Go to contents04

Case-based Reasoning in Collaborative Design:

The role of product models and information structures

POPOVA, Mina 1; JOHANSSON, Peter 2; LINDGREN, Hans 1
Dept. of Visualization and Modelling1 , Dept. of Structural Engineering2
Chalmers University of Technology, 412 96 Gothenburg, Sweden
http://www.arch.chalmers.se/projekt/cbd/

This paper discusses methods for information management through the rational application of IT within collaborative design. We explore the possible integration of the platforms of case-based reasoning and information structures. We examine the potential combination of existing techniques (CAD-tools, word processors, general applications, WWW) and standards (IFC, national classification systems) into a system for information management. We focus on the designers' use of heterogeneous information and the further development of a prototype based on product-model and process-model technology. Today, XML helps us structure various kinds of information before the system performs case-based reasoning sessions. The aim is to promote efficient and flexible information management in a case-based design process. Through the use of standardized product models, this information will be sharable and suitable for reuse and feedback. The more often the information is reused, the more general and adaptable it becomes i.e. it evolves. This scenario requires, though, efficient information management in the design office: a quality system for evaluating the information for reuse, consequent use of standardized product models and IT.

Keywords: *Case-based reasoning, product models, information structures, collaborative design and construction process.*

Background

During the last two decades, the construction industry has become increasingly aware of the need of information sharing during design, construction, and management. The segmented construction process causes insufficient feedback from previous projects resulting in loss of knowledge (Kalay 1997 and Lindgren 1992). In the 1980-ies, IT was mostly associated with the automation of specific tasks and new techniques (Fruchter and Clayton 1993); later on, communication, case-based design (CBD) systems, and shared product and information models were brought to light. In the 1990-ies, the development lead to efforts like ISO 12006-2 (by the International Standardization Organisation) and Industry Foundation Classes (IFC) initiated by the International Alliance for Interoperability. Architects are the prime users of visualisation models but occasionally HVAC and structural engineers do build models for simulating different functions of the building. Nevertheless, this methodology is rarely used, probably due to the cost and the problems with defining the standards for information exchange.

Today, we should ask ourselves what was done well and what ought to be changed in our projects. The answers are conceptualised during a pre-project stage of the design and construction process. They are derived analysing and matching the client's needs

Go to contents04

with the experience of the designers. By integrating these issues in the design process, we can add value to the client through the use of information systems supporting the reuse of design information.

The research problem

Our research focuses on the rational application of IT in collaborative design and, especially, on methods to integrate the design efforts of architects and structural engineers. At this development stage of our prototype, we study the integration of the platforms of information structuring and CBD (Johansson 2000). Here, the product models are used to structure design information and together with the CBD-methodology to search, retrieve, reuse, and evolve the information. Probably, the most significant feature of CBD is motivating the designers to save structured, reusable information. The next step would be to achieve a higher degree of integration between CBD-systems and information models and structures. The general concepts are based on our previous participation in the cross-disciplinary effort Project Wide Databases at Chalmers (Johansson 1996, Popova 1997).

Case-based reasoning and case-based design

Case-based reasoning (CBR) originates from the cognitive observation that humans often rely on past experience to solve new problems. Schank (1982) created a model describing how case-specific information can be stored in a memory and retrieved when needed. The same knowledge structure is used in remembering, understanding, experiencing, and learning, and it changes as a result of its experience. This model has evolved focusing on indexing, storing cases, and adaptation to new situations thus becoming CBR (Kolodner 1993). Because of the importance of experience in design, many CBD-systems have been implemented for problem solving (Maher et al. 1997). The results have proved that case-specific information and CBR are usable a design system (Hartvig 1999).

Information structures concerning CBD

Today, we have standard product models like ISO 12006-2 and IFC, and national classification systems such as the BSAB-system in Sweden (BSAB-96 1999). The ISO standard is a frame for information exchange e.g. in CAD, specifications, and cost estimates (ISO 1997). Since it does not contain classification tables, the use of regional and national ones is recommended. The IFC model is a framework of classes for information exchange between computer systems, aiming at software interoperability in the AEC/FM domain.

Design rationale (intent) is the rationale behind decision-making and the information about the design evolution (de la Garza and Oralkan 1995). A conceptual framework for the AEC industry is needed in order to represent this knowledge since it facilitates CBD (Simoff and Maher 1998). At present, the standard product models: ISO 12006-2 and IFC cannot serve as a conceptual framework since they still do not support all the stages in the design process (Ekholm and Tarandi 2000). We believe that the overall goal of the product models should be to assist the activities of the design team (Popova 1997, and Turk1998), to allow the creation of new types of objects (Fisher 1994), and the evolution of their information structure.

