INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED07
28 - 31 AUGUST 2007, CITÉ DES SCIENCES ET DE L'INDUSTRIE, PARIS, FRANCE

TOWARDS INTEGRATION OF KBE AND PLM

Amer Ćatić[1], Johan Malmqvist[1]
[1]Chalmers University of Technology, Göteborg, Sweden

ABSTRACT

In this paper the issue of integrating knowledge based engineering (KBE) and product lifecycle management (PLM) is addressed at an architectural level. State of the practice and state of the art KBE applications in the literature and in industrial use constituted the empirical base for a categorization of such applications. Two categorizations are presented; one where applications are viewed from the perspective of desired result and one which relates the KBE application to the task it performs and the tool it performs it with. A service oriented PLM architecture has been found to be promising for integrating KBE and PLM. PLM services, whose main role is to retrieve and store data needed or generated by the KBE application, constitute the integration pattern. The KBE applications, which apply PLM services, are in their turn offered as services provided through the PLM environment. Based on these KBE services, a concept called knowledge modules is introduced. The aim of knowledge modules is to map and integrate KBE applications to support or automate engineering activities which today are performed as services between engineering departments such as e.g. complete design verifications or complete configurations.

Keywords: knowledge based engineering, KBE, product lifecycle management, PLM

1 INTRODUCTION

When computers were introduced as tools in engineering work the idea of using them to capture and reuse engineering knowledge arose. The first systems of this kind appeared in the mid 1970's, so-called expert systems used for rationalizing different activities such as problem solving, calculations, simulation, configuration, design and so on [1]. Later on the use of these kinds of systems in engineering was labelled Knowledge Based Engineering (KBE) referring to the fact that some sort of explicit engineering knowledge is embedded in them. Since then, many examples illustrating the potential of this technology have been demonstrated in the literature [1-11].
Applications with embedded knowledge in use in the industry today are mostly developed internally. The applications vary in size and number. Some advanced solutions are integrated in computer aided design/engineering (CAD/CAE) applications [2,4,11] but most of the applications are small and performing partial tasks such as a spreadsheet performing standard calculations as part of a more extensive engineering task. The applications are also for the most part very loosely, if at all, integrated with product data management (PDM) and computer aided design/manufacturing/engineering/other (CAD/CAM/CAE/CAx) systems [12]. A typical situation is depicted in Figure 1 where the KBE applications are isolated islands and with humans as the only interface towards other systems.
The situation depicted in Figure 1 clearly illustrates the issue of a need for better management of the knowledge in product development. This is realized by an implementation of a strategy for application

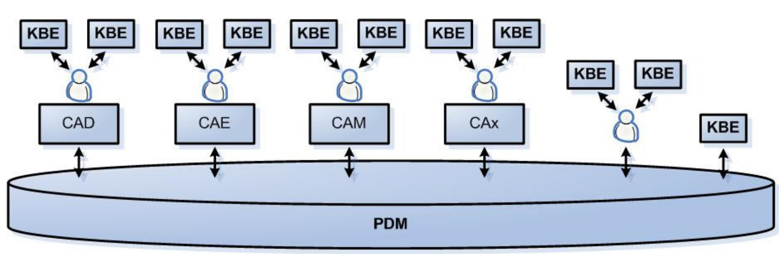

Figure 1 – Current situation of disintegrated KBE applications

of KBE in product development which, in this case, translates to applying a structured approach of integrating KBE with product lifecycle management (PLM).

The development of CAD systems has brought about a tool for more efficient capture and reuse of product data. The introduction of PDM brought about a better way of managing product data. It made it possible to manage the complexity of ensuring a consistent product description at different abstraction levels through versioning, storing, managing access to and structuring the product data created by CAD. Now there is a need to manage the knowledge, from which the product data has resulted, in the same way. However, there is an essential difference between integration of CAD and PDM and integration of KBE and PDM. In the case of CAD integration the management, in terms of e.g. change management or access levels, is facilitated by the fact that the data consists of files and metadata as a package which is easy to manage. In the case of KBE the data to be managed is more fine-grained, e.g. design rules, which makes it harder to manage since the data management needs to be done at such a low level, e.g. versioning individual design rules. Before this issue is addressed the issue of finding a structured approach to transforming the state of KBE applications from being isolated islands, as depicted in Figure 1, towards being an integral part of the PLM environment needs to be addressed. This strategy for the integration of KBE with the PLM environment should, among other things, contain a definition and a categorization of KBE along with an architecture for integration of existing and new KBE applications with the product lifecycle systems. The aim of this paper is to propose such a strategy along with a system architecture that will support its realization.

The paper is structured as follows. The first two sections describe the research approach undertaken along with a summary of state of the practice and state of the art of KBE applications. After that two ways of categorizing KBE applications are presented and finally a proposed architecture for structuring and bringing KBE towards integration with the PLM environment is described with the aim of transforming KBE from being isolated islands to a part of the PLM environment.

2 RESEARCH APPROACH

This study was started with performance of a literature study in the fields of expert systems[13,14], design automation[3-7,9], configuration[2,9], computer aided engineering[4,5,8,10,12] and product lifecycle management[15-18]. This was done in order to find applications implementing KBE; their functionality, implementation and development. A number of applications were found and this constituted the empirical base for a categorization of KBE.

