Computer-Aided Design and Applications

http://www.cadanda.com

Ontologically-based approach for Knowledge Representation for Stamping Die Applications

Margot Ruschitzka[1], Adam Suchodolski[2] and Jerzy Wróbel[3]

[1]Cologne University of Applied Sciences, Margot.Ruschitzka@fh-koeln.de
[2]Cologne University of Applied Sciences and Warsaw University of Technology, Adam.Suchodolski@fh-koeln.de
[3]Warsaw University of Technology, Jerzy.Wrobel@simr.pw.edu.pl

ABSTRACT

The crucial issue in Knowledge-Based Engineering is representation of expert knowledge in a readable way for both a computer agent and human. Properly prepared knowledge (captured, assessed, formalized and structured) plays an important role for success of every knowledge based system; particularly in the discipline of machinery design. Many knowledge model properties are of great significance, in addition to formal structure and consistency, reusability, flexibility, interoperability of knowledge model are also becoming increasingly essential for the knowledge-based systems research community. This paper presents a hybrid knowledge representation for solving deductive engineering tasks in Knowledge Based Machinery Design. We propose two different methods for storing engineering knowledge; as a repository for declarative knowledge, an ontological knowledge based was applied; for procedural knowledge storage as a formal representation, a geometrical model was used. Apart from theoretical foundations the work is also concerned with presenting and discussing an industrial application of the above-mentioned knowledge representation. As a practical part an application for engineer supporting in design process of dies with non-cutting shape forming in automotive field is indicated. This part deals with problems, challenges and bottlenecks coming up during developing process that application and shows advantages and disadvantages of this knowledge representation. For carrying out the effort CATIA V5 as Computer Aided Design system was applied. To fulfill the knowledge model requirements, especially reusability, flexibility and inter-operability, Protégé as an ontology editor and knowledge-based framework with OKBC (Open Knowledge Base Connectivity) protocol was used.

Keywords: knowledge-based engineering, ontology, deductive tasks.
DOI: 10.3722/cadaps.2009.281-289

1. INTRODUCTION

The significant problem of knowledge representation in design has been known for many years. Many researchers have taken up the challenge and have tried to find new directions for knowledge representation in design in order to support a designer in routine engineering processes. Nowadays we know many kinds of supporting systems and in spite of all that facts we cannot say that they are efficient in everyday work of certain engineering fields. Mostly we have problem with too high costs, too complicated maintaining processes, too poor functionality, too difficult handling, etc. A few years ago a big step toward knowledge representation in design (MOKA project) was made. The work of many international researchers brought many new points of view, defined the life cycle of knowledge handling for knowledge-based systems in design and contributed to the acceptance of KBE as a part of CAx. In the MOKA – Project extended Unified Modeling Language (UML) as knowledge representation was used. The scope of semantic knowledge representation utilization is discussed very often, some researchers assumed that every artifact could be represented as knowledge domain [2]; others contradict that [1] and propose building small domains which are easier to maintain. In any case everybody agrees on the importance of knowledge. Figure 1 provides an example of motives for applying knowledge management in German companies.

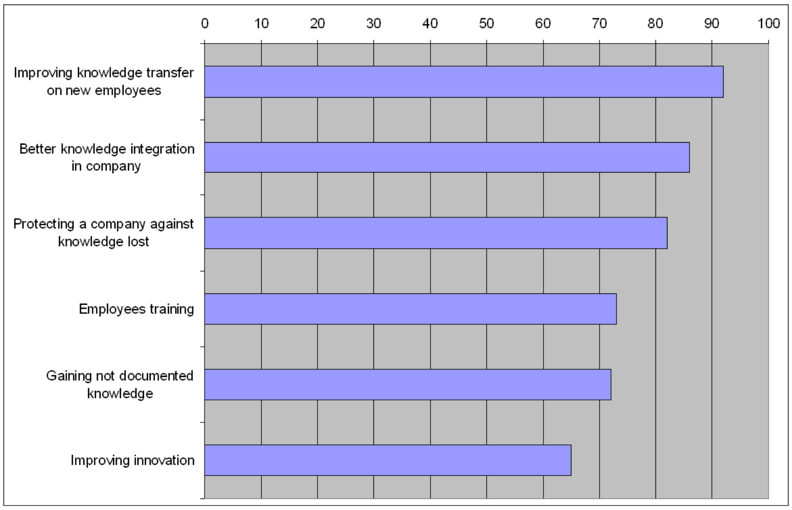

Fig. 1: Motives for knowledge management in 500 interviewed German companies [4] (part of survey).

