INTERNATIONAL DESIGN CONFERENCE - DESIGN 2016
Dubrovnik - Croatia, May 16 - 19, 2016.

DESIGN CATALOGUES: KNOWLEDGE REPOSITORIES FOR KNOWLEDGE-BASED-ENGINEERING APPLICATIONS

P. C. Gembarski, M. Bibani and R. Lachmayer

Keywords: knowledge based engineering, design catalogue, design methodology, knowledge structuring, computer aided design (CAD)

1. Introduction

Increasing demands on the technical performance and quality of new or enhanced products lead to shorter product lifecycles and hence shorter development times. In order to ensure a high level of innovation on the one hand and an efficient adaptation of existing solutions to new fields of use on the other side, the acquisition of actual and valid information and the utilization of existing knowledge are a main challenge in product development [Birkhofer and Keutgen 1999].

This refers both to knowledge about vendor parts or equipment elements as well as about own, already existing, products and product components in terms of solution patterns. Thus, most engineers go back to experience and knowledge from previous projects at the beginning of new ones [Pahl et al. 2013]. Wanke summarizes the benefits of reusing such existing, concrete solutions or solution collections among others as follows [Wanke 2010]:

- Reducing time and cost in product development
- Avoiding double development
- Reducing routine tasks
- Broadening and deepening of own experiences
- Shortening of training times
- Boost of methodological working

App.lying the principles of knowledge-based-engineering (KBE) by implementation of explicit design and process knowledge into digital prototypes has a similar purpose [Verhagen et al. 2007]. App.lications of these principles range from parametric CAD models with implemented mathematical and logical constraints to interactive technical product configurators [Gembarski et al. 2015].

1.1 Motivation

Especially for novice designers, both app.roaches offer a large knowledge base, by means of which they are able to synthesize their own systems from proven (standard) solutions [Winkelmann 2011]. Furthermore, solution collections and expert systems are interesting for the context of open innovation. The term describes a strategy in innovation management, where explicitly external problem solvers in the form of a large, unspecified network of stakeholders are invited through an open call for contributing to a development task [Reichwald et al. 2009].

The amount of information available is not the real challenge in the age of internet and information society, but the structuring and mapp.ing to specific app.lications. Many authors point out that the search for information on existing solutions continues to pose a problem because search efforts are often too

high, the information density is too low and the comparability of information offers from different manufacturers is insufficient [Collin 2001], [Kirchner 2011]. Additionally, existing systems such as online product catalogs, CAD part libraries and design wizards supp.ort the designer in the phases of embodiment design and detailed design. For earlier phases of product development these are, however, only suitable in exceptional cases.
This gap was addressed through Roth and his design catalogues during the 1980ies and 1990ies. These have been developed as a tool for methodological product development and contain collections of known and proven solutions for design problems [Koller and Kastrup 1998], [Roth 2001].
Approaches to devise computer aided design catalogues did not provide the desired results. On the one hand most of these app.roaches solely focused on a computer-based representation of design catalogues [Franke et al. 2004], on the other hand all documented catalogues are set up on part and assembly describing attributes which aid in choosing a component for a given use case [Chandrasekaran et al. 2011]. Further processing of the identified solutions in CAD systems or other development environments has not been presented until now, the same as the connection of catalog systems to KBE-systems. This article bridges a part of these gaps.

1.2 Structure of the paper

Based upon our hypothesis that Roth's design catalogues can be used as knowledge repository or knowledge base for KBE-systems the theoretical background of knowledge-based systems is introduced in section 2. In section 3 Roth's design catalogues are introduced and characterized as knowledge base. Section 4 then presents a framework for a catalogue-based KBE-system which is currently developed by the authors. Finally, section 5 contains a brief summary and drafts further research questions.

2. Theoretical background

Before a KBE-system is modelled, it has to be clarified what type of tasks the system should be able to perform and in which way knowledge has to be app.lied in order to create solutions to a given design problem [Milton 2008]. Firstly, in the following sub-sections different synthesis tasks are presented. Secondly, the overall set-up of KBE-systems is discussed as well as in particular the structure of knowledge bases.

