Product Knowledge Management and Design Support[1]

Penschke, St.; Ort, A.

This paper will identify major strategies on how product knowledge can be used during design and how it can improve the quality of a product. The resources used for this analysis are European and national research projects (the IMW participated) in the field of design, manufacturing and quality assurance. Taking their field of research and their objectives into account, the developed strategies will be extracted and analysed. In parallel, the application of such results will be reviewed. Based on this analysis, the impact and gained efficiency will be described. This will result in a perspective of how such strategies and applications can improve a product.

Der vorliegende Beitrag identifiziert wesentliche Strategien für die Erfassung und Aufbereitung von Produktinformationen und ihre Nutzung im Konstruktionsprozeß. Grundlage für die Analyse bilden Europäische und nationale Forschungsprojekte (an denen das IMW beteiligt war und ist) auf den Gebieten Konstruktion, Fertigung und Qualitätssicherung. Unter Berücksichtigung der Projektziele werden die zugrundeliegenden Vorgehensweisen und entwickelten Ansätze für das Produktdatenmanagement extrahiert und analysiert. Dazu werden einleitend die einzelnen Projekte und resultierende Applikationen (Prototypen) vorgestellt.

1 Introduction

The work with design support systems is basically initiated by the need to assure competitiveness in the future for manufacturing enterprises. Many competitive factors have to be taken into consideration. In general product costs, product quality and the delivery at the right time are the key factors. The pressure to shorten the time to market for new products and therefore to shorten developement time requires an adequate use of know how possessed by a company.

In manufacturing enterprises the product knowledge is distributed across the whole company. This know how represents an essential resource for successful competition in the market and should therefore be preserved and used as efficiently as possible. A way of reaching this goal exists in the use of so called knowledge based systems which contain accumulated product knowledge on different aspects during the product life cycle. Besides, product data modelling and management has become more and more an important feature for enterprises to strengthen their competitive position /1/.

2 Application examples – a case study

Starting point for the case study presented here was to check the possibilities for a harmonisation of developed research prototypes within the IMW and to discover main goals for future work. The following chapters will give an analysis of projects, research prototypes including short descriptions, especially focused on handled product knowledge and aspects of design support.

2.1 PICASSO (Practical and Intelligent CAD for Assembly Objects)

The focus of PICASSO (Brite-EuRam Project 5693) was the development of a design methodology and a supporting CAD tool that makes information on tolerances, machine components, process tools etc. available to the designer and leads him through the design process. This system was applied, at first, in the design of plastic spray moulding and press tools.

PICASSO offers the designer a choice of assembly types and leads him through the design, component by component. As each part is chosen, it constrains the choice of subsequent components. The PICASSO system consists of a component definition module (allowing a catalogue of components to be defined), an assembly definition module (allowing a database of assembly class definitions to be defined, including rules constraining the way in which components fit together), an assembly design module (allowing a particular instance of a class of assembly to be designed), and a functional tolerance module (determining tolerances automatically

[1] Der Artikel basiert auf einem Beitrag zum Workshop ProKSI '97 – Product Knowledge Sharing and Integration, Sophia Antipolis, April 1997

from component function and distributing tolerances between mating parts).

The main information handled by the system are tolerances and knowledge about how mould components are assembled. They are represented in rules and formulae. The functional tolerance module represents the whole ISO 286. The assembly definition module represents all information concerning geometry and relationships of components. This module is used to check if and how components fit together and also how they fit together in a assembly (mould). A super user, the designer and in few cases the manufacturer, must feed the system gradually (step by step). The system is used during embodiment and detailed design.

The software was built up in C++. The knowledge is totally incorporated by the CAD system. Only the funcional tolerance module was programmed as a module which can be placed in any other software. During the project about 14000 parts and about 4200 rules of moulds were adapted to the PICASSO system.

2.2 EQUIP (Work Methodology for Development of Quiet Products)

The main objective of the EQUIP (Brite-EuRam Project 5983) project was to provide the designer with a consulting system to obtain low noise products from well conducted design. A work methodology and tools for industry were developed, that will enable manufacturers to reduce noise levels, time to market, development and material costs. The products of companies involved were earthmoving machines, cooling equipment, heating and airconditioning systems and municipal vehicles.