Nowadays many companies try to manage their knowledge methodically. To achieve this goal, they use different techniques and languages. Because of trying to apply ontologies for semantic services and semantic web as knowledge representation in almost every branch they seem to be the most popular. The uses of ontologies have been summarized by Gruninger and Lee as follows:

- for communication
  - between implemented computational systems
  - between humans
  - between humans and implemented computational systems
- for computational inference
  - for internally representing plans and manipulating plans and planning information
  - for analyzing the internal structures, algorithms, inputs and outputs of implemented systems an theoretical and conceptual terms
- for reuse (and organization) of knowledge
  - for structuring and organizing libraries as repositories of plans and planning and domain information [1]

Applying semantic knowledge representation is very popular for e-commerce, information integration, semantic Web and semantic services. We can find many publications which deal with these issues. Unfortunately we cannot say the same about the automobile industry. The MOKA Project treats some industrial cases but their results were never published. In [1] we find one scenario “where a semantics-based approach is employed for improving the process of testing different configurations of cars”. But represented knowledge does not contain any geometrical items; it deals only with declarative knowledge.

Knowledge representation for geometrical modeling seems to be unsolved. In the following chapters of this work we will present knowledge representation with both kinds of knowledge representation, semantics-based approach (ontological) for declarative knowledge and formally represented procedural knowledge as parametrically driven geometrical model.

Despite non-unified knowledge representation, this hybrid solution brings many advantages, such as: lower cost, smaller knowledge domain, easier maintenance, etc.

2. GOAL OF APPLICATION AND PROCESS OF KNOWLEDGE HANDLING

Taking into account that about 80 per cent of a designer's work has a routine character [2], (Fig. 2) application of knowledge-supported design can bring many benefits (lower designing costs, lower failure rate, shorter time to market, more time for innovation, etc).

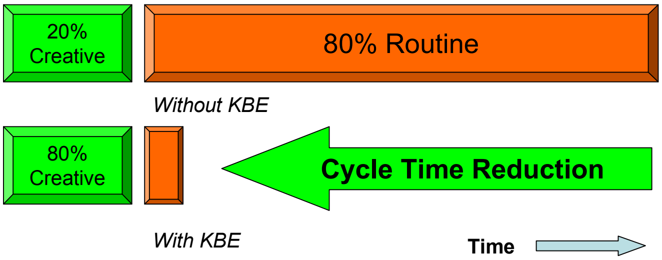

Fig. 2: Division designers' tasks with applying a KBE application and without it [2] (The horizontal axis indicates average design time: the above scenario presents consumed time for engineering **without** applying **KBE**; diagram below reflects time utilization **with** applying **KBE**.)

In reference [3] the developed tool for supported design of trim steels for dies with non-cutting shape forming reduces development time up to 70 per cent (depending upon input parameters), in reference [2] examples of knowledge-based systems from firms Jaguar (interior hood) and British Aerospace (Wing Box) demonstrate design time reductions of up to 99 per cent. Thus both the proper issue identification and fast implementation play a very important role.

In this article we present an application for automated designing process and positioning of trimming elements (punches, trim buttons and related units) for dies with non-cutting shape forming. The goals of the application were as follows:

- to reduce the design process;
- to reduce the failure rate;
- to enable management of knowledge;
- to enable change of some rules and their limiting parameters without programming skills;
- to enable management of knowledge outside the CAD system;
- to enable utilization of one application for many customers' standards; and
- to improve communication between engineers.