The categorization was finalized and verified by interviewing engineers who have developed and are developing applications which implement KBE to support product development either in tasks relating to CAD or CAE. Along with this the proposed architecture for integration of KBE and PLM using the newly proposed concept of ‘knowledge modules’, presented in Section 5.4, was verified in the interviews. The interviewed engineers are active within the Volvo Group.

3 RELATED WORK

The term Knowledge Based Engineering has been used in different contexts ever since its introduction. There is still not a universally accepted definition of exactly what is embraced by the term. There are some different definitions. MOKA [1] defines KBE as: *“The use of advanced software techniques to capture and re-use product and process knowledge in an integrated way”*. CommonKADS [19] does not have a definition of KBE but they define the term “knowledge systems” as a gathering term for expert systems, knowledge intensive information systems and knowledge based systems. Another term called “knowledge engineering” is defined as evolved from the art of building those systems.. Pinfold and Chapman [10] define KBE as a derivative of CAD where, besides geometry, also the design intent is captured and therefore KBE is defined as a framework for capturing and defining the process of design creation. Penoyer et. al. [12] define KBE as computer systems used for engineering, focused on a representation and application of knowledge to specific problem cases, has deep penetration into the problem domain and reasons through the problem solving process using rules of logic rather than mathematical models. Poenisch and Clark (2006) [20] confirm that no definition of KBE has found a general acceptance yet but they believe that an essential ingredient of KBE is a software application that processes some kind of engineering knowledge. In this paper KBE is, in a broad sense, considered to include all kinds of applications whose intent is to capture and reuse engineering knowledge, with the term application not being delimited to any particular type of computer software, e.g. ICAD [21] or CAD integrated KBE module such as Catia V5 Knowledgeware [22], UGS NX Knowledge Fusion [23] and Pro/Engineer's Pro/Program [24].

3.1 KBE Tools

The field of design automation is a typical example of KBE. Applications present in the design automation literature describe computer systems which apply engineering knowledge to automate design tasks in order to save time, relieve engineers from tedious tasks and assure that every instance of the designed component is designed using the same rules, thus ensuring a certain level of quality and standardization.

What is common for all examples of design automation applications found [2-7,10] is that they are implemented on mature components for which all of both product and process knowledge is known. This has also been identified by [7] as one of the basic prerequisites for a successful design automation system. The main focus of these applications is the time saving aspect [19] but they also consider other aspects such as different DFX techniques for increased producibility, higher quality or provide information for better decision making such as e.g. cost estimation for different variants [6]. Many of the mentioned applications are mostly concerned with activities related to the synthesis steps of product development. There are applications which demonstrate examples of automation of both synthesis and analysis activities [4,5,8,10].

Most of the applications above will produce predictable results due to the fact that they deal with a low level of uncertainty (either there is one correct outcome or the maturity of the component is high with a high number of known defining rules). Systems which deal with more uncertainty and where the outcome is not as easy to predict are those applying techniques such as case-based reasoning (CBR)[13,14], neural networks (NN) [13,14] or optimization loops [9]. The purpose of these systems is to navigate through a larger space of solutions. The biggest advantage of such systems is that they have the ability to produce results which human engineers might not have found themselves, at least not in the same amount of time [4,9].

3.2 KBE Methods

The proposed methods for developing KBE applications found in the studied literature are the Methodology for Knowledge based engineering Applications called MOKA [1], the methodology for support of knowledge engineering called CommonKADS (Knowledge Acquisition and Documentation Structuring) [19] and Cederfeldt's methodology [7] for planning design automation systems.

MOKA is a result of research done within the European ESPRIT IT Programme by a consortium whose members consist of several European universities and companies from the automotive and aerospace industries. MOKA proposes a structured way of acquiring, structuring and representing knowledge in order for the knowledge to be used in a KBE application.

CommonKADS was developed at the University of Amsterdam also within the European ESPRIT IT Programme as a respond to the need for a standard for knowledge based systems. CommonKADS has a direction towards knowledge in general implying that they could handle not only engineering knowledge but also other types of knowledge such as diagnosis, scheduling of activities and so on. Both of these methodologies propose a structured way of acquiring, structuring and representing knowledge; engineering or other. Both of them also have a clear focus and intention on implementation of knowledge in some kind of stand alone software.

Cederfeldt's methodology [7] gives a basis for primarily the planning of design automation systems through decision criteria regarding identification of possible components for design automation. The methodology also provides support in whether the knowledge should be stored inside or outside the CAD model, the mapping of design rules, how the CAD model should be defined and so on.

3.3 Product Lifecycle Management

The area of product lifecycle management (PLM) is a vast area embracing many disciplines. CIMdata [15] provides the following definition of PLM:

Product Lifecycle Management (PLM) is a strategic business approach that applies a consistent set of business solutions in support of the collaborative creation, management, dissemination, and use of product definition information across the extended enterprise from concept to end of life – integrating people, processes, and information.

An even broader definition of PLM is provided by John Stark [16]:

PLM is the business activity of managing a company's products all the way across their lifecycles, from the very first idea for a product all the way through until it is retired and disposed of, in the most effective way.

In this paper a narrow definition of PLM relating to tools used and processes involved will be considered for KBE integration. The first delimitation will be a focus on so called product lifecycle systems [12] which usually are referred to as PDM and CAx. PDM is viewed as a concept consisting of one or a system of several applications which will, besides product defining data such as geometry, assembly relations, functional relations and requirements also entail analysis and test results, manufacturing data, configurations and so on. CAx stands for the computer based tools the engineers use to author product and process data, the x is thus replaced by design, manufacturing, engineering, process planning, requirements management and so on where each of CAx-s consist of either one or a system of applications. A schematic view of the considered situation is depicted in Figure 2.