Main achievements are the development of an approach for designing low noise products, the establishing of a model for this approach (SADT), the definition of necessary information for each step of designing low noise products, the characterization of the usage of the model for software definition and the definition of libraries for different kinds of knowledge (e.g. requirements, solids and fluids properties, measurement methods, component instances, formulae). An approach of creation of noise path models was developed and successfully tested. A way to propose and select suitable noise control measures was determined. A method for the elaboration of acoustic requirements for a machine has been built up.

The developed system (**fig. 1**) can be updated and customised for branch specific applications (default systemlibraries and branch specific company libraries). The modeller is a major part of the system and provides the functionality for a component based noise path modelling, calculations, visualisation of result (e.g. noise level ranking and sound power spectra for several components) and optimisation of a model. The scenario manager consists of a visualiser and an execution mechanism. It is an independent program that is intended to help the user build and examine noise path models. Scenarios could be created by manual writing, system logging of user actions or generation from graphical process schema (like SADT models). A scenario is a script like description which allows the scenario manager to provide guidance to a designer in performing a complex task.

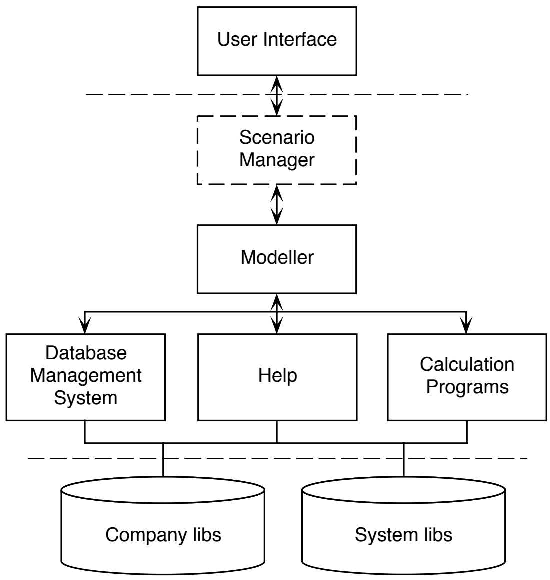

fig. 1: Concept for EQUIP system

The tools and methods developed were to be validated by the design departments of the companies involved in the project and applied to the products mentioned, thereby achieving the noise level reductions required.

2.3 PLUS (Parts Libraries Usage and Supply)

The portability of parts libraries is a major economic concern for CAD system users, component manufacturers and for CAD system vendors. To allow such portability a whole set of concepts, known as the P-LIB approach, has been developed in Europe. The objectives of PLUS (ESPRIT Project 8984) were to validate, possibly improve the P-LIB approach and provoke the practical use of this approach through a complete set of pre-industrial tools as well as contribute to the final standardization work through demonstration of the concepts, preparation of pre-normative specifications as input to European and international standardization.

Parts libraries allow the definition and description of parts for the design of products. Using a standardized description format, the concept enables the comparability and exchange of parts. Through the structured description of parts and sophisticated search strategies during embodiment design the selection of repeated parts will be made easier. Moreover, the standard allows a multi-supplier search, i.e. the search for parts in various supplier catalogues. The focus is the embodiment design with respect to repeated parts (costs).

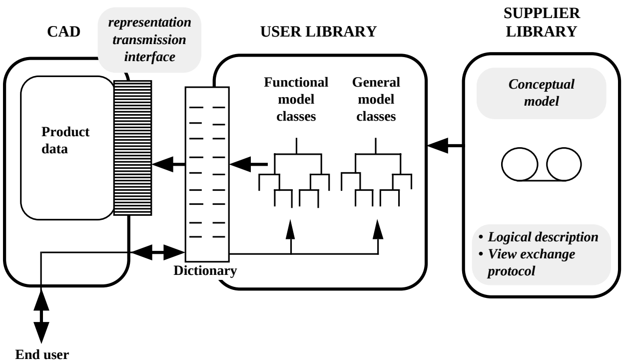

fig. 2: Concept for part libraries

Main results of the project were the development of three pre-competitive library management systems which support the new capabilities resulting from the P-LIB approach and the migration from previous proprietary or standardized systems, the development of a partial description of the components of a part supplier and the development of a practical know-how about the P-LIB approach among the project partners. These results make it possible to demonstrate the interactive generation of a part library, the exchange and automatic integration of this library by library management systems related to different CAD systems /2/. A parts library system (**fig. 2**) is connected to a CAD system and will serve as a parts catalogue. Parts can be selected and inserted into the current design.