Fig. 3: Trimming packet – punch with retainer and trimming button with clamping element.

The time to market in the automobile industry has changed drastically during the last decade (Fig. 4).

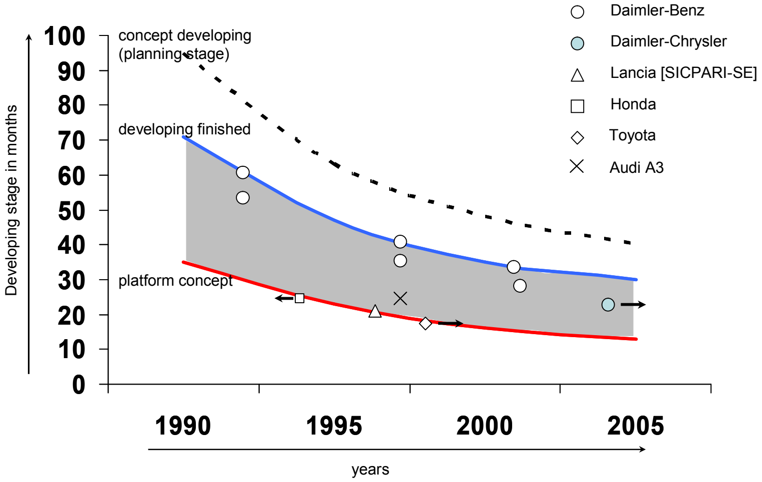

Fig. 4: Shorter developing time for mass production in the automobile branch [12].

Taking into account only the German automobile industry we can see the reduced developing time which has been halved since the 1990s. That fact influenced the design process directly. The process of design a trimming packet is affected by many parameters (thickness of the trim element to be trimmed, material, hardness, function, hole position to work direction of press, hole type, hole size and many others) and because of that, it is very time-consuming. For parts with many different holes the design process of such trimming packets takes up to 15 per cent of whole dies design time. Additionally the trimming element design process and dealing with positioning, making a stock list, ordering, etc., is a typical case which fulfills Pokojski's conditions "that the set of design variables remains the same and each design variable from this set does not change its standards range during the design process" [6] and thereby makes it process-routine.
"Developing knowledge-based applications mostly involves six stages (identify, justify, capture, formalize, package and activate) [2] (Fig. 5). In addition to stakeholders, managers and end-users involved in the process of building of such applications, three others key users (experts, knowledge engineer and developers) are normally involved. They are responsible for the core processes of building such applications."[3]

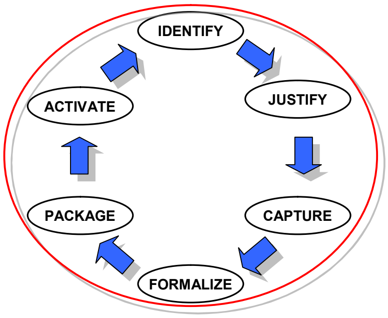

Fig. 5: Lifecycle of Knowledge-Based Applications [2].

In the building process of the trimming elements supported design, the presented lifecycle was adhered to; however several steps were performed in different ways. Four actors (experts, knowledge engineer, general manager and end-user) were involved in the scenario because of task's ad hoc character in process.

Steps "identify" as well as "justify" were combined. At this stage it was analyzed whether the task could be adapted as an automated process, and risks were assessed. Applying automated process design makes sense only under the assumption that the process has a routine character (MOKA et al, 2001; Pokojski, 2004).
The step "capture" is crucial to the whole lifecycle of KBE application [2]. In this step a knowledge engineer and experts were involved. The required knowledge was gained, assessed and structured. The steps from "formalize" to "activate" were carried out by the knowledge engineer. That resulted from two reasons. The first was building the platform (applying geometrical model with driving by an external application). The second one, building the application required skills at three disciplines: design theory, software engineering and ontology engineering. Therefore the knowledge engineers involved had to possess the above-mentioned skills.
The phase of knowledge modeling, due to its hybrid means of knowledge representation was divided into two parts. In the first, a semantic-based approach has been applied (ontological knowledge representation). As an ontology-developing environment, Protégé has been used. Protégé is a free open source ontology editor and knowledge-base framework which enables exporting created ontologies into formats including RDF(S), OWL and XML Schema. In the second part, the procedural knowledge has been structured and formalized. The formal model of procedural knowledge in CATIA V5 as geometrical representation is reflected in this.
Before starting the construction of ontology, the goals of ontology were defined and competency questions were answered. The ontology as knowledge representation for trimming element for dies with non-cutting shape forming was used. The ontological model should be useful for an application that generates output-driven parameters for the CAD system. This semantic knowledge representation should be used and maintained by a trained designer (knowledge engineer).