Structured, weakly structured, and raw data

The acquisition of information is a common problem when using integrated information structures and CBD-systems. The information used by designers is divided into three categories (Simoff and Maher 1998):

1. Structured data, e.g. information covered by the standardized product models and created by applications promoting their use, attribute-value pairs, relational tables, object-oriented structures;
2. Weakly structured data, e.g. information not covered by the product models, texts, tables, calculation documents;

Go to contents04

3. Raw data, e.g. raster and animated images, sketches, audio and video data.

Design calculation documents contain information unavailable elsewhere: concepts (e.g. frame), solutions, and, to some extent, the design rationale in the process. Architects often store information as raw data; IFC deal with structured data, while CBD-systems need information from all the groups. Besides information exchange, IFC enables the information acquisition into a case-base where CBR-sessions can be conducted. Consequently, product models should be used for representing structured data; for weakly structured data, the task is more complicated.

ARCADE

The prototype investigates how to automate the process of capturing, retrieval, reuse, and evolution of weakly structured data in design calculation documents. Also, we have focused on how the process information in these documents could be used in CBD. A document is subdivided into sections by headings on different hierarchical levels. It can contain a list of variable definitions with a name, a physical unit, and a value as well as pictures and comments. ARCADE uses the format of Mathcad 6.0 and acquires cases by using variable definitions, dependency structure, and headings.

Here, the case-base contains calculation documents from six projects done by structural designers in Gothenburg. These documents contain information about the main geometry and the functions of the building (a warehouse) as well as the load calculations. The engineer starts designing the foundation by describing the known and the unknown variables. ARCADE performs a retrieval session by matching relevant old cases where the heading describes the class (concept) of a section (Fig 1).

The five most similar cases are retrieved. After the matching, two main questions must be answered:

1. Does the old case contain the information we need?
2. Does the document contain the variables that the old case needs defined?

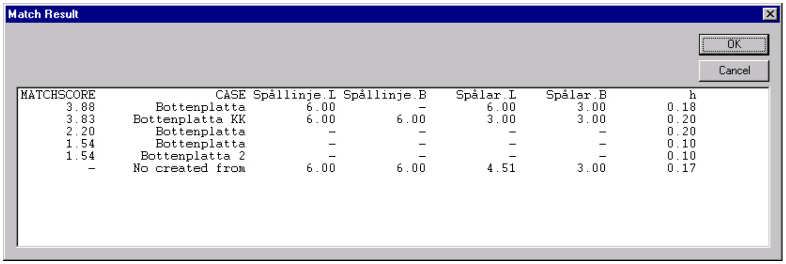
The first question is answered by a goal state match; the second - by a footprint match. ARCADE handles these through the dependency structure at acquisition time:

Foundation Slab (Bottenplatta): q_{NLk} (Live load) and S_L (Centre distance of the columns at length); Foundation Slab KK (Bottenplatta KK): q_{NLk},S_L and S_B (Centre distance of the columns at width).

If the footprint variables from a new case are found in the present design document, the similarity is calculated as that of a member variable. We achieve a better retrieval of information about the design process by using these variables. The value of q_{NLk} in the case Foundation Slab gets a higher match score than in Foundation Slab KK since it corresponds better to the variable value in the current case. If the structural designer now chooses to reuse the retrieved values, it will be registered by the system as a measure of the case's generality.

XML as a common format for CBD

Usually, a CBD-system needs information from several applications for CBR-sessions, which requires



MATCHSCORE	CASE	Spållinje.L	Spållinje.B	Spålar.L	Spålar.B	h
3.88	Bottenplatta	6.00	–	6.00	3.00	0.18
3.83	Bottenplatta KK	6.00	6.00	3.00	3.00	0.20
2.20	Bottenplatta	–	–	–	–	0.20
1.54	Bottenplatta	–	–	–	–	0.10
1.54	Bottenplatta 2	–	–	–	–	0.10
–	No created from	6.00	6.00	4.51	3.00	0.17

Figure 1. The match results.

