear   High   = new Linear  (3000, 20000, ARL.Up  );
    };

 }

  // inputs { }; // Use when testing in rule editor
  inputs { NbrKilometersSinceOilChange, LoadLevel };
  outputs{ KilometersToGo };

  /**
   * ================================================================
   * This rule block receives control when some other Java program
   * wants to determine the interest rate for a loan amount requested
   * by a specific customer.
   * ================================================================
   */
  void process() using Script { // this.name parses but causes runtime error
    : println("Ruleset <" + this.getName() + "> starting process cycle.");

    // These are used only for testing in rule editor
    //: NbrKilometersSinceOilChange = 0;
    //: LoadLevel = 90;

    : invokeRuleBlock("DetermineKilometersToGo");
  }


  void postProcess() using Script {
    : println("   ...defuzzified remaining number of kilometers is  <" +
KilometersToGo + ">.");
    : println("Ruleset <" + this.getName() + "> completed process cycle.\n");
  }

  /**
   * ================================================================
   * Rules for determining number of kilometers to go.
   * ================================================================
   */
  void DetermineKilometersToGo() using Fuzzy {
```

```
    absoluteLimit:
      if ( NbrKilometersSinceOilChange > 25000 ) then
        KilometersToGo is extremely extremely VeryLow;

    highKmsLarge:
      if ( NbrKilometersSinceOilChange is High and LoadLevel is High ) then
        KilometersToGo is extremely extremely VeryLow;
    highKmsMedium:
      if ( NbrKilometersSinceOilChange is High and LoadLevel is Medium ) then
        KilometersToGo is VeryLow;
    highKmsSmall:
      if ( NbrKilometersSinceOilChange is High and LoadLevel is Low ) then
        KilometersToGo is Low;

    mediumKmsLarge:
      if ( NbrKilometersSinceOilChange is Medium and LoadLevel is High ) then
        KilometersToGo is VeryLow;
    mediumKmsMedium:
      if ( NbrKilometersSinceOilChange is Medium and LoadLevel is Medium )
then
        KilometersToGo is Average;
    mediumKmsSmall:
      if ( NbrKilometersSinceOilChange is Medium and LoadLevel is Low ) then
        KilometersToGo is above Average;

    lowKmsLarge:
      if ( NbrKilometersSinceOilChange is Low and LoadLevel is High ) then
        KilometersToGo is somewhat High;
    lowKmsMedium:
      if ( NbrKilometersSinceOilChange is Low and LoadLevel is Medium ) then
        KilometersToGo is High;
    lowKmsSmall:
      if ( NbrKilometersSinceOilChange is Low and LoadLevel is Low ) then
        KilometersToGo is extremely extremely High;

  }
}
```

Figure 14. Fuzzy rule base developed at HUT using ARL (ABLE rule language) for A4 scenario.

Figure 15 shows a short Java program for using an ARL rule base for estimating remaining number of kilometres before oil change. If a neural network should be used instead of a fuzzy rule base, it is sufficient to replace the three first lines in boldface with code for loading an agent with a trained neural net instead.

```
//======================================================================
// Imports
//======================================================================
import java.rmi.RemoteException;
import java.util.Iterator;
import java.util.Vector;

import com.ibm.able.Able;
import com.ibm.able.AbleBean;
import com.ibm.able.AbleDataSource;
import com.ibm.able.AbleDefaultAgent;
import com.ibm.able.AbleObject;
import com.ibm.able.rules.AbleRuleSet;
```

```
/**
 * This class shows how to call a rule base in the ABLE framework for
 * estimating the remaining number of kilometers until the next oil change.
 */
public class CRF_MOL_Main {


    public static void main(String[] args) {
      try {
          AbleObject agent = null;

          // Default rule set.
          String rules = "KmsToOilChange.arl";

          // Default input values. We use double[] buffer because all
          // values are numeric, otherwise use String[][].
          String[][] data  = new String[][] {{"0.0", "0.0"}};

          // Name of rule set can also be given as first parameter.
          if ( args != null ) {
            if ( args.length > 0 ) {
                int argind = 0;
                // Get rule base name if applicable
                if ( args[argind].endsWith(".arl") ) {
                  rules = args[0];
                  argind++;
                }