When describing knowledge capture almost every interviewed expert has used different terms for one and the same thing. Additionally, several automobile producers use only English as the lingua franca and apply different acronyms for different concepts. Because of that, a mapping between the languages and terms was provided. In the subsequent stages of mapping important terms were enumerated, classes and their hierarchy were defined (combination approach of top-down and bottom-up approaches for developing a class hierarchy was applied) and the properties of class-slots and then facets of the slots were defined. Figure 6 illustrates a part of ontology of trimming packets for dies for non-cutting shape forming..

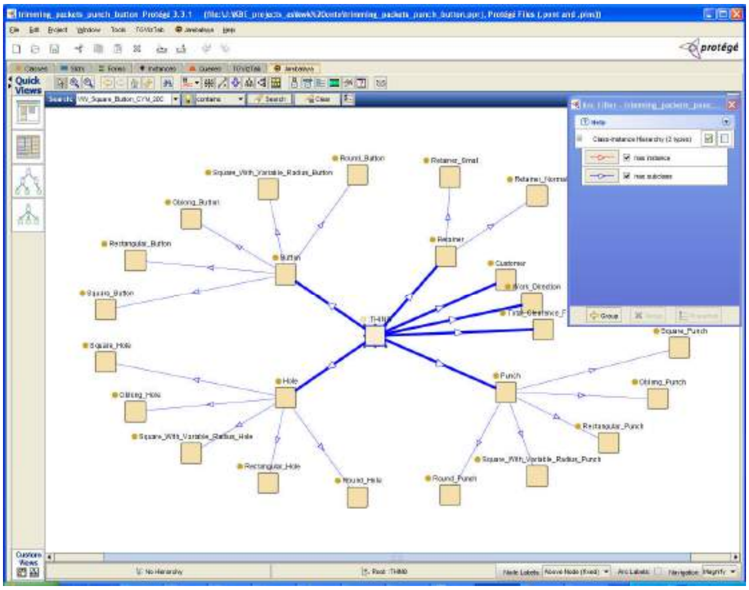

Fig. 6: Declarative knowledge representation as ontology of trimming packages of dies for no-cutting shape forming (classes and subclasses concerning domain of interest).

During the second stage of formal knowledge modeling, certain design theory skills were required. Many scenarios of designing process of trimming packets in CAD package were taken into account. These solutions were discussed, evaluated, structured and afterwards the best were chosen. Because of the ad hoc application character the procedural knowledge as a formal model was reflected. The geometrical formal model contains every variant of standards of every customer. Depending on generated output-driven parameters, some variants have been inactivated or activated in order to achieve the desired geometrical model. Every item of the trimming packet has been communicated with a design table, where the generated output-driven parameters were included.

3. APPLICATION BUILDUNG AND RELATED ALGORITHM

Figure 7 depicts a schematic model of the application. Firstly in the holes' table the input data will be filled in. The holes' table is an excel file, which contains all necessary information (hole parameters, customer, material, work direction, etc.) in order to start computing of output-driven parameters. The input data has to be consistent with the method plan. The knowledge database uses the assigned form hole table information in order to compute driven parameters for the geometrical model. In order to leave the choice of parameters to the designer, the application works sequentially (Fig. 8). At the first stage the input parameters are checked and, depending on their accuracy, are returned to the previous step or computed. The designer can accept or correct them. Afterwards the geometrical model is conformed to the generated driven parameters. The geometrical model is checked against plausibility and, depending on results a correct solution will be proposed or the next set of output parameters for the stock list will be generated. After approval or correction of these parameters, the selection process of suitable independent items will be performed.