The theoretical background is, that parts are ordered in a tree hierarchy and defined by properties (applied to each node in the hierarchy). The approach makes use of object oriented concepts. The data itself is represented as entities and relations. The data structures can cope with the highly complex organized data, but the approach of managing the data requires deep knowledge and may not result in best performance. The applicability and the efficiency of the approach has still to be proved.

2.4 AMANIS (Advanced Manufacturing Information System for the Designer)

The aim of the research project AMANIS (Brite-EuRam Project 5139) was the development of an approach which allows the collection and preparation of current company-specific manufacturing information as well as to provide this information to the designer in a convenient way. Based on this provided manufacturing experience, the manufacturing properties of future designed products should become more transparent. AMANIS was focused on metal removing manufacturing processes.

The main information handled in the system are manufacturing times, costs and problematic events to be expected during manufacturing. Knowledge about optimised and actually used NC-programs are collected by manufacturing data collection systems and represented as tables representing manufacturing cases. The system is used during embodiment and detail design. Developed tools for knowledge management and representation are NC-Analyzers for the extraction of relevant information from NC-programs, a tool for the induction and representation of NC-sequencing rules based on algorithm for autonomous knowledge acquisition, a manufacturing-oriented feature recognition tool based on algorithm for autonomous knowledge acquisition and a manufacturing-oriented CAD-model classifier based on neural network techniques. The tools are integrated within the CAD-system Pro/Engineer for extraction of relevant CAD-information from part and feature models in order to both populate the case base and reason about newly designed part and feature models.

In general, the practical experience gained in the project was, that techniques for autonomous knowledge acquisition can help to reduce the efforts during creation and administration of knowledge bases. However, the training and subsequent unsupervised knowledge acquisition requires a lot of consistent cases. Adaptations to new situations are carried out with a certain time gap. As far as AMANIS is concerned, the case base mainly represents information from a dynamic and unstable environment which results in reduced reliability of the provided information to the designer who is not able to interpret the results with respect to their correctness.

2.5 QCIM-PM (Quality through CIM - Product Model) - Registration and analysis of requirements

Within the QCIM project the IMW developed a prototype tool for the capture and analysis of requirements in the design process. This tool is based on an integrated product model to support quality management in product design. The product model was developed in the national german project Quality through CIM (QCIM). Main aim of the tool named demanda is the handling and visulization of complex, multidimensional requirement structures and interrelations to support the designer in the early phases of the design with the reuse of already existing knowledge from all product life cycle phases.

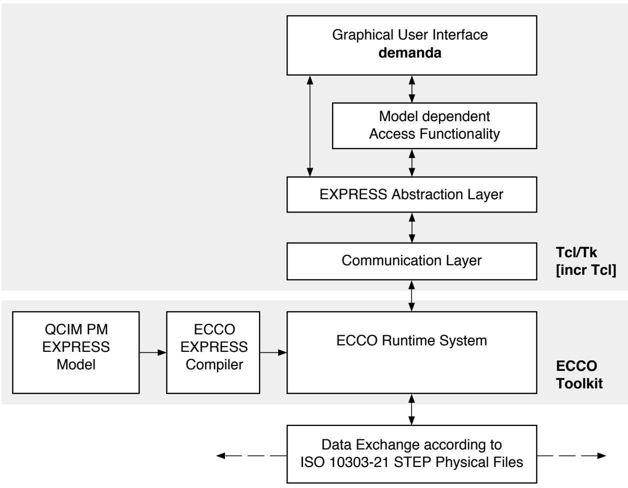

fig. 3: Implementation concept of demanda

Demanda uses a Toolkit for the implemention of the QCIM product model and the standard data exchange (**fig. 3**). The QCIM product model handles information from all phases of the design process. Requirements, functions, solution principles and geometric information are fully integrated and can be weighted. Therefore, when the designer defines a requirement he can access already existing weighted solutions stored in the product model. One speciality of the model is the concurrent handling of the actual design requirements for a specific product and the solutions already in the database. The product model does not only capture one set of requirements but the whole series of dynamic changes to the requirements which represent the flow of the ongoing design.

Knowledge is stored in several facettes. Requirements are defined by their properties. Properties can be stored as a textual description or as mathematical expression. The interrelationships provided knowledge about the interactions between two requirements. The most importent knowledge stored in the product model are the definitions. Definitions in the product model are the super class of all entities in the design process namely requirements, functions, solution principles and geometry.