                // Get input values
                for ( int i = 0 ; argind < args.length && i < data[0].length ;
i++, argind++ )
                  data[0][i] = args[argind];
            }
          }

          // Get rule set, create agent
          agent = new AbleRuleSet();
          ((AbleRuleSet) agent).parseFromARL(rules);
          agent.init();

          // pass the agent data and get the result
          Object[] output = (Object[]) agent.process(data[0]);

          // show the input that was passed and the result obtained
          Able.setTraceConsoleHandlerLevel(Able.TRC_LOW);
          Able.TraceLog.text(Able.TRC_LOW,"Input  result: " +
displayBuffer(data[0]));
          Able.TraceLog.text(Able.TRC_LOW,"Output result: " +
displayBuffer(output));
          Able.setTraceConsoleHandlerLevel(Able.TRC_NONE);

      } catch (Exception exp) {
          System.out.println(exp);
          exp.printStackTrace();
      }
      System.exit(0);
    }

    private static String displayBuffer(Object[] buf) {
      StringBuffer buffer = new StringBuffer("");
      if (buf == null) {
```

```
        buffer.append(buf);
    } else {
        for (int i = 0; i < buf.length; i++) buffer.append(buf[i] + " ");
    }
    return buffer.toString();
  }

  private static String displayBuffer(double[] buf) {
    StringBuffer buffer = new StringBuffer("");
    if (buf == null) {
        buffer.append(buf);
    } else {
        for (int i = 0; i < buf.length; i++) buffer.append(buf[i] + " ");
    }
    return buffer.toString();
  }