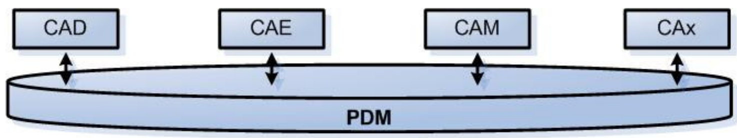

Figure 2 – Product lifecycle systems

Definitions of PLM, such as the one provided by CIMdata, cover all the stages and processes that take place during a products lifecycle, from idea to disposal. In the same manner as for the tools used this paper will consider the earlier stages of a product lifecycle as depicted in Figure 3. These stages provide an overview of how the product is viewed during its lifecycle.

Figure 3 – Product lifecycle stages [25]

3.4 Conclusion

As can be seen there are many examples which demonstrate the capabilities of KBE applications to improve the outcome of engineering work and simultaneously rationalize it. The described methods provide a basis for creating new KBE applications from as well the technical software as knowledge capture and representation point of view. The issue of integrating KBE with PLM has been addressed at a general level [12] but there is so far no proposed way of going from the current situation towards a more structured way of integration. As was mentioned in Section 1 there are two issues relating to this integration. The first one is the fine-grained nature of data in KBE applications which makes it more difficult to manage. The second is the lack of a structured approach to and view of KBE applications. The addressing of the first issue, which is at a more detailed level, needs an addressing of the second issue which is at a more general level. The structured approach needs to provide support and a holistic view of KBE from a product development perspective. This also implies that the implementation of KBE will differ at different sites depending on the product development process it supports and the proposed approach needs to provide the flexibility needed to meet this requirement. The need for flexibility reflects on the proposed system architecture, which addresses the first issue.

4 CATEGORIZATIONS OF KBE

In this section two ways of categorizing KBE are presented. The first is based on whether the focus of the KBE application primarily is automation or increased quality of the solution. The second categorization relates the engineering task, which the KBE application rationalizes, to the tool in which the KBE application is realized. The common requirement for both categorizations is that the applications they categorize capture and reuse engineering knowledge.

4.1 Result oriented categorization of KBE

As the description implies this categorization of KBE applications is based on what is the desired result from the application. Some KBE applications focus on improving quality or performance of the solution while other applications primarily focus on saving time thus reducing cost. Both do it by applying knowledge. This discussion is delimited to applications which only handle explicit knowledge. The categorization is depicted in Figure 4.

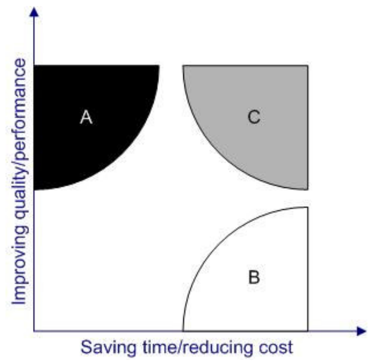

Figure 4 – Categorization of KBE applications according to the expected result

A – Examples of applications in this field are books containing guidelines, reports containing lessons learned from past projects, software applications which have a guiding role e.g. for material selection or CAD integrated warnings for e.g. sharp edges and databases containing documentation of e.g. QFD, FMEA. Examples found in this field are [7,22,26-28]

B – The aim here is to automate repetitive engineering tasks. Examples are applications for standard calculations such as bearing or screw joint dimensioning, quality statistics or parameterized components for which the geometry defining rules are well known [2-7,10].

C – Applications in this field are those who deploy some kind of optimization loops e.g. [4,9].

The fourth corner of the graph is not discussed since there is no point in having KBE applications which neither improve the solution nor save time/cost.

4.2 Task oriented categorization of KBE

The activities performed during development of a product can generally be described by the process depicted in Figure 5. In this categorization focus will be on the first three activities.

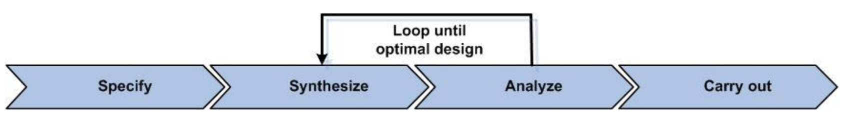

Figure 5 – Product development activities [29]

During these activities engineers author product data. The two kinds of knowledge involved in this work are knowledge about the product and knowledge about the processes used in the development of the product [1]. For the performance of these activities engineers use different tools. Originally the tools used were drawing boards together with equipment needed to test, measure, calculate, produce prototypes and so on. It can be argued that the first approach to capture and reuse engineering knowledge were media containing design guidelines, e.g. sketches from the 15th century depicting exemplifying solutions for different problems. Later on, in the trails of the rising industrialism, handbooks containing design and calculation guidelines along with standards became common applications for engineering knowledge management. The use of books as KBE applications reflects the engineering tools used at the time. All engineering activities were performed manually and thus the KBE applications were also manual in the form of handbooks. Today engineers use a wider range of tools as they create product data; computer aided (CAx) systems with different kinds of dedicated software. The existing KBE applications in industrial use are realized in the same software tools.