The main goal in implementing demanda was to test the theoretical approach of multidimensional requirement structures and relations. One main insight was that the designer does not think in the way that the product model handles the data. Therefore the designer has to be supported by the user interface when structuring the requirements. The requirement interrelationship is a major problem, since the designer is seldom aware of these often conflicting interrelationships. Further research will focus especially on the automated detection of interrelationships.

2.6 Sheet metal design for manufacturing

The special research project 'processing of sheet metal' (SFB 362) is a common project with the University of Hannover under the leadership of the technical University of Clausthal. Aim of the project is to investigate the basic material and technological processes of forming and joining of sheet metal. The IMW is working on a subproject 'sheet metal design'. Aim is the development of basic design methodologies for sheet metal products to meet the requirements of function and manufacturing.

After several analyses of practical sheet metal use concerning products, design processes and manufacturing technologies the main information was extracted and systematised. Different criteria for an efficient support of sheet metal design processes were recognised. Information on materials and manufacturing that determine geometrical parameters must be realised if possible in early phases of the design process for a further consideration. Therefore material and technological information must be represented adequately to support the transition from product planning and conceptual design to embodiment and detail design. The representation of such information should permit an interactive optimisation of geometric parameters with regard to material and manufacturing.

The developed system concept (**fig. 4**) is characterised by several layers such as the integrated product model containing all data describing the product with declarative and procedural knowledge (e.g. requirements, geometry, materials, manufacturing technologies or calculation methods) and the tools for knowledge adaptation and presentation. A tool for requirements handling allows the designer to register requirements in textual form and to convert these requirements into references or constraints on desired or required product properties (e.g. preferred materials, manufacturing technologies, machines or geometries). By means of editors and search mechanisms the designer will be supported in conversion of requirements into references or constraints. The steering mechanisms for calculation methods allows an administration of formulae, their solution in all feasible directions and their use according to the design progress by means of priorities. A tool for pre-definition of manufacturing processes allows an interactive optimisation of geometrical, material and manufacturing parameters for deep drawing processes. The designer can set parameters or choose them from referenced requirements. Undetermined parameters will be calculated if possible. The validity of referenced requirements will be checked during the pre-definition process. For instance: if various drawing presses are preferred, the system checks the manufacturability of the part by means of the individual presses. The results of these checking processes will be visualised in a suitable way. As a result the designer has not only specified the geometry of the part but also bound possible materials and manufacturing processes and machines.

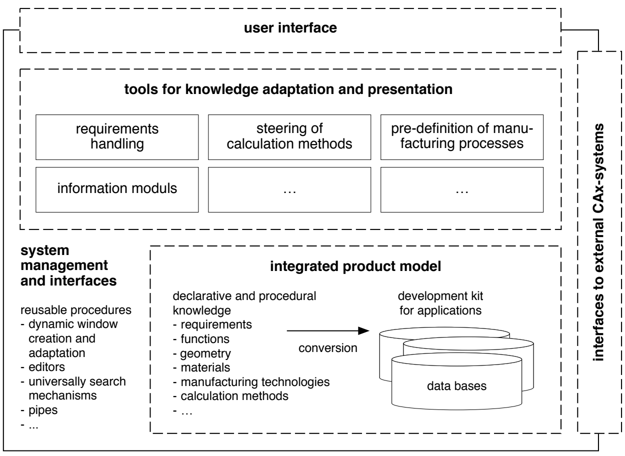

fig. 4: Concept for sheet metal information system

The data base was filled with some test records. In general the approach was useful. One insight was, that the designer reaches an optimum dependent on his individual experiences. There is no methodical help like optimisation strategies or similar. Further research will focus especially on the provision of a higher semantic level through a feature based approach to support methodical operating /3/ and on the recovery of manufacturing experiencies to support particularly the determination of tolerances.

3 Analysis of handled engineering knowledge

The described systems are focused on small domains of certain disciplines. The range of systems varies from ‘simple’, specific knowledge based systems (used in one specific phase of design) to complex, self acquisitioning knowledge based system with active feedback from later product life cycle phases.