}
```

Figure 15. Java code for loading an ARL rule base and running the inference engine for obtaining an estimate of the remaining number of kilometres before it becomes necessary to change the engine oil. The lines that create the agent, load the rule base and run the inference engine are indicated in bold. The rest of the program is for handling program input/output.

8 Comparison with state of the art, use and limitations of the proposed concept/method/model

8.1 Management of tacit knowledge

The effective management of tacit knowledge requires the consideration of KM processes as a whole. For the management of tacit knowledge in the context of the PROMISE project the SECI model was utilized which is considered as state of the art in the KM domain. Major challenges were resulting from applying this KM model developed in an organizational context to the inter-organizational context, which is mainly given by the application scenarios.

Thus, the various actors or stakeholders of the product needs to be integrated to product related processes and supporting systems independently from organizational borders in order to close the information flow for the entire product life cycle.

In the literature various concepts, methods and technologies are proposed to support he four layers of knowledge conversion given in the SECI model. But again these concepts are coming from the organisational perspective. Approaches for the management of tacit knowledge in the PROMISE application scenarios are requiring a product-centric and inter-organisational perspective.

The proposed Product Information Portal (PIP) as a platform for communication and management of rather unstructured product related knowledge provides this perspective to all stakeholders of a product. Based on the PIP concept existing state of the art methodologies and / or technologies can be applied to support the processes as described in the application scenarios.

Although there are already existing approaches for product specific and inter-organizational information systems the proposed approaches needs to be considered as clearly beyond the current state of the art because of the following reasons.

The proposed approach is joining the application area of product information management consequently with a structured approach for the management of tacit knowledge to support processes of the products life-cycle. Contrary to an organizational approach where employees can be forced to enter data into a system the PROMISE approach provides a solution which is minimizing the barriers for sharing knowledge aiming at the consequential utilization of the tacit knowledge which resides in the head of the product stakeholders.

8.2 Management of explicit knowledge

The PROMISE application scenarios represent major challenges for existing information management tools to store, communicate and use explicit knowledge during the product lifecycle. Current centralized information management tools might be sufficient for some of the current application scenarios but would give great problems when adding new requirements. Especially requirements concerning distributed information management in embedded devices or in different organizations would be problematic.

The following main concepts and technologies are proposed as a solution to these requirements:

- Light-weight "intelligent" agent technology
- Semantic nets
- Peer-to-peer communication between agents

Such concepts and technologies have already been implemented and used by some partners of the PROMISE consortium. Many of those implementations can as such be considered as "beyond state-of-the-art". The combination of these concepts and technologies for the PROMISE framework will go even further beyond the state-of-the-art.

It should also be pointed out that the proposed concepts and technologies are more powerful than what would be strictly necessary for fulfilling the requirements of most current PROMISE application scenarios. However, implementing the application scenarios with these concepts and technologies should require a smaller total effort than doing it with current state-of-the-art technology. For future requirements on product lifecycle management, the advantage of using the concepts and technologies proposed here should become even bigger.

9 Needed changes/amendments to Requirements and Specifications

The application scenario descriptions that have been used here mainly come from PROMISE deliverable DR3.2. When analyzing the application scenarios for the purposes of this document, the following was noticed:

- A2 is not specific enough for defining IT system needed.
- A3 definition of the "decision maker" user would need to be more explicit.
- A10 would still need to be more detailed about users before a suitable IT system can be defined.
- A11 would still need to be more detailed about users before a suitable IT system can be defined.