These ideas lead to the viewing of KBE and KBE applications, in general, to be applications whose main purpose is to capture and reuse engineering knowledge, as was stated in Section 3. This implies that the nature of the KBE application has to be put in relation to the engineering tool by which it is realized. The requirement on KBE to entail a particular kind of computer software, as many of the found definitions require, results in KBE only embracing applications related to engineering software tools. This neglects the fact that "an engineer's notebook" still is a tool widely used by many engineers [30]. If the requirement is posed in terms of "advanced software techniques" [1], KBE applications realized in e.g. spreadsheets performing simple calculations and saving time would be scoped out.
If taken to the other extreme this discussion also implies that engineers themselves can be viewed as KBE applications implemented in the engineering tool of the human brain. From this perspective a mapping showing the distribution of the total knowledge required in engineering work can be done. From this the following three categories of tasks can be deducted:

- **Creative tasks** which require humans to use their experience, creativity and imagination to produce product data [31]. The "KBE applications" needed for the execution of these tasks are humans executing implicit (non-expressible) knowledge. Both product and process knowledge are possessed by the engineer.
- **Semi-standardized** tasks which require humans to use standard formulas or rules to produce product data. The KBE application for this type of task usually relies on a human for execution and a media, such as a book or a database, for storing the explicit knowledge. This can be viewed as a semi-automatic KBE application where one of either process or product knowledge are contained in the KBE application and the other is possessed by the engineer.
- **Standardized tasks** which can be executed without human intervention. These KBE applications rely on computers for execution and a digital knowledge base whose form is such that it is suitable for computer execution. This can be viewed as an automatic KBE application which contains both product and process knowledge.

There are two reasons for why the semi-standardized tasks might require a semi-automatic KBE application and not an automatic one. The first reason is that the knowledge about the task execution (process knowledge) might be of implicit form and thus can not be expressed in a suitable way for computer execution. The second reason is that the product knowledge might have uncertainties which require further clarification upon execution which can only be done by an engineer. Cederfeldt [8] has referred to this as a relation between how many design rules that are known and how many design rules there are. Here "design rules" refer to as well product as process knowledge. The closer this relation is to unity the higher is the potential of a design automation application for that particular design task. Adding the dimension of type of task to product development activities gives the categorization in Table 1. For every category there are applications listed. These are only examples which illustrate the different KBE applications used in the different cases.

Table 1 – Categorization of KBE applications with examples

	Specification	**Synthesis**	**Analysis**	
Creative tasks	Designer/Marketer	Designer + TRIZ [27]	Analyst	**Human**
Semi-standard tasks	Requirements database	Design guidelines [22][26][28]	Calculation guidelines	**Semi-automatic KBE application**
Standard tasks	Feature packages Options	Parameterized solid + Rule Base [3][5][7]	Automatic simulation program[5][11]	**Automatic KBE application**

5 KBE AS SERVICES IN PLM

In this section an architecture for integrating KBE and PLM is proposed. The proposed solution is delimited to automatic and semi-automatic KBE-applications according to the categorization presented in section 4. This delimitation is justified by the fact that the general strive in both academia and industry is to make the capture and reuse of knowledge as automatic and digital as possible, as can be observed among the application examples referred to in Section 3 about related work. In addition to this a concept called knowledge modules is introduced.

5.1 Service oriented PLM

The application of a service oriented architecture and mindset in PLM has come up as a trend in recent years which has been illustrated by recent initiatives [17,18,32,33] which strive towards creating a flexible PLM environment with best of breed PLM tools, both commercial and in-house developed, performing specialized tasks. The general idea is that every database where information is stored offers its information as services. These services are basically retrieval or storage of information with basic PDM functionality such as versioning, identification, effectivity and so on. The information is communicated using a communication standard and the information is modelled using an information model standard [33]. This is done for the sake of setting the ontology in the PLM environment. The idea is to use these basic services to enable higher level information integration and visualization from heterogeneous sources from different engineering domains and from different databases in these domains. These basic services are referred to as PLM services [17,33]. To make sure that all the databases containing different kinds of information can offer this information as services there is a need for some kind of translator, schema mapping or ontology mapping mechanism to ensure that the services provided by the different information sources comply with the overall standard in the PLM environment.

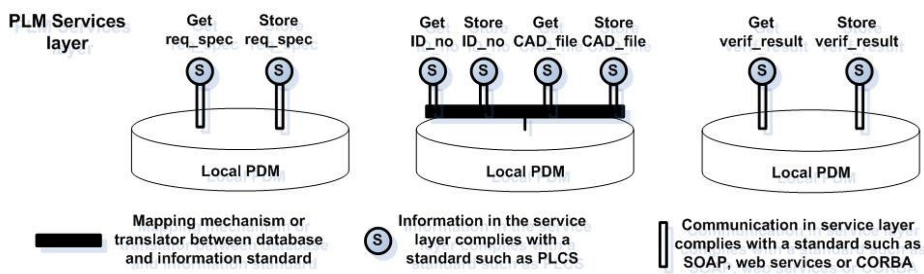

Figure 6 – PLM services

5.2 KBE and PLM services

Based on the above mentioned PLM services performing retrieval and storage of information there is a higher potential for integrating KBE with PLM by implementing PLM services to enable KBE applications to communicate with the PLM environment. This integration has several benefits. When integrated; the KBE applications can be easier created since more of the information needed, such as product models, configurations, analytical results and so on, can be retrieved from other sources and thus do not need to be created locally in the application for its sole purpose. There is a higher possibility of separating the storage of the knowledge from storage of the models executed by the application, e.g. the storage and versioning of a CAD model can be done in a CAD vault while the rule base, which controls the parameters in the CAD model, can be stored elsewhere without bothering the user to store them separately and in different databases. This is done automatically as the application uses different PLM services for the different actions. This example is depicted in Figure 7.