The first aim in analysing these systems is to reuse experiences gained in the projects, i.e. concerning information modelling, applicability and usefulness of developed approaches. This includes the reuse of information models or parts of the models and of system components or specific tools. A second aim consists in a detection of general strategies for provision, processing and use of product knowledge during the design stage for a capable decision support. From this result two considerable tasks:

1. One needs to describe/characterise/systematise the knowledge which is handled in the systems in order to compare them and to discover possible overlaps.
2. One needs to detect major approaches concerning knowlege handling and adaptation in the sense of DFX (design for x) respectively the feedback of knowledge to the designer.

3.1 Systematisation of engineering knowledge

There is a multitude of exertions dealing with the problem of systematisation of knowledge in general and especially of engineering knowledge. For instance each research on design object representation can be regarded more or less as an effort to systematise design knowledge. Therfore the following small survey could surely not be complete. In general all of these efforts represent different views on the problem, taking several aspects into account.

One important point of view may be human memory structures and psychological interrelationships in problem solving. So the cognitive structure of a human distinguishes in two fields, the epistemic and the heuristic structure /4/. The first contains a system of categories to structure the knowledge about facts and applicable operators. It determines the capability to solve tasks reproductively. The epistemic structure may be interpreted as a kind of semantic network whose nodes represent contents and whose connections are relations between the contents. The heuristic structure represents a library of procedures (methods) for problem solving, i.e. a system of meta or inside operators to build operators. It will be used if no solving method is retrievable immediately. In order to systematise engineering knowledge this approach seems to be too abstract. Nevertheless, it is useful to support information modelling and structuring of knowledge bases analogously to the human memory.

Another point of view is from the field of artificial intelligence (AI). Resulting from this are a number of distinctions concerning knowledge. One distinction follows the general subdivision of AI systems into three broad categories /5/. The knowledge about a problem domain that is represented in a global database is called declarative knowledge. It would include specific facts like data and relations between the data. The knowledge about a problem that is represented in a set of rules is called procedural knowledge. It would include general information that allows a manipulation of the declarative knowledge. The knowledge of a problem that is represented by a control strategy is called control knowledge. It includes a variety of processes, strategies and structures used to coordinate the entire problem solving process.

Elsewhere /6/ the kernel of intelligent knowledge based systems should consits of a knowledge base containing knowledge about a problem domain (e.g. facts, information, rules of judgement; also called domain) and an inference mechanism for manipulating the stored knowledge to produce solutions to problems (also known as inference engine, control structure or reasoning mechanism; also called task). Ideally domain and task are independent.

In addition there are several efforts to systematise knowledge from the engineering domain. Following an analysis of main sub-activities in design /7/ introduces a division of engineering knowledge into:

- documentation of design considerations (e.g. sketches and drawings),
- modelling/drawing (usually based on CAD systems),
- textual information (e.g. descriptions, specifications, instructions),
- rules/calculations (e.g. re-design of complex mechanisms like bearings or gears),
- materialinformation (to explore and select materials) and
- process information (to take the manufacturing implications into consideration).

The information handling behaviour of the designer could be another criteria for knowledge systematisation. To observe empiric experiments a framework for describing informational behaviour is presented in /8/. To classify so called information fragments serveral definitions are made:

- informational activity (e.g. generate, access, analyse),
- descriptor (e.g. requirement, operation, location, comparison, alternative, relation),
- subject class (e.g. requirement, concept, assembly, component, connection, feature, attribute),
- subject,
- medium (e.g. text, graphic, list, simulation),
- level of abstraction (e.g. associative, qualitative, quantitativ) and
- level of detail (e.g. conceptual, configurational, detail).

In order to build very large knowledge bases (VLKB) for intelligent CAD (ICAD) in /9/ the necessity for a knowledge standard is emphasised. There is a given classification of knowledge in two dimensions:

- recognised - unrecognised knowledge and
- codified - tacit knowledge.

Tacit knowledge could be explicitly or implicitly recognised by human beings and used for reasoning but very difficult to describe (e.g. the so called commonsense). Codified knowledge is always recognised and described with symbols, figures and so on (e.g. textbook knowledge, information stored in a database). Expertise and skill are mostly composed of unrecognised and tacit knowledge. Unrecognised and codified knowledge is meaningless. The primary goal of systematisation of knowledge should be to convert recognised and tacit knowledge to recognised and codified knowledge to make it computable and improve its reusing and sharing. Besides two types of design knowledge are identified:

- design process knowledge (which describes how) and
- design object knowledge (which is largely fact knowledge to discribe what).