The declarative knowledge database consists of ontology which depicts the knowledge domain and a database that contains parameters describing every instance of the knowledge domain. The secondary goal of that database is storage of rules. The knowledge designer is enabled to change these rules according to the new customer standards' release. Due to this solution, the knowledge engineer is not required to have developer skills in order to change the release of that application. Of course the solution has some limitations; in the event of significant changes the code has to be adapted.

The procedural knowledge database is a repository for the geometrical CAD model. The required knowledge is represented as a formal model, but it is comprehensible to any experienced designer. Every geometrical element is driven by many dozens of parameters, relations and rules. Thus, a knowledge engineer can maintain both the declarative knowledge and procedural knowledge database.

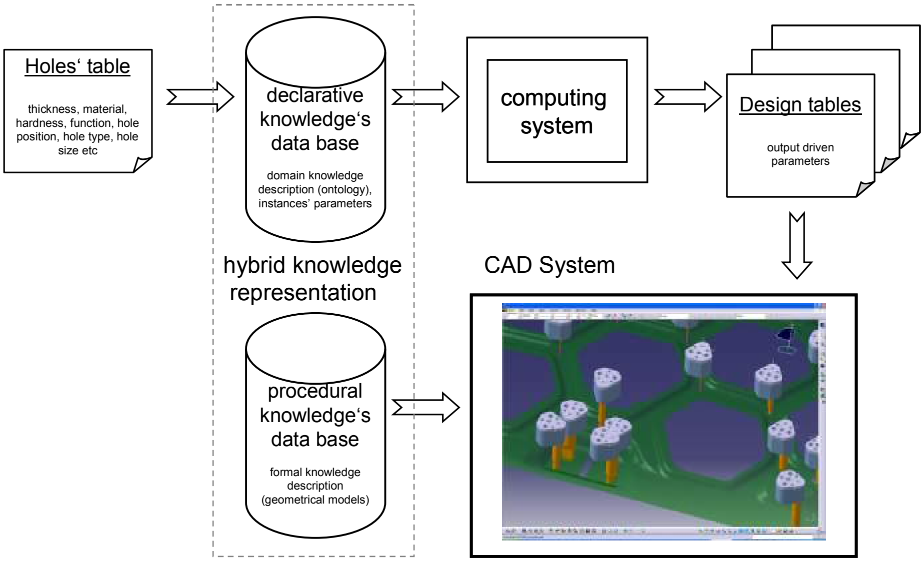

Fig. 7: Hybrid model of knowledge representation – concept.