2.1 Synthesis as design task

Generally, design tasks are differentiated into two groups. Analysis refers to all activities where a system or product already exists (to a certain extent) and its behavior or properties are examined by predefined methods. In contrast, synthesis corresponds to all activities where a system has to be constructed according to some given requirements [Schreiber et al. 2000]. More focusing on engineering, Gero defines synthesis as transforming functions to an expected system behavior from which then the system's structures can be derived that fulfil all requirements [Gero 1990].
Most product development processes use a similar separation between synthesis and analysis steps. As an example, the characteristics-properties modelling / property-driven development (CPM/PDD) according to Weber allows both views on the design process. The app.roach is based upon the clear distinction between characteristics (aspects of the design a designer can directly take influence on, e.g. shape, dimensions, material) and properties (aspects of the design which are determined by the corresponding characteristics, e.g. moments of inertia, weight or the fulfilment of a certain stress distribution). Analysis on the one hand is the examination of a product's behavior and the verification that it matches all given requirements. Synthesis on the other side has to be understood as modification of a product's characteristics whereby its properties are influenced and converged to the requirements [Weber 2007]. Furthermore, product knowledge can be expressed by the relation between characteristics and properties. This may be stated then as formulae, tables, linked simulation tools and all kind of expert knowledge [Conrad et al. 2008].
Regarding the possible automation of design tasks or more general the supp.ort of a human designer by knowledge-based systems a further differentiation of synthesis tasks can be made with respect to the particular problem solving methods which are addressed. To those belong [Milton 2008]:

- (Synthetic) Design: Designing a structure that fulfils certain requirements
- Configuration Design: A subset of synthetic design where all components are fully predefined. Another known label of this task is composition.
- Assignment: Creating relations between two groups of objects
- Planning: Generating an ordered set of single activities to meet certain goals
- Scheduling: Creating a schedule of temporally sequenced activities

In times of parametric CAD, there exists another type of synthetic task which is parametrization [Cunis et al. 1991]. Here, a given design has defined degrees-of-freedom regarding dimensions and topological constraints. These have to be eliminated according to given requirements and constraints, e.g. a base frame for a mounting rack which can be varied within certain lengths and heights. Table 1 summarizes the above task types with their inputs, output and knowledge.

Table 1. Synthesis task types (adapted from [Schreiber 2008])

Task Type	*Input*	*Output*	*Knowledge*	*Features*
Synthetic Design	Requirements	Artefact Description	Functions, Components, Skeletal Designs, Constraints, Preferences	May include creative design of components.
Configuration Design	Requirements	Artefact Description	Functions, Components, Constraints, Preferences	All components fully predefined.
Parametrization	Requirements	Artefact Description	Components, Constraints, Restrictions, Preferences	Components have dimensional and topological degrees-of-freedom.
Assignment	Two Object Sets, Requirements	Mapp.ing Set 1 on Set 2	Constraints, Preferences	Mapp.ing need not be one-to-one.
Planning	Goals, Requirements	Action Plan	Actions, Constraints, Preferences	Actions are ordered in time.
Scheduling	Job Activities, Resources, Time Slots, Requirements	Mapp.ing Activities on Time Slots and Resources	Constraints, Preferences	May include buffers.

2.2 Knowledge-Based-Engineering systems

With respect to the above design tasks, KBE-systems mainly tackle those of them which deliver artefact descriptions. According to Chapman and Pinfold, ‘KBE represents an evolutionary step in computer-aided-engineering (CAE) and is an engineering method that represents a merging of object-oriented programming (OOP), artificial intelligence (AI) and computer-aided-design (CAD) technologies, giving benefit to customized or variant design automation solutions’ [Chapman and Pinfold 2001].