5.3 KBE as services

In the same manner as PLM services KBE applications can be modelled as services in the PLM environment. These services would be performed by automatic KBE applications performing such engineering tasks for which there is a high level of maturity in both product and process knowledge which makes them suitable for automatic KBE applications according to the categorization in section 4. Examples of such automatic KBE applications could be a synthesis of a solid model by execution of a parameterized CAD model along with a rule base or perhaps a standard calculation of e.g. bearing lifetime by execution of governing formulas along with loads and surrounding parameters. KBE services would be given certain inputs by the user and perform a processing of information according to some intelligence either through some standard predefined process, through application of computational techniques such as CBR or rule sets along with an inference engine. The KBE services rely on the PLM services to retrieve and store the correct information in a correct manner as described in the sub-section above.

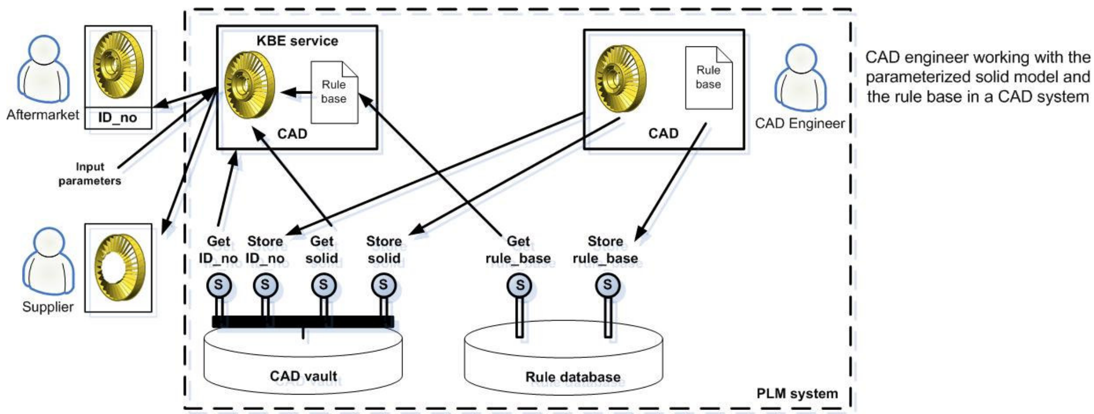

Figure 7 – KBE services

5.4 Knowledge modules

A knowledge module is an integrating entity for KBE applications and it contains both product and process knowledge needed for the performance of an engineering task. The idea of knowledge modules is to integrate KBE applications which perform partial tasks and package them as services across engineering, product lifecycle process or company domains. There is already today a service oriented way of working across engineering and product lifecycle process domains at the observed companies and it is these product information related engineering activities performed as services which should be subjects for knowledge modules. A typical example is an engineer in the design department who makes a request for some kind of analysis (e.g. component strength verification) of a new design. The analysis (verify component strength) is provided as a service by the calculation department. Other examples are simulations and manufacturing adaptations performed as services towards the designer or configurations and design information provision performed by the designer towards sales and aftermarket or a supplier. The idea with the knowledge modules is that these services are provided through the PLM system. The engineering task could be performed by one KBE application or several KBE applications which use each other as services, all depending on the task and the KBE applications themselves, an example is depicted in Figure 10.

The knowledge modules, or rather services which they perform, consist of several activities. The activities can be performed either by humans, by automatic KBE applications or by humans who use semi-automatic KBE applications as support. The knowledge module could be modelled using e.g. the IDEF0 technique [34] depicted in Figure 8.

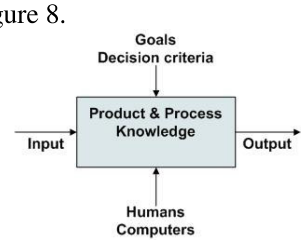

Figure 8 – An IDEF0 model [34] of a knowledge module

The idea of knowledge modules has several objectives. When modelling engineering processes using IDEF0 the performed activities are mapped and their interdependencies are revealed together with their inputs, outputs, mechanisms and controls. An analysis of which of these activities are performed by automatic or semi-automatic KBE applications is done in order to map out known and find unknown applications and finally see where new KBE applications could be, or need to be, developed. This provides a tool for a strategic approach towards applying and developing KBE applications. The proposed process for performing this mapping and analysis is depicted in Figure 9.

The use of IDEF0 to map out the engineering process is just used to exemplify. Any other tool for modelling processes could be used depending on the engineering process to be mapped. The second step in the flowchart, the analysis of the KBE environment could be done by e.g. interviewing the engineers or by performing a survey or a questionnaire. A method and a tool for this kind of analysis is currently being investigated by the authors. In the final step where an implementation of new KBE applications is done the methodology proposed by Cederfeldt [7] could be applied.