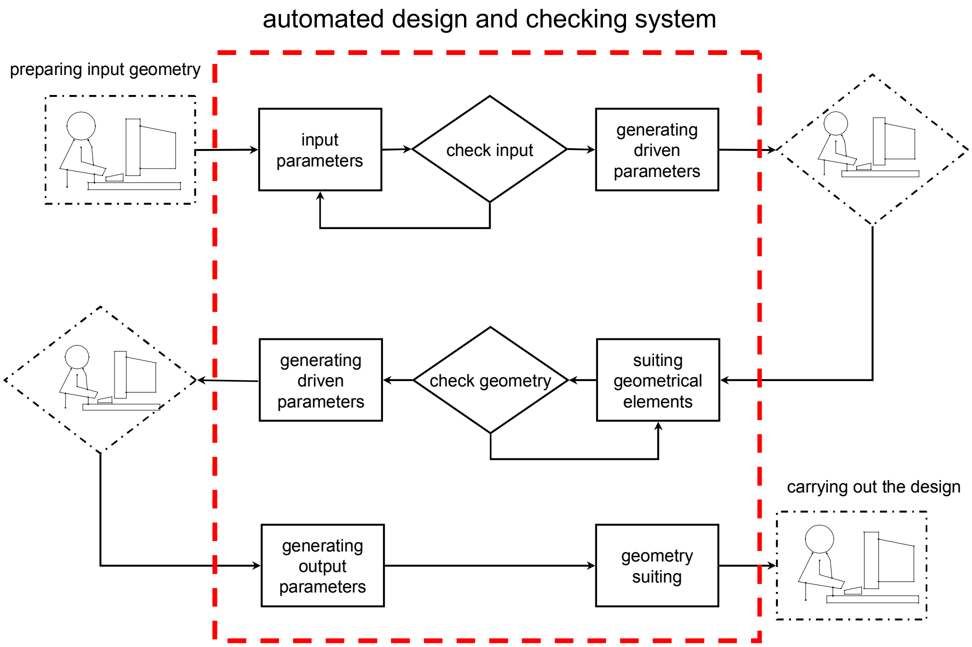

Fig. 8: Application algorithm (engineers in the rhombi symbolize acceptance or correction of generated by system parameters).

4. CONLUSIONS AND SUMMARY

Applications based on previously gained knowledge are found everywhere. Generally such applications are created ad hoc and the construction process is not considered and analyzed properly. As result, many applications have a structure that is not only incomprehensible for the novice but for the developer as well. As many surveys have shown, many projects which have not been analyzed, considered, scheduled and planned fail. It doesn't matter in which discipline. Well-considered and analyzed processes have a greater chance of success. The question "to be or not to be successful" often could be answered before the first significant costs are incurred. Aside from providing more opportunities to build a working application, documentation of knowledge has further advantages. Acquired, structured, assessed and represented knowledge in understandable language for humans and computer agent (or other computer systems) could be used not only for communication between humans (e.g. updating the employees' knowledge about some processes, knowledge exchange, etc.) but additionally for communication between intelligent platforms or design supporting systems.

The process of knowledge handling is very difficult and time-consuming. Very often it is a deciding argument about preparing the right documentation for building a knowledge-based application. Furthermore, the process of building such applications (because of time and costs) starts very often with the package step (Fig. 5) without taking into account the most important stage of knowledge modeling (capture and formalize).

The main goals of this paper were as follows: to emphasize the importance of knowledge for the automobile industry and to present the process of constructing a KBE application using an ontological knowledge base. The solution presented assumed a division of knowledge into two parts – declarative and procedural; the segmentation was made to solve the three groups of contemporary issues in KBE; knowledge representation, high failure rate of support tools for the designer, and high maintenance costs. A declarative knowledge model is independent of any generating CAD application. Thus the significant problem of representation of geometrical items doesn't arise. Thank to applying the software Protégé with OKBC protocol the captured declarative knowledge remains flexible, and interoperable for other ontology-development platforms. Because of a semantic approach, declarative knowledge representation can be built up and maintained by an expert; an ontological approach parallels perception of the real world by humans. Knowledge modeling by experts contributes also to the solution of problems of poor knowledge quality and high maintenance costs. During knowledge capture for both setting up and upgrade of application, the elicitation stage between an expert and knowledge engineer is not required; therefore knowledge quality is accordingly better and upgrade costs lower. On the other hand, we have procedural knowledge represented as a geometrical model. This part of reproduced knowledge typically for every design process is system-dependent, inflexible and not interoperable but is still quite easy to comprehensive for every expert. The part of knowledge contains a high level of parametric and many logical rules in order to drive the geometrical model. Although there were many attempts to make geometry itself interoperable among heterogeneous CAD systems [17] [18] many OEM's break the chain of interoperability intentionally because of protecting their "know how".

Despite many opinions on ontology maturity [1], [10] there is a need for developing tools which would enable construction of ontologies faster and more precisely. There is also a lack of techniques for cost estimation for ontologies. Cost estimation methods for ontology constructing are hard to find at all. "There is a need to perform an in-depth analysis of the cost factors relevant for ontology engineering projects…[1]" Cost estimation methods for software engineering could be applied to ontology engineering (Sure et al, 2008). But the process of ontology construction distinguishes the classes building in software engineering. Thus, the cost estimation methods could be not significant enough in order to produce accurate results. Noteworthy is the fact that cost estimation process for applications where knowledge in different forms is represented (declarative knowledge as an ontology, procedural knowledge as a geometrical model) requires skills in the disciplines design theory, software engineering and ontology engineering.