When developing the example for application scenario A4 in section 7.4, it was also noticed that there is a further need for interaction with the end users in order to develop the final PROMISE demonstrators. Even though the main ideas of A4 are well defined, there is for instance still a need to specify the exact results that should be provided by the PDKM and what kind if decision making methods are to be used. However, these detailed specifications will be addressed in other PROMISE work packages.

10 Conclusions

This document aims at proposing concepts and related technologies for the enhancement of product knowledge with respect to the demands of the PROMISE application scenarios and demonstrators.

In a first step we introduced the concepts of tacit and explicit knowledge management for a further structuring of relevant issues and classified then the types of static and dynamic product data to refine the scope of work of the task TR 7.2.

Within a next step we presented existing approaches to the representation, access and enrichment of product information. Due to the nature of the subject of interest we treated from there on the issues of tacit and explicit knowledge management concepts and methods in different sections.

For the utilization of tacit knowledge in the context of the PROMISE application scenarios the SECI model (Nonaka and Takeuchi, 1995) was employed to classify relevant KM methods and concepts. Major challenges were resulting from applying both the KM model itself and related methods and concepts to the inter-organizational context that is mainly given by the application scenarios.

The overall concept and associated methods and technologies proposed in section 7 is consequently addressing the conditions in an inter-organizational working environment with “loosely linked” actors. In accordance with the existing Enterprise Information Portal concept we proposed the Product Information Portal (PIP) as a common platform for the stakeholders of a product. With reference to the SECI model and the four layers of knowledge conversation the proposed concept is addressing the following aspects

Socialization: The PIP platform shall provide communication means such as phone or video conferencing in direct relation to the product relevant processes. This provides a means for direct exchange of tacit knowledge.

Externalization: The strategy for the externalization of tacit knowledge is twofold. Firstly externalization on the fly by the means of speech recognition applied to the communication means mentioned above. Secondly the provision of text based communication means such as Blogging or forums that are allowing the stakeholders of a product to share knowledge that is articulated in a common speech. These communication means shall also be direct relation to product relevant processes.

Combination: Due to the fact that above depicted means for the externalization of tacit knowledge are resulting in moderately structured information the application of methods is required for processing this information to ensure the reusability of it. Proposed means are either directly implemented in the externalization method (e.g. semantic Blogging) or afterwards applied to the results of the externalization methods.

Internalization: Provision of “intelligent”, structured and user specific access for the stakeholders of a product to the externalized knowledge.

Although single methods contained in the proposed concept have to be considered as state of the art it is noticeably that the combination of methods and technologies for the systematically support of the four layers of knowledge conversation in the focused application area goes clearly beyond the current state of the art.

For representing and accessing explicit knowledge in the context of the PROMISE application scenarios, we proposed extending existing PDM models in order to support a more multi-organizational and distributed concept than current PDM systems do. The main building blocks of this distributed concept are semantic nets and distributed (intelligent) agent systems. Ongoing research and development efforts on these two building blocks provide sufficient proof-of-concept of their usability. The challenge of PROMISE is to study if and how these concepts can be applied to PLM.

In this document, a common goal both for the management of both tacit and explicit knowledge is to provide a rather complete model solution that is both generic and covers the needs of PROMISE application scenarios. However, this document should not be taken as a definition of the future "PROMISE knowledge management solution" but rather as a basis for making a realistic requirement analysis for it.

11 References

Anderson, J. R., Bower, G. H. (1973). Human associative memory. Washington, DC: Winston and Sons.

Allan, J. 2001. Perspectives on Information Retrieval and Speech. Lecture Notes on Computer Science. Springer Verlag. Forthcoming.

Angeles, R., Nath, R. (2001). Partner congruence in electronic data interchange (EDI)-enabled relationships. Journal of Business Logistics, Vol. 22, No. 2. pp. 109-128.