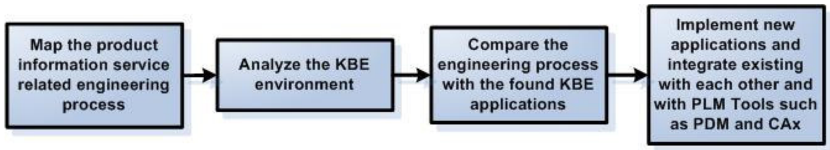

Figure 9 – A strategic approach towards integration of KBE and PLM

The analysis of the engineering processes together with the existing KBE environment also provides information about which KBE applications need to communicate with other PLM engineering tools such as PDM, CAx or requirement management tools. In this paper this translates into which PLM services the KBE applications need. An example of one such mapping is illustrated in Figure 10 where a current process of component strength verification along with the existing KBE environment is illustrated in the upper half. After a discussion of the process and analysis of the KBE environment some possible new KBE applications (offered as KBE services) and PLM services were identified depicted in the lower half of the figure.

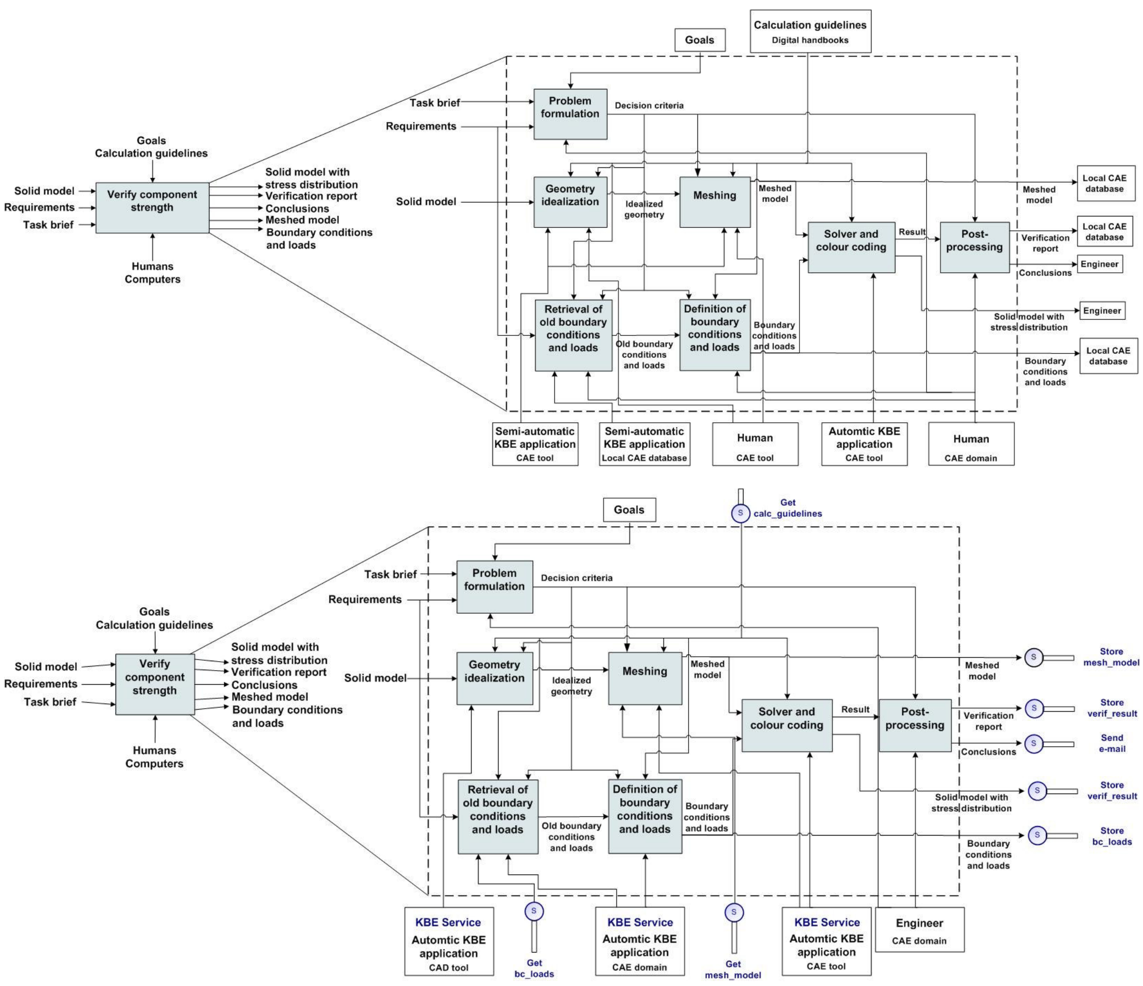

Figure 10 – Upper half: component strength verification process at one observed departments. Lower half: result of process from Figure 9 where some suggestions for new KBE applications were proposed along with application of PLM and KBE services