A design process begins with ambiguous or rough descriptions of the design object and they will be gradually detailed and completed.

Stemming from the field of feature modelling a matrix representation for product description is known /10/. So a feature should be defined in the scope of a specific view onto the product description with respect to:

- classes of product properties (a class of properties is a combination of properties, which can be logical structured) and
- to phases of the product life cycle (product life cycle is a model to distinguish between significant phases of product creation, usage and replacement).

product property classes \ product life cycle	marketing	product planning	conceptual design	embodiment design	detail design	production planning	production	...
requirements		2, 5, 6	1, 2, 5, 6	1, 2, 5, 6	1, 5, 6	5	5	
functions			2	2				
solution priciples			2	2				
geometry		6		1, 3, 6	1, 3, 6			
tolerances				1, 3	1, 3			
materials		6		3, 6	3, 6			
manufacturing methods and parameters		6		4, 6	4, 6	4	4	
strength and durability				3	3			
costs				1, 3, 4	1, 3, 4	4	4	
time				4	4	4	4	
usage properties		2	2	2, 3	2, 3	2		
environmental properties				3	3			

1 - PICASSO, **2** - EQUIP, **3** - PLUS, **4** - AMANIS, **5** - demanda, **6** - sheet metal design

fig. 5: Description of handled knowledge with respect to information content and usage during life cycle

A view describes the way to look at the product and its properties during the product life cycle. In some disciplines there are similar, but not quite identical frameworks for product life cycle definition. Anyway, in practise each product has to follow its own life cycle.

The selection of exertions above shows that it is not very easy to systematise knowledge. It seems to be almost impossible to structure knowledge in a uniform manner. The method of systematisation strongly depends on the view of the problem. Nevertheless several similarities are recognisable. More or less there is a subdivision of knowledge concerning:

- **objects**
  e.g. epistemic structure, domain, design object, product properties and
- **processes**
  e.g. heuristic structure, task, design process, product life cycle.

In compliance with the various approaches it seems to us that the matrix representation considering product properties and life cycle is quite suitable to represent engineering knowledge handled in the systems as described in chapter 2 (**fig. 5**). In this way we could describe mainly the content of information concerning the product and the usage of system functionality with regard to each stage of design/developement. The matrix indicates clearly the main field of application. It seems that the major need for support lies in the embodiment and the detailed design phase. The heavy loaded vertical axis of the product property class requirements shows the importance of an integrated requirement management.

Another aspect of knowledge systematisation may be the design science /11, 12/. From this we tried to derive a kind of 'sophistication level' of possible design support. Certainly this level matches with different representation forms of knowledge. Concerning the complexity of engineering knowledge it may be devided into:

- structured data description (e.g. verbal or graphic information, data bases)
- rules (e.g. if -- then)
- functions (e.g. formulae, constraints)
- analysis methods (e.g. calculation, simulation, neural nets, reasoning mechanisms)
- single processes (e.g. design subactivities) and
- connected processes (e.g. methodical approach, computable design process model).

Simple knowledge is represented as plain structured data whereas complexity grows with relationships such as rules or formulae up to interactively working (system) processes triggered by design activities. Main points are structured data descriptions, rules and functions (**fig. 6a**). All of the systems are able to handle such information. Analysis methods are slightly underrepresent because there are a lot of powerful applications on engineering problems (e.g. FEM systems, calculation programs, simulation tools) that are not part of the systems itself. There seems, however to be a lack of applications concerning process oriented information.

a) complexity of handled knowledge

1, 2, 3, 4, 5, 6	1, 2, 3, 4, 5, 6	1, 2, 3, 4, 5, 6	2, 4	(2)	(2)
structured data description	rules	functions	analysis methods	single processes	connected processes

b) temporal durability of handled knowledge

2	1, 3, 5, 6		4	
1 durable (static)	2	3	4	5 unsettled (dynamic)

c) kind of adaptation of knowledge (feedback strategies)

1, 2, 3, 6	5			4
none/manual	quasi feedback	feedback with manual adaptation	feedback with automatic-interactive adaptation	feedback with automatic-autonomous adaptation