As sub-category of knowledge-based systems, KBE-systems consist of the following components [Milton 2008]:

1. Knowledge-Base: Container for all kind of static domain-specific information, structures and rules. In context of mechanical engineering this may be e.g. information about dimensions of standard parts or restrictions due to manufacturing processes.
2. Inference Engine: Separated kind of knowledge for control of the solution space's exploration done by the KBE-system. It describes how to use the knowledge-base in order to solve a given

problem. App.lied on engineering this may be the design and configuration process of an elevator.
3. Blackboard / Working Memory: Holds the instantiation of case-relevant knowledge base elements and all intermediate results.
4. User Interface: Allows interaction between user and KBE-system.
5. Editor: Allows interaction between knowledge engineer and KBE-system in order to alter knowledge base and inference engine.

Additionally for KBE, integration into a computer aided engineering environment has to be maintained. This can be e.g. realized through direct CAD-integration [La Rocca 2012].

2.3 Knowledge representations

Generally, domain knowledge and control knowledge is differentiated [Schreiber et al. 1994]. The first is usually declarative and defines content and structure of the embodied facts. Furthermore, domain knowledge is sub- divided into task-dependent (e.g. dimensioning formula for calculation of a bolt diameter from given loads) and task-independent knowledge (sizes of ISO 2341 standard bolts or the knowledge about what dimensions determine an ISO 2341 standard bolt).

Control knowledge is further classified into inference knowledge and task knowledge. The first describes how to use the domain knowledge in elementary reasoning steps (single inferences like computations or evaluations), the latter contains information about how to implement those inferences together with user inputs and predefined AI-methods like constraint propagation to complex reasoning constructs like semi-automated configuration of cement plant modules [Hvam et al. 2008].

Basically, the formalization of reasoning techniques can be done in three different ways [Sabin 1996]:

1. Rule-based reasoning: The knowledge representation relies to design rules which mirror IF-THEN-ELSE-statements. Those rules are fired procedurally and can execute subordinate rules or delete them from the working memory in order to realize more complex tasks. A major disadvantage of this kind of systems is their lack of separation between domain knowledge and control strategy. As reported by many authors, this results in bad maintainability when the system exceeds a certain amount of rules, so this kind of technique may only be deployed for local and limited problems [Cunis et al. 1991].
2. Model-based reasoning: The limitation of the possible solution space is done based upon a physical and/or logical model (constraint-based) or by representation of resource consumption and allocation (resource-based).
3. Case-based reasoning: In this app.roach, the knowledge representation is not explicitly modelled in form of rules or constraints. The knowledge necessary for reasoning is stored in cases that represent former configurations. Depending on the degree of maturity of the inference engine the system either is limited to search for existing solutions, which match exactly to a given requirements profile, or the system is able to assort a set of existing cases, which represent the best-fit. Highly developed case based systems are able of mixing or altering exiting cases in order to adapt them to new situations.

3. Design catalogues as knowledge repository

In the VDI guideline 2222b the design catalog is presented as information storage, which was specially developed for product development. The design catalog is ought to supp.ort the design engineer in his tasks of acquiring new expertise, organizing it and encourage him to develop new solutions in a methodological way [VDI 1982].

3.1 Structure of design catalogues

Design catalogues are organized in a tabular form and are divided into three types regarding their index structure (Figure 1):

