

New Ideas for Knowledge Management in Product Development Projects

Christian Weber, Martin Pohl, Michael Steinbach

Keywords: knowledge capturing, semantic networks, weblogging, worksheet technique

1. Introduction

Due to their high complexity, product development processes should be expected to be a highly rewarding field for the application of knowledge management systems. Nonetheless, such systems are rarely used in practice. One important reason is that the extra effort of writing down knowledge is widely perceived as slowing down the actual design work without offering visible benefits. Also, writing documents is typically not a favourite pastime of technicians and engineers. There is a high inhibition threshold due to the expectation that the text within a document is well structured and formulated. Furthermore, documents as knowledge carriers require additional effort to retrieve the knowledge: Often the user has to screen several complete documents to find the piece of information he/she is looking for. Full text searches can help, but quite frequently lead to irrelevant information or miss out an important passages because different words were used.

So, economically speaking, knowledge management does not yield sufficient return on investment. The authors claim that this can be changed when

- knowledge capturing is not a separate activity, but is integrated in the daily design work,
- knowledge is integrated into a comprehensive representation of the design and the ongoing design process,
- knowledge capturing can be done with very low effort,
- knowledge retrieval is very efficient, i.e. the user finds relevant information quickly and reliably,
- the captured knowledge can be automatically processed in order to extract different views and to create reports for different purposes.

It will be shown that a system that handles a large number of small notes is able to fulfil these requirements. Writing a small note on an adhesive paper is surely the easiest and most popular way of putting down knowledge.

The rising popularity of Blogs and Wiki systems for documentation purposes, even in business, shows that people are willing to put down and to share knowledge if they can do it in small portions.

1.1 Weblog (Blog)

The word “weblog” is composed of “web” (for World Wide Web) and “log” (for logbook). It can also be called a “Blog” as well.

The following definition is provided by Radio Userland [www1]: “A weblog is just a web site organized by time”, which describes the main characteristic very well.

A Blog is a Website that contains short, frequently updated contents that are chronologically published.

An outstanding feature of this technology is the openness of the system, which permits each user to change the contents of the Weblogs without knowledge of special software (Frontpage or Netscape Composer) or knowledge of HTML. Contents of a Weblog can be entered over a simple user surface. As the user can also define web page links quickly and easily, a vast field of knowledge can be assembled with minimal effort.

This approach could be a promising IT-adaptation of the worksheet technique suggested by Hansen and Andreasen [Hans-99].

Only the setup of a Weblog requires a license and a knowledge of special software, entailing a small expenditure. The main reasons for the fast popularity of Weblogs lie in the following advantages [www2, www3]:

- simple handling and low time exposure when publishing contents
- separation of contents from layout
- automatic archiving
- can reach high detail depth of contents
- high knowledge-networks-effects by selective linking

It is problematic, however, that the data can be viewed in one perspective only, i.e. in a chronologically way.

1.2 Wiki

Wiki is another internet technology for publishing knowledge on internet or intranet sites, drawing its name from Hawaiian, meaning “fast-fast”. An impressive example of collaborative knowledge publishing with Wiki is www.wikipedia.org.

Similar to the Weblog, Wiki is based on the fact that each participant (dependent on the user rights) may change and delete everything. Caution should be taken as the openness of the system may result in the loss of data, intentionally or otherwise.

Wikis don't use HTML to create the website, but a particular not standardized, simplified syntax, that can differ from one Wiki to the next. This depends on the programming language used to create the Wiki (Perl, C, PHP, etc.) [www4]. Wikis which are written with the software TwikiClone (based on Perl) are called “Twikis”.

Contrary to weblogs, in which entries are stored and appear chronologically, Wikis can change anywhere on the site.

1.3 Interweaving knowledge management and project management

Although both these systems are promising alternatives to document-based knowledge management, the return on investment does not yet seem to be high enough to achieve a breakthrough. One reason is certainly that knowledge retrieval is not yet optimized. Also, they do not allow the processing of content by algorithms. This becomes important when knowledge management is extended to cover not only the rather static knowledge about the product, but also the highly dynamic knowledge about the ongoing design process.

Having a well-structured documentation of the “object” of the design process is helpful, but the really urgent questions are usually related to the design process itself. There are deadlines, quality gates, etc. to be surveyed, and unfortunately, conventional project management tools don't really have any understanding of the project content.

A very popular way of managing projects is the additional maintenance of spreadsheet tables containing tasks and their interdependencies. This allows to obtain different views by filtering, but also has some shortcomings:

- The screen always seems to be too small.
- It is almost impossible to include reasonably sized comments or descriptions in the spreadsheet cells.
- Consistency of the spreadsheet tables can not be assured and is usually lost very soon due to numerous changes.