Further, maintenance has to be considered. Maintenance of pervasive applications, where different pieces of knowledge are represented as formal model in these systems is much more difficult than maintenance of uniform systems. Maintenance of diversified systems where knowledge is distributed requires a developer or knowledge designer with skills in the disciplines covered in the application. However in this case the issue of pervasive applications is not so influential, because the major of the entire system is still carried out by an expert (building knowledge base with ontology-development platform and CAD system). In case of significant upgrades, programmer skills could be necessary in order to customize only the control of the application.

5. REFERENCES

[1] Hepp M.; de Leenheer P.; de Moor A.; Sure Y.: Ontology Management; Semantic Web, Semantic Web Services and Business Applications, 2008, Springer.
[2] Stokes M.: Managing Engineering Knowledge MOKA: Methodology for Knowledge Based Engineering Application, 2001, WileyBlackwell.
[3] Suchodolski A.; Ruschitzka M.; Wróbel, J.: Application of Unified Model Language in Knowledge Based Design of trim steels for dies with non-cutting shape forming Development Trends in Design of Machines and Vehicles, Warsaw, Poland, June 2008
[4] Source www.fraunhofer.de - Frauenhofer Institute for Systems and Innovation Research, 2003
[5] Noy, N. F.; McGuiness, D. L.: Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880 Ontology Development 101: A Guide to Creating Your First Ontology, 2001, Stanford University.
[6] Pokojski, J.: IPA Concepts and Applications in Engineering, 2004, Springer.
[7] Klein, R.: Proceedings of the CIRP Conference on Advanced Engineering Information Systems, ICED'01, Vol. 1, Haifa, Israel, May 2000.
[8] Sriram, R. D.: Intelligent Systems for Engineering, 1997 (Springer).
[9] Schreiber, G.; Akkermans, H.; Anjewierden, A.; de Hoog, R.; Shabolt, N.; Van de Velde, W.; Wielenga, B.: Knowledge Engineering and Management – The CommonKADS Methodology 1999 The MIT Press.
[10] Gómez-Pérez, A.; Fernandez-Lopez, M.; Corcho, O.: Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web, First Edition 2003, Springer.

[11] Davenport, T. H.; Prusak, L.: Knowledge Management – How Companies Manage what they know? 1998, Mcgraw-Hill Professional.

[12] Gerhardt, A.: Wissensmanagement im Externen Simultanem Engineering des Maschinen- und Anlagenbaus – VDI-Berichte 1732 - Wissensmanagement im Ingenieursalltag 2002, VDI Verlad GmbH.

[13] Schuler GmbH A. Handbuch der Umformtechnik; Metall Forming Handbook 1996, Springer.

[14] Heynen, C.: Wissensmanagement im Berechnungsprozess der Produktentwicklung; Fortschritt-Berichte VDI Nr. 341 2001, VDI Verlad GmbH.

[15] de Leenheer P. ; de Moor, A. : do-Science : Context driven disambiguation in ontology elicitation, 1st Context and Ontologies Workshop – AAAI/AAI 2005, Pittsburgh.

[16] Ye, J.; Coyle, L.; Dobson, S.; Nixon, P.: Ontology-based models in pervasive computing systems, The Ontology Review, 22(4), Cambridge University Press, 2007

[17] Dartigues, Ch.; Ghodous, P.; Gruninger, M.; Pallez, D.; Sriram, R.: CAD/CAPP Integration using Feature Ontology, Concurrent Engineering, 15(2), 2007, 237-249.

[18] ProSTEP - Parametrical 3D Data Exchange via STEP – Final Project Report, Version 2.2, 18.12.2006 – www.prostep.org – April, 2009.