6 DISCUSSION

The proposed architecture addresses the problem of the lack of a structured approach towards KBE as a supporting tool for product development. There are however many issues related to this to solve. One of them, mentioned in the introduction, is the fine-grained nature of KBE knowledge structure which makes them hard to manage. In the proposed architecture the knowledge is managed in batches in the form of e.g. text or tables containing rule sets meaning that a solid model or a calculation model and the rule sets which belong to them are managed separately. Upon execution the executing application merges the correct model with the correct rule set by some ID number which means that every model needs to have its own rule set even if the rule sets are 90% identical. However this is dependable of the granularity of the PLM services. If for example the database containing design rules for a product platform in which many solid models contain global platform parameters stores the rules for each parameter separately then the PLM services provided by this database need to reflect this granularity. This refinement of PLM service granularity could be a means for managing the fine-grained nature of KBE applications. The need to model the knowledge in the system also becomes more apparent when the level of granularity is reduced opposed to managing rule sets in files. In other words the integration pattern has to be further investigated in order to set the PLM service granularity at a suitable level. This applies especially to PLM services needed for the support of KBE applications. The application of PLM services and KBE services as described in this paper might lead to a need for many translators and small applications for e.g. separation of the rule base and the solid model in CAD. However the use of widely accepted information and communication standards ought to minimize this need. Related to this are also interface issues concerning software packages which are unwilling to share information. This protectionist behaviour is recognized both by [12] and the interviewed engineers who see it as one of the main obstacles for software integration. This behaviour also leads to a conflict between where KBE should be implemented and where it should be used. In this paper it is argued that logically a KBE application should be implemented in the engineering tool used for the task that the KBE application is supposed to perform. The problem with this appears when the results of the KBE application are needed by another KBE application working in some other tool. If the software tools, in which the KBE applications are implemented, do not communicate then these KBE applications remain isolated islands with a need to manually transfer data, if even that is possible. In the worst case, as mentioned by one of the interviewed engineers, there is a need to build and manage two identical models of the same component in the different software tools.

The proposed use of information modelling standards in the service oriented architecture has by some of the interviewed engineers been appointed as an area containing issues related to the fact that some information is lost in the conversion. This is especially true for information related to special functionality in CAx systems. However in the service oriented PLM and KBE the important issue is not which functionality that was applied within the CAx systems to obtain the result but rather that the result can be modelled in a standard format. It is only when information in the result cannot be modelled due to special functionality that this issue appears which to some extent could be resolved by inclusion of metadata.

An important issue which has been discussed in other related literature about KBE and design automation is the need to show the end user of the KBE applications exactly what is going on to avoid the risk of a black-box effect leading to the user being suspicious towards the application due the simple fact that its way of providing the answer is not clear. The KBE services should clearly show which PLM services and other KBE services have been used to retrieve information and how the provided result has been realized.

In this paper, and in the reviewed literature, there is a general aim towards reasoning about the reuse of engineering knowledge from earlier product lifecycle phases in later phases. There is, however, not much discussion about how the knowledge from later stages, such as production and aftermarket, is to be fed back into the earlier stages. This was pointed out by the interviewed engineers as an important issue to consider and further investigate.

7 CONCLUSIONS

From the presented results and ideas in this paper it can be concluded that KBE is still an area without an established definition and lacking methods or approaches for integrating it with product development and PLM. This paper argues that a broad view of KBE is needed in order to entail all applications which capture and reuse knowledge when a structured approach towards integrating KBE with product development and PLM is considered.

It is also argued that KBE applications should be implemented in the software tools which are dedicated to the engineering task which the KBE application is supposed to support or rationalize. This leads to issues concerning that actors within other domains who might have use of the KBE application also need to have the dedicated software. It also leads to issues concerning the need to integrate each software containing KBE applications with the PLM environment separately. These issues are addressed by application of a service oriented architecture where KBE applications use PLM service to retrieve and store data and KBE applications themselves are offered as services towards other actors who don't need to have dedicated software to obtain a result from the KBE application. This service oriented mindset is already applied by the observed and interviewed engineers who perform much of their work as services towards actors in other domains.

It was finally concluded that the service oriented PLM architecture seems promising at an architectural level but there is further investigation needed in the issue of what level of granularity the PLM services should be implemented at to support KBE services.

8 FUTURE WORK

A demonstrator of PLM services, KBE services and knowledge modules will be developed in future work. Other issues that will be more closely investigated are how the management of KBE applications can be done with respect to their fine-grained nature and how the issue of software interfaces is addressed. The issue of applying KBE as a means for feedback of knowledge from later to early phases will also be investigated. Also the proposed process for a more strategic approach towards implementing new KBE applications in product development, see Figure 9, will be further studied.

ACKNOWLEDGEMENTS

This study was funded by the Swedish Governmental Agency for Innovation Systems "VINNOVA" through the Vehicle Research Programme. The authors would like to express their gratitude to Dr. Ola Isaksson at Volvo Aero along with the group of experts from various parts of the Volvo Group who have provided insightful comments about this work. A great deal of gratitude is also accredited to the interviewed engineers for their time and insights in the topic.

REFERENCES

[1] Stokes, M. (2001) *Managing Engineering Knowledge: MOKA: Methodology for Knowledge Based Engineering Applications*, Professional Engineering Publishing Limited, London, UK

[2] Fuxin, F. (2005) *Enabling Preconditions for Geometrical Application in Preconceptual Stages of Engineering Design*, Proceedings of ICED 05, page 79-80, Melbourne, Australia

[3] Colombo, G., Girotti, A., Rovida, E. (2005) *Automatic Design of A Press Brake for Sheet Metal Bending*, Proceedings of ICED 05, page 124-125, Melbourne, Australia

[4] Merkel, M., Schumacher, A. (2003) *An Automated Optimization Process for a CAE Driven Product Development*, Transactions of the ASME, vol. 125, page 694 – 700