So if there is a tool that is as flexible as a spreadsheet table, and at the same time ensures consistency like a database, and offers appropriate mechanisms for mass data processing, and additionally allows to stick several notes to each cell, you can integrate knowledge management and project management. The synergy effect is that the project model offers additional structure for the notes, while the knowledge management part provides the actual project content. In other words, while the user of a conventional project management system can only see that a task is not yet done, a more comprehensive model would allow him/her to directly read the notes describing where the problem is or to evaluate complex interdependencies.

For this purpose the semantic net, a structure of terms and their semantic relationships, offers an ideal solution, because it is almost as flexible and understandable as natural language, yet formal enough for automatic processing. Semantic nets are state-of-the-art in knowledge representation, but their creation and maintenance is rather expensive with conventional tools.

2. The Semaril

The Semaril - it is a imaginary name - has evolved out of several semantic net prototypes created at the Institute of Engineering Design/CAD (LKT) by continuous improvement of the usability. It is optimized for extremely quick handling and therefore the User Interface is quite different from conventional applications. Almost all operations are done on the same screen and usually by drag & drop. Therefore, there are very few control elements on the screen and almost no menus.

The Semaril may be seen as

- a very powerful memo pad,
- an HTML editor,
- a document management tool,
- a tool for project documentation and reporting,
- a knowledge management tool,
- a tool to harmonize different spreadsheet tables,
- a semantic net editor and navigation tool,
- a universal database.

The data model consists of terms, relations between terms, notes, groups of terms (“bags“) and filters.

A term can represent anything imaginable. A term in the Semaril corresponds to an entity in a conventional database. Technically, it consists of a name and (optionally) a memo text. This memo

text can contain hyperlinks to terms, files on a hard disk or websites. Terms can also be linked to each other by means of relations.

Notes are very much like terms, the difference being that a term has one "is a" relation telling something about its meaning. So every term is part of an abstraction hierarchy.

A term can be linked to other terms by various relation types ("is prerequisite for", "conflicts with"...), while a note can only be attached to one or more terms.

Bags and filters are used for mass operations on terms. Filters can be defined by a number of relations a term must have to pass the filter. These relations can also refer to abstract terms. For example, a filter relation "manages some project" would apply to all persons who manage a specific project. Filters can be used to search the whole semantic net and put the results into a bag or to filter the contents of a bag. Bags can be combined by the simple set operations: unification, subtraction and intersection.

Combinations of these few basic mechanisms allow very complex queries. Such queries can be recorded with a macro recorder and re-evaluated when the database contents change. Bags also enable the user to see and manipulate which relations all elements in the bag have in common, which, in combination with the queries mentioned above, offers a very comfortable and quick way of mass data maintenance.

The contents of bags can be used to create view-specific reports in form of HTML files containing the notes and relations or in form of spreadsheet tables. In the latter case, very complex queries on the semantic net can be defined by simply filling in the title row of a table, for instance: "task | handled by person | status | is prerequisite for task".

Information can be maintained in the spreadsheet table and re-imported to the Semaril.

The set-up of the software tool the Semaril is realised without the use of hard-coded structures, so the functionalities described later can be applied not only in a special branch but also in any enterprise. All necessary relations and their correspondent complements can be defined in the Semaril without any restrictions.

The software was created in such a way that almost all daily work can be done by using one single window, i.e. the main window as it is shown in figure 1. All required functions needed, for example, to edit (create, change, delete) elements and all their relations are included in the main window, which can be divided into five main sections (relations, main elements, clipboard, structure tree and details) (see fig. 1).

Each of the sections represents an independent work area that processes all the data, which is put into it by drag & drop applying its specific methods.