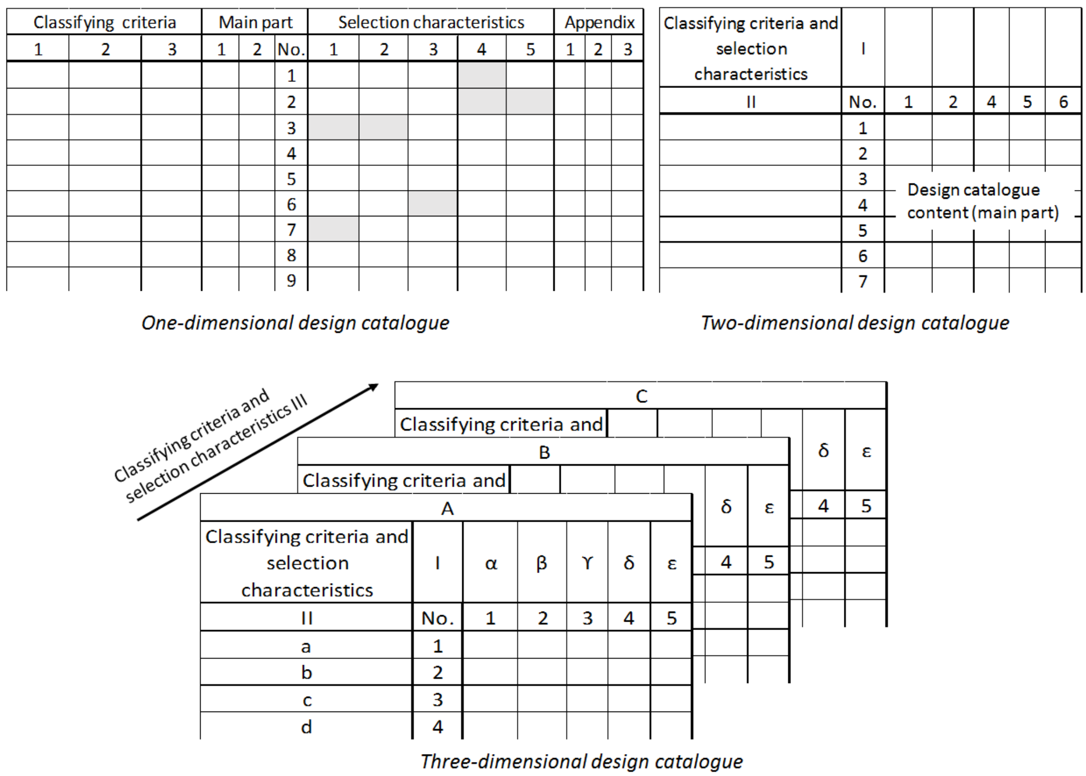

Figure 1. Structure of design catalogues [acc. to Roth 2001]

One-dimensional index structure

This type of catalogue is very common. Regarding setup it is divided into classification part containing the index structure, main part embodying the content and selection / solution criteria. The content in the main part may consist of sketches, equations and texts, ideally with a same level of abstraction. The part containing the solution criteria is in particular characteristic of design catalogs and should be app.ropriate to the purpose [Roth 2001]. Basically, the corresponding solutions can be accessed via the solution criteria regarding given requirements so that the best possible solution is easily identified (e.g. suitability for lightweight design or cost efficiency).

Two-dimensional index structure

The two-dimensional catalog is organized in a matrix form of rows and columns. Through the matrix form the content is displayed in a very compact and clear way, items can be found via two essential classifying parameters. Disadvantageous is the lack of solution characteristics [Roth 2002].

Three-dimensional index structure

This type of catalogue is assembled of multiple two-dimensional catalogues. A solution is found in the crossing of three box index coordinates. The catalog is created on index sheets or books. In times of printed catalogues the solution characteristics were printed on transparent sheets and then app.lied layer by layer to restrict the solution space [Wanke 2010].

3.2 Types of design catalogues

According to VDI guideline 2222b [VDI 1998] and [Roth 2001] design catalogues are further distinguished with respect to their contents. The distinction is made between object catalogs, operation catalogs, solution catalogs and relationship catalogs (Figure 2).