[5] Boart, P., Andersson, P., Elfström B-O. (2006) *Knowledge enabled pre-processing for structural analysis*, Proceedings of NordPLM '06, page 89-97 Göteborg, Sweden,

[6] Elgh, F. (2006) *Automated Cost Estimation of Products Variants – A Tool for Enhanced Producibility*, Licentiate thesis, Chalmers University of Technology, Göteborg

[7] Cederfeldt, M. (2005) *Planning Design Automation Systems – Criteria and Guidelines*, Licentiate thesis, Chalmers University of Technology, Göteborg

[8] Bylund, N., Isaksson, O., Kalhori, V., Larsson, T. (2004) *Enhanced Engineering Design Practice using Knowledge Enabled Engineering with Simulation Methods*, Proceedings of DESIGN 2004, page 841-846, Dubrovnik, Croatia

[9] Cederfeldt, M. (2006) *Towards a Strategy for Mapping of Design Problems to Suitable Solutions – A Case of Design Automation Using CBR*, Proceedings of DESIGN 2006, page 471-478, Dubrovnik, Croatia

[10] Pinfold M., Chapman C. (1999) *The Application of Knowledge Based Engineering Techniques to the Finite Element Mesh Generation of an Automotive Body-in-white Structure*, Journal of Engineering Design, Vol. 10, No. 4, page 365-376
[11] Strinning, P. (1995) *A Computerized Trial and Error Technique for Impeller Design*, The ITT Flygt Scientific Magazine, no. 3, pp. 19-27
[12] Penoyer, J.A., Burnett, G., Fawcett D. J., Liou, S.-Y. (2000) *Knowledge Based Product Life Cycle Systems: Principles of Integration of KBE and C3P*, Computer-Aided Design, no. 32, page 311 – 320
[13] Sriram, R. D. (1997) *Intelligent Systems For Engineering*, Springer-Verlag, London, UK
[14] Jackson, P. (1999) *Introduction to Expert Systems*, Addison Wesley Longman Ltd, Harlow, UK
[15] CIMdata, www.cimdata.com, access date: 2006-01-20
[16] Stark, J. (2005) *Product Lifecycle Management: 21st Century Paradigm for Product Realisation*, Springer Verlag Ltd, London, UK
[17] Bergsjö, D., Burr, H., Malvius, D., Mueller, M., Vielhaber, M., Malmqvist, J., (2007) *Product Lifecycle Management for Cross-X Engineering Design*, Proceedings of ICED '07, Paris, France
[18] CIMData (2006) *Service-Oriented Architecture for PLM, an Overview of UGS' SOA Approach*, Ann Arbor, MI, USA
[19] Schreiber, G. et. al. (2000) *Knowledge Engineering and Management: The CommonKADS Methodology*, The MIT Press, Cambridge, Massachusetts, USA
[20] Poensich, V., Clark, A. (2006) *Elements of an Interface Standard for Knowledge Sources in Knowledge-Based Engineering*, ASME Journal of Computing and Information Science in Engineering, vol. 6, no. 1, page 78-83, USA
[21] Knowledge Technologies International (2007) www.ds-kti.com, access date: 2007-01-20
[22] Dassault Systemes (2007) www.3ds.com, access date: 2007-01-20
[23] Unigraphics Solutions (2007) www.ugs.com, access date: 2007-01-20
[24] PTC (2007) www.ptc.com, access date: 2007-01-20
[25] Ulrich, K.T., Eppinger, S.D. (2000) *Product Design and Development*, 2nd edition, McGraw-Hill, New York, USA
[26] Cambridge Engineering Selector – CES (2007) www.grantadesign.com, access date: 2007-01-20
[27] Hipple, J. (2006) *The Use of TRIZ Principle in Consumer Product Design*, TRIZ Journal, June 2006, USA
[28] Thilmany, J. (2004) *Share the Wealth*, Mechanical Engineering, v. 126, no. 12, pp. 32-35
[29] Ullman, D.G. (2003) *The Mechanical Design Process*, 3rd edition, McGraw-Hill, New York, USA
[30] Chakrabarti, A., Kota, S., Rao, N., Chowdary, S. (2005) *A Designer Workbench with Real-time Capture and Reuse of Information on Product Evolution and Rationale*, Proceedings of ICED 05, page 496-497, Melbourne, Australia
[31] Regli, W. C., Hu, X., Atwood, M., Sun, W. (2000) *A Survey of Design Rationale Systems: Approaches, Representation, Capture and Retrieval*, Engineering with Computers, no. 16, page 209-235, Springer-Verlag Ltd, London, UK
[32] Georgiev, I., Ovtcharova, J., Georgiev I. (2007) *Modelling web services for PLM distributed system*, International Journal of Product Lifecycle Management, Vol. 2, No.1 pp. 30 - 49
[33] Object Management Group Inc. (2005) *Product Lifecycle Management Services V2.0*, 250 First Avenue, Needham, MA 02494, USA
[34] FIPSPUB 183 (1993) *Integration Definition For Function Modelling (IDEF0)*, Federal Information Processing Standards, Publication 183, National Institute of Standards and Technology, U.S. Department of Commerce, Springfield, VA, USA

Contact:
Amer Ćatić
Department of Product and Production Development
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
Tel: Int +46 31 772 5850
Fax: Int +46 31 772 1375
Email: amer.catic@chalmers.se
URL: www.chalmers.se/ppd/EN/