Berners-Lee, T., Hendler, J., Lassila, O. (2001). The Semantic Web. Scientific American, May 2001. Available online (May 31st, 2005): http://www.scientificamerican.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&catID=2

Beulens, A.J.M, Jansen, M.H., Wortmann, J.C. (1999), The information de-coupling point. In: Global Production Management, IFIP WG5.7, Int. Conf. on Advances in Production Management Systems (Kluwer Academic Publishers, Boston). pp. 51-58.

Bigus, J. P., Schlosnagle, D. A., Pilgrim, J. R. , Mills III, W. N., Diao, Y. (2002). ABLE: A toolkit for building multiagent autonomic systems. IBM Systems Journal, Vol. 41, No. 3. pp. 350-371.

Brown, E., Srinivasan, S., Coden A., Ponceleor, D., Cooper, J.,Ami, A., Piepe, J. (2001) Towards Speech as a Knowledge Resource, (Abstracted, with permission, from the full paper published in the IBM Systems Journal 40, No. 4, (2001))

Bausch et al. (2002). Using Blogs in Business, Chapter 8 in: We Blog: Publishing Online with WeBlogs, John Wiley & Sons

Cayzer, S (2001). Semantic Blogging: Lessons Learnt. Tech. Rep. SWAD-E 12.1.8, Sept. 2003; Available online (May 31st, 2005): www.w3.org/2001/sw/Europe/reports/demo-lessons-report/

Cayser, S., Semantic Web research group at Hewlett-Packard Laboratories, Bristol, U.K (2004). Communications of the ACM,Volume 47 , Issue 12 SPECIAL ISSUE: The Blogosphere, Pages: 47 – 52, ISSN:0001-0782, ACM Press New York, NY, USA Available online (May 31st, 2005): http://delivery.acm.org/10.1145/1040000/1035164/p47-cayzer.pdf?key1=1035164&key2=0810603211&coll=GUIDE&dl=GUIDE&CFID=51501796&CFTOKEN=11407816

Clark, D. (2005). The Continuum of Understanding. Available online (June 30th, 2005): http://www.nwlink.com/ ~donclark/performance/understanding.html.

Colin W. (1999).Using Information Portals in the Enterprise. Article published in DM Review Magazine, April 1999

Dowling, G.R. (2004). Corporate reputations: should you compete on yours? In: California Management Review, 46. Jahrgang, Heft 3. pp. 19-36.

EAN International (2001). Global Trade Item Numbers (GTIN), Application Guideline. EAN International, available online (January 8th, 2002): http://www.ean-int.org/

Eckerson Wayne (1999). Plumtree blossoms: New version fulfils enterprise portal requirements. Patricia Seybold Group, Boston, M.A.

EKMF– European KM Forum(2002): D1.1 - KM Framework. Available online (July 2005): http://www.knowledgeboard.com/library/deliverables/ekmf_d11_v07_2002_03_07_iao.pdf

McFarlane, D., Sarma, S., Chirn, J.-L., Wong C.Y., and Ashton, K. (2002). The Intelligent Product in Manufacturing Control and Management. In: Proceedings of IFAC World Congress, Barcelona.

Främling, K. (1996). Modélisation et apprentissage des péférences par réseaux de neurones pour l'aide à la décision multicritère. PhD thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne, France. 304 p.

Främling, K. (2002). Tracking of material flow by an Internet-based product data management system (in Finnish: Tavaravirran seuranta osana Internet-pohjaista tuotetiedon hallintaa), Tieke EDISTY magazine, No. 1, 2002 (Tieke: Finnish Information Society Development Centre, Finland).

Främling, K., Kärkkäinen, M., Ala-Risku, T., Holmström, J. (2004). Managing Product Information in Supplier Networks by Object Oriented Programming Concepts. In: Taisch, Marco, Filos, Erastos, Garello, Paolo, Lewis, Kevin, Montorio, Marco (eds.) *Proceedings of IMS International Forum* , Cernobbio, Italy, 17-19 May 2004. pp. 1424-1431.

Gamma, E., Helm, R., Johnson, R., Vlissides, J. (1995). Design Patterns: elements of reusable object-oriented software. Addison-Wesley, Reading, Massachusetts.

Hamilton, J. (2001). The European Union's consumer guarantees directive. Journal of Public Policy & Marketing, Vol. 20, No. 2. pp. 289-296.

M. Hardwick (2004). On STEP-NC and the complexities of product data integration. ACM/ASME Transactions on Computing and Information Science in Engineering", Vol.4, No. 1.

Hartshorn, J., Wheeler, D. (2002). Facilitating strategic business responses to sustainability. In: Greener Management International, 40. Jahrgang, Winter 2002. pp. 107-119.

Terry Hill (2000). Operations management: strategic context and managerial analysis. Basingstoke: MacMillan Business. ISBN 0-333-77592-9.

Gourlay, S. (2002). Tacit Knowledge, Tacit Knowing or Behaving? Paper presented at the Third European Conference on Organizational Knowledge, Learning, and Capabilities, Athens, Greece, April 2002. Available online (July 2005): http://www.alba.edu.gr/OKLC2002/Proceedings/pdf_files/ID269.pdf

Huvio, E., Grönvall, J., Främling, K. (2002). Tracking and tracing parcels using a distributed computing approach. In: Solem, Olav, ed., Proceedings of the 14th Annual Conference for Nordic Researchers in Logistics (NOFOMA'2002), Trondheim, Norway, 12-14 June 2002. pp. 29-43.

Ives, B., Vitale, M. (1988). After the Sale: Leveraging Maintenance with Information Technology. MIS Quarterly & The Society for Information Mgt. Vol. 12, No. 6. pp. 7-21.