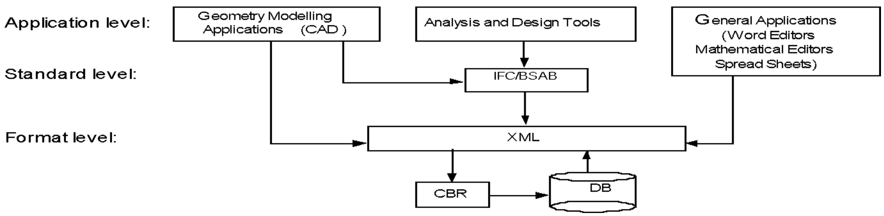
an independent format. Although most CBD-systems can use the ASCII- and the HTML-format there are limitations: values of calculated variables are not stored in the ASCII-format while the equations created by Mathcad are stored as pictures in the HTML-format. The Extensible Markup Language (XML) can solve the problem <www.w3c.org/XML/>. We use the format in our work because it is neutral, flexible, and lets us define our own tags to tag both structured and weakly data. The Mathematical Markup Language (MathML) is an XML-application for expressing mathematical notations <www.w3c.org/TR/MathML2>. Today, a query language is being included in the XML-standard that will most probably replace the ASCII- and the HTML-format.

Today, the work with mapping the EXPRESS-schemas of the IFC product models to XML progresses (Liebich 2000). The project ‘Application of IFC in Sweden ‘ proves that classification codes can be attached to a CAD-model and the information can be translated into the IFC Part21- and XML-format (Ekholm and Tarandi 2000). The BSAB Demonstrator shows that national classification tables can be structured with XML-tags and texts written in MS Word can be exported as XML-files (Häggström et al. 2000). Once in this format, the information in classification systems and product models can be used together in CBR-sessions. Figure 2 illustrates our concept of the information structure using XML as a common platform.

Structure of the Kit of Design Parts

Our work focuses on developing applications based on available technology: WWW, XML, CAD etc. and existing standards: IFC and BSAB. The outlined background shows a new way of understanding and dealing with structured, weakly structured, and raw data. It is possible to build standardised and yet user-friendly design information systems. According to our experience, even persons with limited computer skills can learn the basics and understand the prototype. The prototype Kit of Design Parts is at present organised as a file structure letting the user navigate through the archive (XML-files and images) in search of projects, design objects, spaces, and previews. The information is stored, edited, and searched through HTML-forms. The system allows various categories of information: references to classification systems (BSAB 96), standards (IFC/BSAB), and information defined by the designers: project title, key words, designer’s notes, ID number, etc. We can store and retrieve general project information: text documents, project briefs, multimedia data as well as specific i.e. design objects: windows, doors, or spaces, often of particular interest to architects during design. A database of about 100 cases is considered sufficient for performing CBR-sessions, which makes the method useful even to small design teams (Schmitt et al. 1997). It is important that the information is stored easily in the database otherwise the designers would find the method time-consuming. Other issues are the routines and the user rights i.e. the consequent use of standards (IFC/BSAB), the systematic storing of design information, editing should be supervised by the project manager; each user should be provided an “own” space etc (fig 3).

Figure 2. Representation of the information structure



Go to contents04

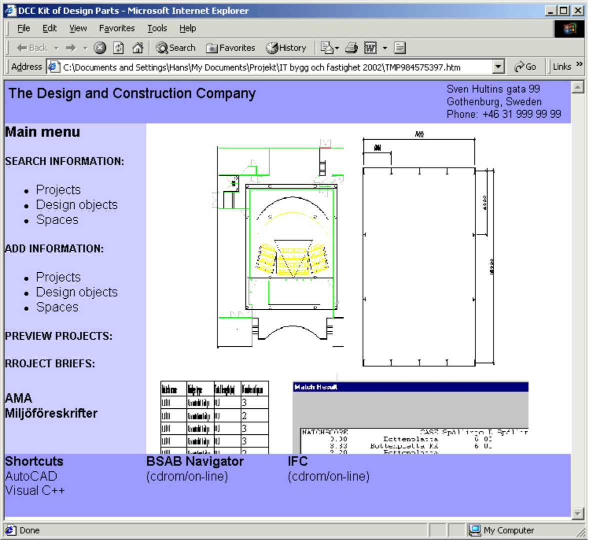


Figure 3. The interface of the Kit of design parts.

Our next task is to investigate how changes/ improvements in the design can be easily captured thus updating the XML source documents and offering the designer evolved information. As the information is adapted to new projects it becomes not only more general but also more reusable. In short, we will be able to perform CBR-sessions by using weakly structured data produced by architects. The final step in our project will be to combine the two prototypes - ARCADE and Kit of design parts into one CBD-system accessible through a common interface (Fig.3). Since both professional will use the same product models to structure design information and CBR-sessions to manage it, we can expect a better collaboration and a higher efficiency and quality through testing different solutions.