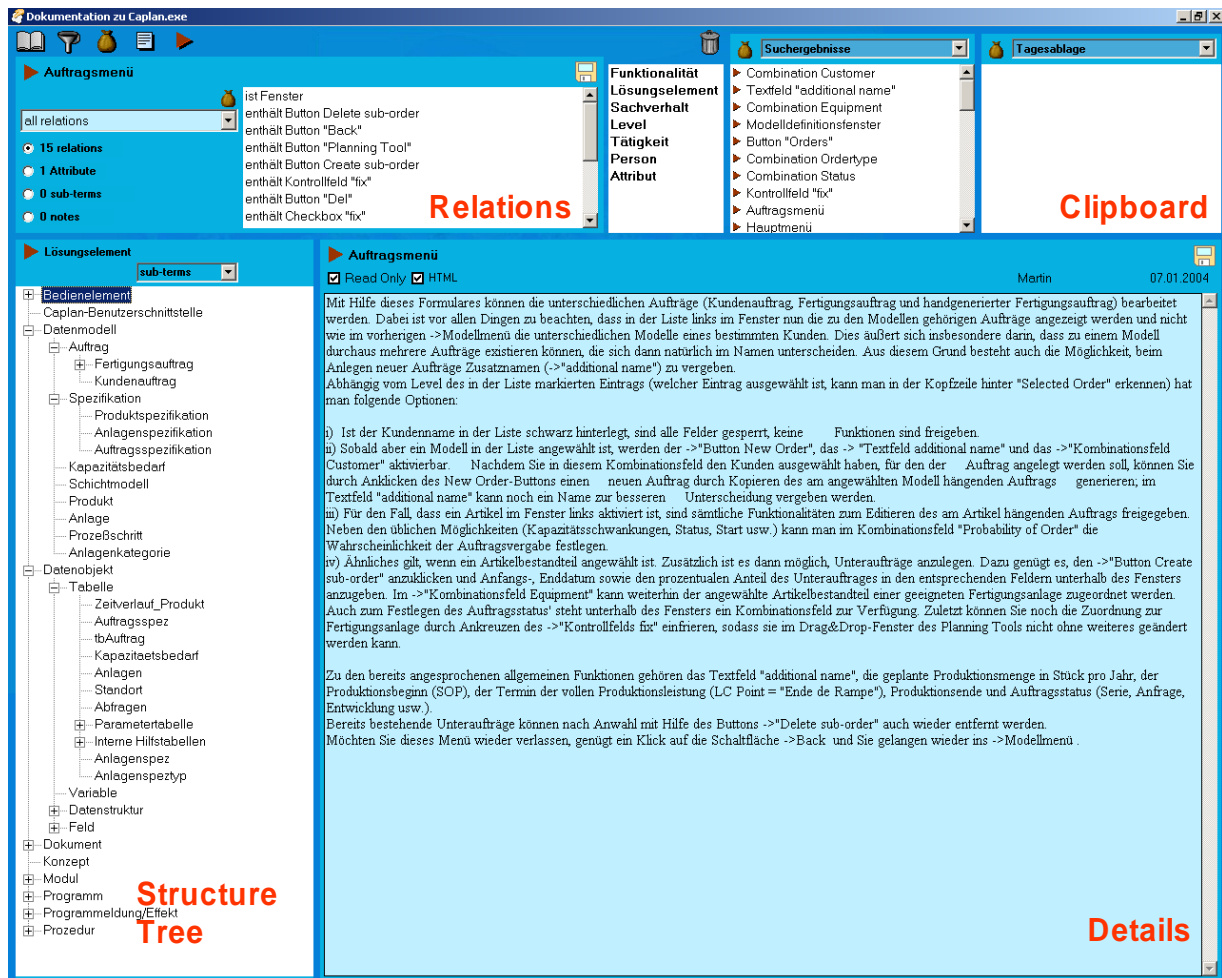


Figure 1. Main window of the Semaril

3. Using the Semaril in product development projects

In an engineering scenario, the Semaril may contain entities like:

- Engineering entities:
 - Requirement, Solution Scenario, Module, Concept, Part, Conflict of Objective, Product structure,
- Project Management entities:
 - Objective, Development Task, Problem, Role, Decision, Milestone, Meeting, Test, Quality Gate, ...

Subsequently, a possible industrial application of the Semaril is displayed, supporting the design of a hammer drill cooperatively done by a team of about 30 persons.

In the case of the person responsible for the design of the safety clutch of the drilling machine, the daily work probably includes activities like discussing with other designers, production planning and manufacturing, sketching concepts, creating CAD models, filling in spreadsheet tables, and writing specifications in MS Word. So the designer spends about 50-65% of his/her time at the PC. He/she finds it difficult to keep track of all the loose ends of an ongoing project and also complains that sometimes important information does not arrive early enough, forcing the revision of two or three days of work.

Before the development team can start to use the Semaril at the beginning of this new project, the software supervisor, an especially trained employee, has to adapt the present template including all above mentioned entities to the usually applied terms and the specific product range of the enterprise. This task may take about one or two work days, if the basic structure of the template is not significantly changed.

After this initial preparation of the software, all requirements are put into the knowledge base using a common spreadsheet table, including a column “affected modules”. When the table is imported to the Semaril, any new requirements and modules are created in the database, and they are linked by the appropriate relations. So the designer responsible for the module “safety clutch” can immediately see the requirements concerning his/her work.

The developing designer can now define one or more tasks associated with each requirement, either within the Semaril, or by exporting the requirements to a spreadsheet table and filling in the appropriate columns (figure 2). When importing the data from the table back into the Semaril, the new tasks are created (after a confirmation) and linked to the corresponding requirements and other topics in the same row.