1 - PICASSO, **2** - EQUIP, **3** - PLUS, **4** - AMANIS, **5** - demanda, **6** - sheet metal design

fig. 6: Description of handled knowledge

One more representation supplement the reflection. In our opinion temporal behaviour of knowledge is also an important aspect on knowledge characterisation. The durability of knowledge in terms of time means the amount of changes that take place on the data the system is fed with. Durable means that once the data is inserted, it will not change anymore (i.e. lifespan of several years, e.g. bearing calculation method, general guidelines for embodiment design) and can be used unreflected (premising that it is correct) in this state. Unsettled means that the data underlies permanent change (i.e. lifespan of several days, e.g. capacity utilisation, occupation of machines, stock information) and that the use of it depends highly on the time. We used a scale from one to five to roughly characterise the data handled in the systems of each project (**fig. 6b**). The spearhead unambiguously lies on durable information. Certainly one had to ask how useful is a design support with unsettled information. But nowadays due to concurrent engineering approaches and a production in a turbulent environment unsettled information become more and more of interest for a support of development processes.

The last two representations show that most of the projects handle rather static data and stick to a more simple representation of knowledge. This is naturally due to the enormous degree of complexity one has to handle with more sophisticated systems. However, this analysis shows in which direction further research should head, i.e. systems which can handle more time critical data (as a major requirement form the industry) and with it naturally more complex information structures and support activities.

3.2 Knowledge feedback strategies to design

From our experience we have observed that design support relies mainly on feedback information, i.e. information from a later life cycle phase of a product back to the early stages of a new one. The different systems showed various aspects of the feedback which will be elaborated in this chapter.

In the beginning, when a product is developed from scratch, ideally no experience at all exists for this product. The designer will enter each phase of the development trying to make optimal decisions for the future. Only when a phase is finished or the product is already in use and when he observes possible drawbacks in the handling of his product, he may be able to connect those drawbacks (the symptom) to decisions in some phases (the cause) he had made. This process is purely manual based on designers own experiences since the system itself does not know or recognise anything of these interrelationships, i.e. none feedback.

Having products with a more or less complete product description, the designer is able to extract such interrelationships manually and to use them when he is going to make improvements to his products or to create new products, i.e. using the experience and knowledge of already concluded designs. Still, this is a manual process and that is the reason we call it quasi feedback. This means, that there is no intelligent retrieval to detect such interrelationships.

Therefore the nearest goal is to build up such retrieval mechanisms. So the designer could search for situations similarto his current stage (e.g. products with comparable properties, information on a specific life cycle phase). He can look after related results, i.e. possible solution for his problem. That's why we call it feedback with manual adaptation.

Moreover a next step may be to dedect possible interrelationships between decisions and related results by the system itself, i.e. a more sophisticated support to deduct support actions from arising design contexts. Relationships between experience and decisions taken before shall be extracted automatically. The designer there will have the opportunity to select either one or more of the proposed support actions to discover the most suitable one. It might also be that he will go another way in problem solving. We call this feedback with automatic-interactive adaptation.

Still using this mechanism, but in a strictly automated form, may be another strategy. This means preventing the designer from making decisions. The most suitable solution/action will be detected and initiated by the system. Here we speak of feedback with automatic-autonomous adaptation. This may be a disputable strategy. It might however be useful, e.g. in a strongly bounded (high-complex) domain the designer has neither experiences nor reliable information. In our opinion this should be the exception.

Summarising the analysed projects from the viewpoint of feedback it is clear that most of them have only manual or quasi feedback (**fig. 6c**). In practise this is to feed a data model with the more or less formalised experience and to make use of it during the design phases. However one system strive for more and we think that this will be the direction to go for in the future.

4 Conclusions

This paper has introduced a selection of design support relevant project run at the IMW. From that, an analysis of knowledge engineering within these projects has been made. Although this examination is largely an internal one, we believe that the applied methods and the gained abstraction hold for more than these projects. In this sense, we like to summarise as follows (statements to think about when building a design support system):

Relieving a designer in the way of taking ever repeating actions away from him means to model what in his mind is represented in an epistemic structure. This is to model certain data structures which will be matches with arising cases during the design and, in cases of fits, pre described actions could take place. A more sophisticated support is to deduct support actions from arising design contexts. Here, a metamodel is needed and the system has to cope with the heuristic structure of the user. This view on knowledge representation indicates the level of modelling and the complexity of application a system designer has to think about.