Johnston, R. B., Yap, A. K. C. (1998). Two-Dimensional Bar Code as a Medium for Electronic Data Interchange. International Journal of Electronic Commerce, Vol. 3, No.1. pp. 86-101.

Krikke, H., Kokkinaki, A., van Nunen, J. (2003b). Information technology in closed loop supply chains. In: Guide, D. and van Wassenhove, L. (eds.): Business Aspects of Closed-Loop Supply Chains, Carnegie Mellon University Press, Pittsburgh, 2003.

Kärkkäinen, M., Ala-Risku, T. (2003). Facilitating the integration of SME's to supply networks with lean IT solutions. In: eChallenges e-2003 conference proceedings, 22-24 October, Bologna, Italy.

Kärkkäinen, M., Ala-Risku, T., Främling, K. (2003). The product centric approach: a solution to supply network information management problems?. Computers in Industry, Vol. 52, No. 2. pp. 147-159.

Linthicum, D.S. (2001). B2B application integration: e-business-enable your enterprise. Addison-Wesley, Boston.

Luckham, D. (2002). The Power of Events. Addison-Wesley, Boston, USA.

Markeset, T., Kumar, U. (2003). Design and development of product support and maintenance concepts for industrial systems. Journal of Quality in Maintenance Engineering. Vol. 9, No. 4. pp. 376-392.

Marsh, L., Finch, E. (1998). Using portable data files in facilities management. Facilities, Vol. 16, No. 1-2. pp. 21-28.

Newman B. and Conrad K.W. (1999). A Framework for Characterizing Knowledge Management Methods, Practices, and Technologies; First Publication, January 1999, in support of The Introduction to Knowledge Management, George Washington University Course EMGT 298.T1, Spring 1999.]

Nonaka, Ikujiro, Takeuchi, Hirotaka. (1995). The knowledge creating company. How Japanese companies create dynamics of innovation. Oxford University Press, New York, USA

Parlikad, A.K., McFarlane, D., Fleisch, E., Gross, S. (2003). The Role of Product Identity in End-of-Life Decision Making. Auto-ID center white paper, Institute for Manufacturing, University of Cambridge, UK. Available online (June 21st, 2005): http://www.ifm.eng.cam.ac.uk/automation/publications/w_papers/cam-autoid-wh017.pdf

Polanyi, Michael (1962). Personal knowledge. Towards a postcritical philosophy. Univiversity of Chicago Press, Chicago, USA

Porter, M.E., Kramer, M.R. (2002). The competitive advantage of corporate philanthropy. In: Harvard Business Review, December 2002. pp. 56-68.

Probst, Gilbert, Raub, Steffen and Kai Romhardt (1997). Wissen managen. Wie Unternehmen ihre wertvollste Ressource optimal nutzen. Frankfurt am Main: Frankfurter Allgemeine, Zeitung für Deutschland; Wiesbaden: Gabler, 1997

Quillian, M.R. (1968). Semantic memory. In: M. Minsky (ed.) Semantic Information Processing, MIT Press: Cambridge, Massachusetts. pp. 216-270.

Simon, M., Bee, G., Moore, P., Pu, J.S., Xie, C. (2001). Modelling of the life cycle of products with data acquisition features. Computers in Industry, Vol. 45. pp. 111-122.

Stock, J., Speh, T., Shear, H. (2002). Many happy (product) returns. In: Harvard Business Review, July 2002. pp. 16-17.

SWAD-Europe / Semantic Web Advanced Development for Europe (2001) SWAD-Europe deliverable 12.1.8: SWAD-E Demonstrators - Lessons Learnt. Available online (May 31st, 2005): http://www.w3.org/2001/sw/Europe/reports/demo-lessons-report/

Thoben, K.-D. Weber, F. and Wunram M. (2000). Pragmatic approaches for knowledge management: Motivation from cases. In Proceedings of the 6th Internationl Conference on Concurrent Enterprising (ICE 2000), 28–30 June 2000, Tolouse, France, pages 165–169, 2000.

Timm, I.J., Woelk, P.-O., Knirsch, P., Tönshoff, H.K., Herzog, O. (2001). Flexible mass customisation: Managing its information logistics using adaptive co-operative multiagent systems. In: Pawar, K.S., Muffatto, M. (eds.) Logistics and the Digital Economy, Proceedings of the 6th International Symposium on Logistics, Salzburg, Austria. pp. 227-232.

Tsang, A. (1995). Condition-based maintenance: tools and decision making. Journal of Quality in Maintenance Engineering, Vol. 3, No. 1. pp. 3.

Tsang, A. (2002). Strategic dimensions of maintenance management. Journal of Quality in Maintenance Engineering, Vol. 8, No. 1. pp. 7-29.

Töyrylä, I. (1999). Realising the potential of traceability – A case study research on usage and impacts of product traceability. Finnish Academy of Technology, Espoo.

UCC (2002). GLN Implementation Guide. Uniform Code Council inc., available online (October 22nd, 2004): http://www.uc-council.org/ean_ucc_system/pdf/GLN.pdf

Weggeman, Mathieu (1998). Kennismanagement. Inrichting en besturing van kennisintensieve organsiaties. Schiedam Scriptum

Weerawarana, S., Curbera, F., Leymann, F., Storey, T. Ferguson, D.F. (2005). Web Services Platform Architecture. Prentice-Hall, Upper Saddle River, US. 413 p.

Wong, C.Y., McFarlane, D., Zaharudin, A.A., Agrawal, V. (2002). Intelligent Product Driven Supply Chain. In: Proceedings of IEEE SMC 2002, Tunisia.

Yoo S.B.; Kim Y. (2002). Web-based knowledge management for sharing product data in virtual enterprises. International Journal of Production Economics, Volume 75, Number 1, 10 January 2002, pp. 173-183(11)