Module	(affected by Requirement)	(solved by task)	handled by person	depends on task))
safety clutch	output torque 20 Nm	assure strength of transmission	Frank Meyer	define connection clutch - transmission
			Kurt Klauber	

Figure 2: a typical spreadsheet table for the representation/maintenance of knowledge

Another designer, who is in charge of the development of other transmission components of the drilling machine, can now read the entry “define connection clutch - transmission”. Without having to create a new entry the second person can add himself/herself to the list of designers responsible for or connected with this sub-task and place a note, that the bearings will be closer together than is usually the case, and that vibrations at the connection may occur.

Alerted by this entry, the designer responsible for the clutch creates a new task “simulate dynamic behaviour”, where the further proceedings are defined (e.g. a Finite Element Analysis).

In any case, this task has to be accomplished before the next task “dimensioning of the connection” can be carried out. In the Semaril this sequential dependency of tasks is displayed by the relation “has successor” and can easily be created by drag&drop.

After a while, the project manager asks all designers to fill in a list of questions which need to be answered in order to pass the first quality gate. This list is, of course, just another Semaril spreadsheet view, more or less with headlines like “question | task | responsible | estimated time for task”. By means of the sequence information, the project manager can then do a backwards termination to find potential conflicts (e.g. the person to create the CAD model of the clutch has other urgent tasks at the same time), and to re-assign some tasks to other persons if necessary.

Since the communication within the team can be mostly done by means of the Semaril notes, the whole development process is very well-documented, and whenever a decision has to be re-evaluated, all necessary information is quickly at hand. Also, re-using components imposes a lower risk, as the boundary conditions of their development are trackable.

4. Summary

Because of its flexibility in relation to the uncommitted definition of entities and relations as well as to the possibilities of establishing links to other data, the Semaril is a powerful tool for the modelling and the management of knowledge, which could especially be used to display design and design process knowledge.

The abilities of the Semaril to import and process data from (commonly used) established tools like, for example, CAD-data, which is still meant to be managed by specialised CAD- and/or PDM-systems, have to be examined and evaluated in the near future.

The creation of templates for different industrial branches and thus requirements is projected to make a structural proposal for the user, allowing quick and effective work with new software tool.

Also in research it seems to be very useful to apply the Semaril, as is shown by testing it in this application inside the institute. Ideas and considerations can be captured and structured, references can be stored and connected with certain topics, which enables the user to find them easily and completely when necessary.

References

- [Hans-99] Hansen, C.T.: Identification of Design Work Patterns by Retrospective Analysis of Worksheets, Proceedings of ICED 99, WDK 26, vol. 2, pp. 775 ff., München 1999
- [Sund-01] Sundqvist, K.: Implementierung einer Werkstoffdatenbank mit Suchmaschine auf Basis eines semantischen Netzes, Diplomarbeit am Lehrstuhl für Konstruktionstechnik/CAD, Saarbrücken 2001
- [Wank-00] Wanke, S.: Implementierung einer Dokumentensuche basierend auf einem Semantischen Netz, Studienarbeit am Lehrstuhl für Konstruktionstechnik/CAD, Saarbrücken 2000
- [WeAh-99] Werner, H.; Ahmed, S.: Design Capturing with a Model System Using Event Triggered Procedures, Proceedings of ICED 99, WDK 26, vol.2, p. 1011-1014
- [Wern-01] Werner, H.: Integration von CAx-Funktionalitäten in einem neuartigen Konstruktionssystem, Dissertation, Universität des Saarlandes, Saarbrücken, 2001
- [WeWD-02] Weber, C.; Werner, H.; Deubel, T.: A Different View on PDM and its Future Potentials, Proceedings of DESIGN 2002, vol. 1, p. 101-112
- [www1] <http://radio.weblogs.com/0107019/stories/2002/02/12/whatIsAWeblog.html> (08.01.2004)
- [www2] Schmidkonz, C.: Blogging auf der Schwelle zur Business-Anwendung, 05/2003, http://www.contentmanager.de/magazin/artikel_333_blogging.html (08.01.2004)
- [www3] <http://www.abseits.de/weblogs.html> (08.01.2004)
- [www4] <http://twiki.org> (30.07.2003)
- [www5] <http://www.ibs-academy.de/pagesum5/semanetz.html> (30.06.2003)

Prof. Dr.-Ing. Christian Weber,
Dipl.-Ing. Martin Pohl
Dipl.-Ing. Michael Steinbach
Saarland University, Engineering Design/CAD
PO Box 15 11 50
D – 66041 Saarbrücken, Germany
Phone: +49/ (0)681 / 302 – 3075; Fax: +49/ (0)681 / 302 – 6111
Email: pohl@cad.uni-saarland.de
URL: <http://www.cad.uni-saarland.de>