As already pointed out, several feedback strategies have been extracted. It is evident that future research has to concentrate on those systems which model information beyond simple data structures. Of interest are complex relationships which allow precise reductions of the solution space to be made (refer to /13/, tactics during the search for solutions - balanced search).

A design support system has to accept decisions taken by a designer (and might be allowed to propose solutions as well).

It is state of the art to document what decision has been made. Not that often, but still common is to document, how something was decided. This is derived from the decision itself and its context. It seems more important to us to document not only what (design object) and how (design process) but also why something has been decided. Still, design is a creative process and some decisions might be arbitrary. It is important to document them because they are the ones to be reconsidered first in cases of unsatisfaction. Documenting why something has been decided is also important for possible feedback strategies to make design processes comprehensible.

The analysis seems to show a weighting towards of research for support systems concerning the embodiment and detailed design phase. This is also reflected in the overwhelming market for design tools which exactly concentrate on those phases (CAD systems and supplementing modules). On the other hand, two important gaps have shown up. One is the missing support of the planning and conceptual phase. The other is the missing extraction of information from the later phases. We see the early phases still as a information target while the later phases are a source of them. Further research should head into the direction of making use of as much information as possible to support decisions in the very early phases of a design.

The information sources used for current support systems seems to be rather static and research concentrates on these. Actually, more unsettled data is more important, because this data is the critical type in terms of actuality, time, costs and so forth. This means, that future support systems should be nearer to such data.

Moreover, a 'new' design support system shall behave in the same manner as the 'previous' one and be used in the same way – otherwise it will not be accepted. Generally, this means that design support systems should help the designer in the way he is working right now, otherwise it will be rejected. In contradiction to that, innovation often requires new thinking and sometimes this new thinking (or methodology) affects the users. Although this paper is not primarily concerned with this topic, we would like to point out, that a smooth and sound user interface (following results of industrial sciences and ergonomic studies) is a strict requirement. Otherwise all the finest and best methodological approaches of knowledge engineering are useless because their benefits do not reach the user.

5 Literature

/1/ PDTAG-AM (Hrsg.): PDT Days '97 European Conference on Product Data Technology. Berkshire: Quality Marketing Services, 1997

/2/ P. Dietz und A. Ort: Verwendung von Wiederholteil- und Normteilkatalogen gemäß der ISO 13584 "Parts Library" unter besonderer Berücksichtigung der Anforderungen in der Konstruktion. VDI Berichte Nr. 1289, Düsseldorf: VDI Verlag, 1996, S. 399-408

/3/ P. Dietz, S. Penschke und A. Ort: Ansätze zur parallelen Gestaltung von Produkten und Fertigungsprozessen. VDI Berichte Nr. 1322, Düsseldorf: VDI Verlag, 1997, S. 313-329

/4/ D. Dörner: Problemlösen als Informationsverarbeitung. 3. Aufl. Stuttgart: Verlag W. Kohlhammer, 1987

/5/ N. J. Nilsson: Principles of Artificial Intelligence. Berlin: Springer Verlag, 1982

/6/ D. T. Pham (Hrsg.): Artificial Intelligence in Design. Berlin: Springer Verlag, 1991

/7/ T. Lenau und L. Alting: Intelligent Support Systems for Product Design. Annals of the CIRP Bd. 38 (1989) Nr. 1, S. 153-156

/8/ V. Baya und L. J. Leifer: A Study of the Information Handling Behavior of Designers during Conceptual Design. DE-Vol. 68, New York: American Society of Mechanical Engineers, 1994, S. 153-160

/9/ H. Yoshikawa: Systematization of Design Knowledge. Annals of the CIRP Bd. 42 (1993) Nr. 1, S. 131-135

/10/ C. Weber: What is a Feature and what is its use? - Results of FEMEX Working Group I. (ISATA '96, Florence 3.-6. Juni 1996) 1996, S. 109-116

/11/ G. Pahl und W. Beitz: Konstruktionslehre - Methoden und Anwendung. 3., neubearb. u. erw. Aufl. Berlin: Springer Verlag, 1993

/12/ V. Hubka und W. E. Eder: Design Science - Introduction to the Needs, Scope and Organization of Engineering Design Knowledge. Berlin: Springer Verlag, 1996

/13/ G. Fricke: Successful Individual Approaches in Engineering Design. Research in Engineering Design Bd. 8 (1996) Nr. 3, S. 151-165