Conclusions

Our work so far has shown that the concepts of product models and CBR can support efficient information management in collaborative design. CBD-systems cannot replace the designer's expertise but they can manage great information quantities and provide a basis for problem solving. ARCADE proves that weakly structured information can be acquired automatically during design and then used for CBR-sessions. Designers should extend their use of integrated product models to produce and share reusable information. The systematic use of models and CBD-systems will promote the evolution of the design process, the storage and retrieval of information becoming a natural part. In short, the implementing of new work methods and routines is indispensable if the transition from a sequential building process to an integrated, collaborative one is to succeed.

References

BSAB 96 System och tillämpningar, (SR-Rekommendationer 10)(1999). AB Svensk Byggtjänst, Stockholm.

de la Garza J. M. and Oralkan G. A. (1995). Using design Intent for Interpreting Brand-Name-or Equal Specification. Journal of Computing in Civil Engineering, Vol. 9, No. 1, pp. 43-56.

Ekholm A. and Tarandi V. (2000). Application of IFC in Sweden - phase 2. Final report. The Swedish Building Centre Systematics, Stockholm.

Fischer G. (1994). Domain-Oriented Design Environments. In “Automated Software Engineering”, Kluwer Academic Publishers, Boston, MA, pp. 177-203.

Fruchter R., Clayton M., et al. (1993) Interdisciplinary Communication Medium for Collaborative Design. The Third International Conference AI CIVIL-COMP93. Edinburgh.

Holgate A. (1986) The art in structural design. Oxford University Press.

Häggström L., Johansson B., Rosengren P. (2000). BSAB Demonstrator - etapp 2. Final report. The Swedish Building Centre Systematics, Stockholm.

Hartvig Susanne C. (1999) A Framework for IT-based Design Tools - Enabling Integration and Design Support. Report 6 Department of planning, Technical University of Denmark.

ISO (1997). ISO/DIS 12006-2 Organisation of information about construction works - Part 2: framework for classification of information. Geneva: International Standardization Organisation.

Johansson P. (1996) Case-Based Structural Design - reusing design calculation document information. Chalmers University of Technology, Division of Steel and Timber Structures, Publ. S 96:3, Göteborg.

Johansson P. (2000) Case-Based Structural Design - using weakly structured product and process information. Chalmers University of Technology, Division of Steel and Timber Structures, Publ. S 00:7, Göteborg.

Kalay Y. E. (1997) P3: An Integrated Environment to Support Design Collaboration. In Representation and Design, ACADIA, Cincinnati, Ohio.

Kolodner J. (1993) Case-Based Reasoning. Morgan Kaufmann, San Mateo.

Liebich T. (2000) XML schema language binding of EXPRESS for ifcXML, MSG-01-001(Rev 3), International Alliance for Interoperability.

Lindgren H. (1992) A Framework for Computer-aided Reuse of Architectural Design Solutions. CIB W78 workshop on Computer-Integrated Construction, 12-14th of May, Montreal, Canada.

Maher Mary Lou, Pu Pearl (1997) Introduction to the Issues and Applications of Case-Based Reasoning in Design. In Issues and Applications of Case-Based Reasoning in Design, Maher and Pu (editors), Lawrence Erlbaum Associates, Mahwah, NJ, ISBN 0-8058-2312-3. pp. 1-10.

MathSoft (1995) Mathcad User’s Guide Mathcad 6.0. Cambridge, MA, MathSoft, Inc.

Popova M. (1997) IT and the Architect’s Role. Division of Design Computing, Chalmers University of Technology, CTH-A CADLAB 1997:1, Gothenburg.

Schank R. (1982) Dymanic Memory: A theory of learning in computers and people, Cambridge University Press.

Simoff S. and Maher M.L. (1998) Ontology-based multimedia data mining for design information retrieval, Proceedings of ACSE Computing Congress, Cambridge.

Schmitt Gerhard, Bharat Dave, Shen-guan Shin (1997) Case-Based Architectural Design. In Issues and Applications of Case-Based Reasoning in Design, Maher and Pu (editors), Lawrence Erlbaum Associates, Mahwah, NJ, ISBN 0-8058-2312-3. pp. 241-260.

Turk Ziga (1998) On Theoretical Background of CAD. Lecture Notes in Artificial Intelligence, 1454, Springer, ISBN 3-540-64806-2, pp. 490-496.