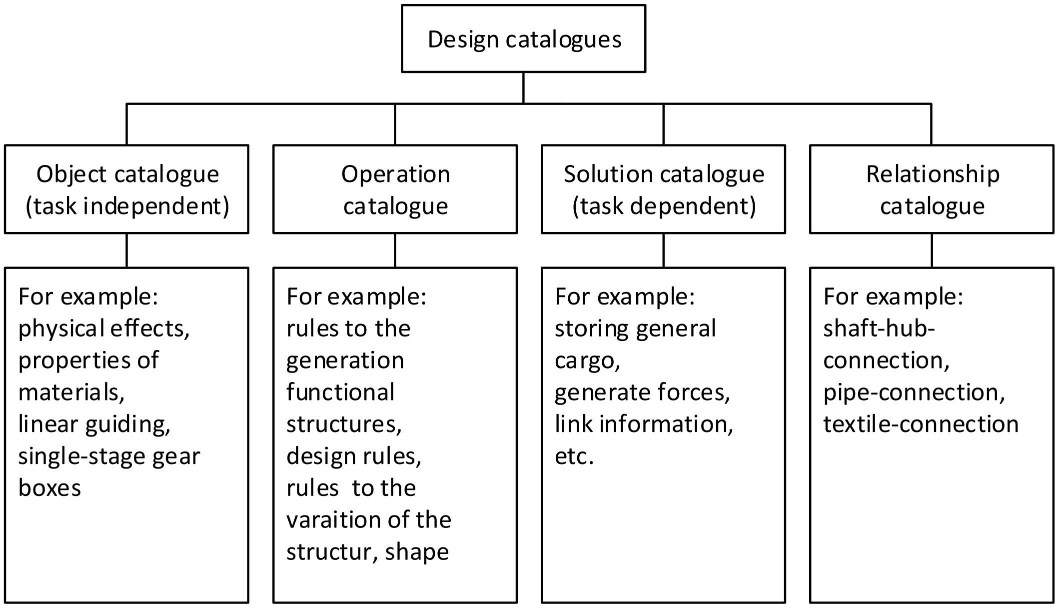

Figure 2. Types of design catalogs with different app.lication examples [acc. to Roth 2001]

Object catalogue

This type is defined as tasks independent design catalog and provides basic facts, especially physical, geometrical, technological nature. This, however, is not assigned to individual functions or tasks. An example of this represent bearing catalogs.

Operation catalogue

Operation catalogues contain operations (process steps) or sequences of operations (methods) that are interesting with regard to design methodology. These include e.g. rules for producing various functional structures and to produce solution variants, as well as methods for solution space exploration.

Solution catalogue

In contrast to object catalogues the solution catalogue is a tool for structuring task dependent knowledge. The indexing for solution catalogues is a function or a class of functions that can be implemented by various means, e.g. changing rotational speed and direction by use of different gear types.

Relationship catalogue

Relationship catalogs are a kind of solution catalog with the restriction to the relationship between two specific objects. The relationship forms a functional solution, e.g. shaft-hub connections.

3.3 Design catalogues as knowledge repository

Regarding the different knowledge representation presented in section 2.3, it easily can be seen that nearly all of them are expressible within design catalogues. Figure 3 depicts a possible assignment.

	Domain Knowledge, Task independent	Domain Knowledge, task dependent	Inference Knowledge	Task Knowledge	Rule-based Reasoning	Model-based Reasoning	Case-based Reasoning
Object Catalogue	X		X		(X)	(X)	X
Solution Catalogue		X	X		(X)	(X)	X
Relationship Catalogue		X	X		(X)	(X)	X
Operation Catalogue		X	X	X	(X)	(X)	X

Figure 3. Knowledge representation in design catalogues

By nature, object catalogues represent the only type of design catalogues which storages task independent domain knowledge. All other types are focused on task dependent domain knowledge since relationship catalogues are a sub-category of solution catalogues and operation catalogues have to be app.lied in a certain design context. Inference knowledge is documented in every class of catalogues. In principle, this is done via the solution and selection characteristics so that depending on the design task or given requirements the content of the catalogue can be assessed and weighted. Task knowledge may only be expressed as operation catalogue. Here, clear task descriptions like e.g. varying solution principles, designing a gear or the translation from functional requirements in design parameters may be documented.

Regarding reasoning mechanisms, all design catalogues are founded on case-based reasoning since the originator of the catalogues expresses experience by the selection characteristics. Nevertheless, all design catalogues may contain rules (e.g. a design rule like IF part is die casted THEN mold release slopes with an angle of two degrees have to be app.lied) or models (e.g. the formula for the calculation of the moment of inertia of a tube) in their main part.

Thus, the synthesis tasks of synthetic design and configuration design are supp.orted by the design catalogue, parametrization not.

4. Concept for a catalogue-based KBE-system

As shown in the last section, design catalogues are meant as knowledge repository for engineering knowledge. In order to use this knowledge in a computer-aided way, the authors are currently developing a catalogue-based KBE-system. The overall aim is to generate a system which is also suitable for the early phases in the product development process where the idea of the design catalogue is continued but linked to a CAE-environment. Finally, the catalogue system is ought to guide the designer to a specific solution but also to document the whole translation process from requirements to the specification of the product including CAD-Drawings. The corresponding framework is depicted in Figure 4. Note, that only the system architecture is described below, a detailed discussion of all data and information flows within the system is beyond of the scope of this paper.

The computer aided design catalogue mainly consists of the three traditional parts index structure, main part and selection characteristics which are setup as database system like in previous documented app.roaches. Therefore, an object oriented, relational database is established with SQL. Tables and queries are organized according to previous implementations of computer aided design catalogues like shown in [Kirchner 2011]. New is the link to a case-base, where requirement profiles are stored. These profiles are used to configure the selection characteristics for the targeted purpose so that solutions may be assessed and weighted with respect to certain design boundaries (e.g. lightweight design or total cost reduction). The case-base itself is also a database system where requirements and the corresponding selection characteristics are stored. If no precise match is found, the system is able to retrieve similar cases. E.g. when designing a milling machine, the case-base could contain requirements for speeds, forces, precision, wear behaviour and energy consumption of drive systems and linear axis. Then the selection characteristics filter the solutions available within the main part of the design catalogue.

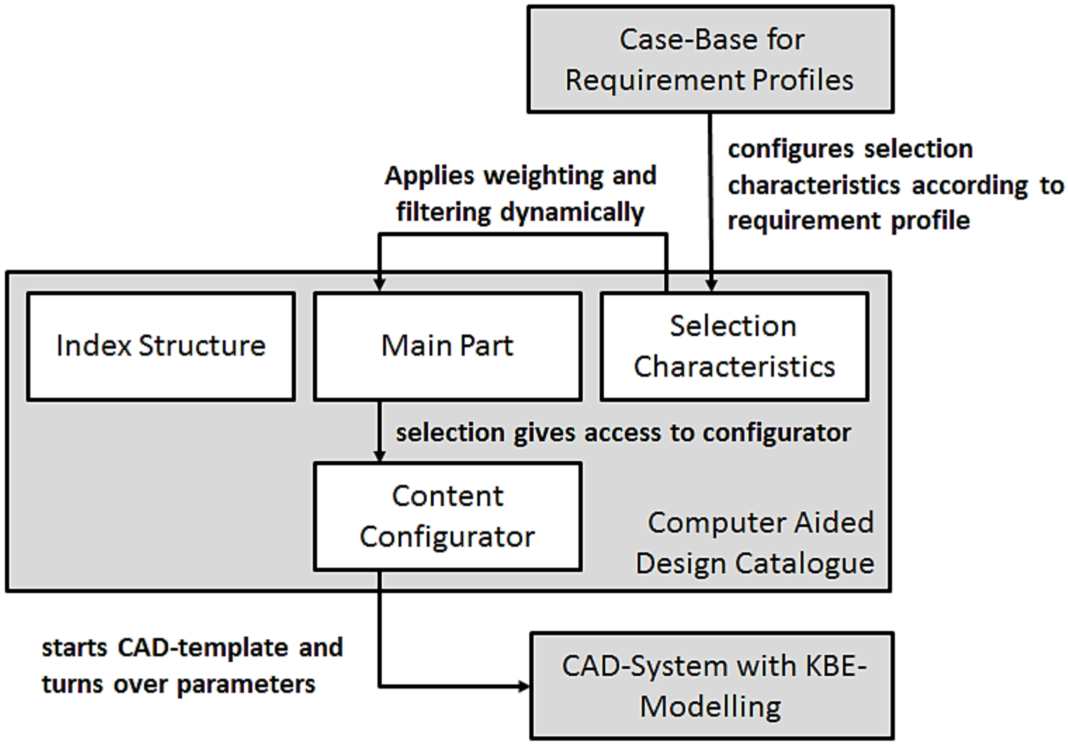

Figure 4. Framework for catalogue-based KBE-system

Additionally in contrast to traditional design catalogues, the developed systems consist of a content configurator. That means, that all solutions found in the main part of the design catalogue have own configuration models which consist of either constraint-based or resource-based models. The configurator checks all inputs for consistency and informs the user about properties of the configuration. Referring to the example with the milling machine, the designer can adapt dimensions and other characteristics of a chosen drive system, linear guiding and the axis. For implementation in the design catalogue different possibilities may be chosen. One way is the implementation of a spreadsheet environment like MS Excel, simple and stable models may be hard coded within the content configurator with own GUI.

The gathered properties and parameters are then transferred to a CAD-template, which has KBE-functionalities implemented. Depending on the used CAD-system multiple ways of implementation may be addressed. To those belong the direct input of parameters into the CAD-model via macros or remote app.lication or the exchange of parameters via external parameter control mechanisms like excel spreadsheets which can be imported into the CAD-system (e.g. the CAD-system Autodesk Inventor features all these capabilities, refer also to [Gembarski et al. 2015]). Afterwards, calculation and dimensioning formulae process the parameters from the catalogue system, design rules activate the necessary features so that the CAD-model is automatically reconfigured and rebuilt faultlessly. For most app.lications this template creates am embodiment design where the engineer then can start the detailed design from. Nevertheless, a complete design automation is possible for certain, limited app.lications (e.g. in fixture design), so that a complete CAD-model with all corresponding drawings for documentation and manufacture is generated. Generally, by this the lack of supp.orting the synthesis task of parametrization is eliminated through the computer-aided catalogue system.

Beside the milling machine example, demonstrators for robot-gripp.ers and for agitator systems are currently developed and implemented.

5. Conclusion

In the present article a framework for a computer-aided design catalogue was presented. In order to supp.ort different kind of synthesis tasks in engineering design, the catalogue is linked to KBE-models that can adapt to requirements and parameter changes.

The mentioned framework is still in development, first tools have been implemented around the CAD-system Autodesk Inventor. As a next step, the user interfaces for the case-base and the content configurator will be designed. Furthermore, interfaces to other CAD-systems have to be defined.

As research question it has to be examined in which way the described framework can be used to accelerate design processes and what level of abstraction has to be used for the CAD-templates. The point of interest here is the question to what extent the templates have to be detailed so that efforts for creating the templates and for configuring and detailing the solutions are in balance.

Acknowledgments

The results presented in this paper were obtained under the umbrella of Collaborative Research Centre 1153 „Process Chain for Manufacturing Hybrid High Performance Components by Tailored Forming", preliminary inspection project C2. The authors would like to thank the German Research Foundation (DFG) and the CRC 1153 for its financial and organizational supp.ort.

References

Birkhofer, H., Keutgen, I., "Vom Konstruktionskatalog zum agentengestützten Online-Informationssystem", Konstruktionsmethodik—Quo Vadis, 1999, pp. 43-52.

Chandrasekaran, K., Djuric, A., ElMaraghy, W. H., "Selection Catalogue of Kinematic Configuration for Pick and Place App.lication", Enabling Manufacturing Competitiveness and Economic Sustainability, Springer Berlin Heidelberg, 2012, pp. 41-46.

Chapman, C. B., Pinfold, M., "The app.lication of a knowledge based engineering app.roach to the rapid design and analysis of an automotive structure", Advances in Engineering Software, 2001, pp. 903-912.

Collin, H., "Management von Produkt-Informationen in kleinen und mittelständischen Unternehmen", Techn. Univ. München, 2001.

Conrad, J., Köhler, C., Wanke, S., Weber, C., "What is design knowledge from the viewpoint of CPM/PDD?", Proceedings DESIGN 2008, the 10th International Design Conference, Dubrovnik, Croatia, 2008.

Cunis, R., Günter, A., Strecker, H. (eds.), "Das PLAKON-Buch: ein Expertensystemkern für Planungs-und Konfigurierungsaufgaben in technischen Domänen", Springer Berlin Heidelberg, Vol.266, 1991.

Franke, H.-J., Löffler, S., Deimel, M., "Increasing the efficiency of design catalogues by using modern data processing technologies", DS 32: Proceedings of DESIGN 2004, the 8th International Design Conference, Dubrovnik, Croatia, 2004.

Gembarski, P. C., Li, H., Lachmayer, R., "KBE-Modeling Techniques in Standard CAD-Systems: Case Study – Autodesk Inventor Professional", Managing Complexity - Proceedings of the 8th World Conference on Mass Customization, Personalization and Co-Creation (MCPC 2015), Montreal, Canada, 2015.

Gero, J. S., "Design prototypes: a knowledge representation schema for design", AI magazine, Vol.11, No.4, 1990, pp. 26-36.

Hvam, L., Mortensen, N. H., Riis, J., "Product customization", Springer Science & Business Media, 2008.

Kirchner, K., "Entwicklung eines Informationssystems für den effizienten Einsazu generativer Fertigungsverfahren im Produktentwicklungsprozess", Techn. Univ. Braunschweig, 2011.

Koller, R., Kastrup, N., "Prinziplösungen zur Konstruktion technischer Produkte", Springer, 1998.

La Rocca, G., "Knowledge based engineering - Between AI and CAD - Review of a language based technology to supp.ort engineering design", Advanced engineering informatics 26.2, 2012, pp. 159-179.

Milton, N. R., "Knowledge technologies", Vol.3, Polimetrica sas, 2008.

Pahl, G., Beitz, W., Feldhusen, J., Grote, K. H., "Pahl/Beitz Konstruktionslehre", Heidelberg, Berlin, Springer, 2013.

Reichwald, R., Piller, F., "Interaktive Wertschöpfung", Gabler, Wiesbaden, 2009.

Roth, K., "Konstruieren mit Konstruktionskatalogen - Band II Konstruktionskataloge", Springer-Verlag Berlin Heidelberg New York, 2001.

Roth, K., "Design catalogues and their usage", Springer London, 2002.

Sabin, D., Weigel, R., "Product configuration frameworks-a survey", IEEE intelligent sys-tems 13.4, 1998, pp. 42-49.

Schreiber, G., "Knowledge engineering and management: the CommonKADS methodology", MIT press, 2000.
Schreiber, G., "Knowledge Engineering", Handbook of knowledge representation, van Harmelen, F., Lifschitz, V., Porter, B. (eds.), Elsevier, Vol.1, 2008.
Schreiber, G., Wielinga, B., de Hoog, R., Akkermans, H., van de Velde, W., "CommonKADS: A comprehensive methodology for KBS development", IEEE expert, Vol.9, No.6, 1994, pp. 28-37.
Vajna, S., Weber, C., Bley, H., Zeman, K., "CAx für Ingenieure", Springer-Verlag Berlin Heidelberg, 2009.
Verein Deutscher Ingenieure, "VDI-Richtlinie 2222, Blatt2, Konstruktionsmethodik, Erstellung und Anwendung von Konstruktionskatalogen", Beuth Verlag, Berlin, 1982.
Verhagen, W., Bermell-Garcia, P., van Dijk, R. E., Curran, R., "A critical review of Knowledge-Based Engineering: An identification of research challenges", Advanced Engineering Informatics 26.1, 2012, pp. 5-15.
Wanke, S., "Neue Konzepte zur Verwaltung und Bereitstellung von Lösungen im Produktentwicklungsprozess: CPM/PDD-Lösungsmuster als Grundlage eines verhaltensbeschreibenden Lösungskataloges", Universität des Saarlandes, 2010.
Weber, C., "Looking at "DFX" and "Product Maturity" from the Perspective of a New App.roach to Modelling Product Development Processes", The Future of Product Development - Proceedings of the 17th CIRP Design Conference, Krause, F.-L. (ed.), Springer-Verlag Berlin Heidelberg, 2007, pp. 85-104.
Winkelman, P., "A theoretical framework for an intelligent design catalogue", Engineering with computers 27.2, 2011, pp. 183-192.

Paul Christoph Gembarski, Dipl.-Ing. (FH)
Leibniz Universität Hannover, Institute for product development
Welfengarten 1a, 30167 Hannover, Germany
Email: gembarski@ipeg.uni-hannover.de