Conceptual Models for Communicating Knowledge in the Building Industry — Implementation of the Cube System

PER CHRISTIANSSON[1]
JÖRGEN MODIN[2]

ABSTRACT

This paper presents the conceptual models that emerged during the development of the Cube demonstrator in the Cube project at Lund University in Sweden. The Cube project was led by the Department of Construction Management with the systems development being done by the authors, at the KBS-Media Lab of the Department of Structural Engineering. The paper presents the underlying conceptual models for knowledge transfer used in the Cube system. In the Cube system the formalized rules do not apply to the knowledge itself but rather to the way it is conveyed. The goal for these models is to assist a growth of building knowledge that is driven from the building site and the daily situations there. It is showed how knowledge chunks are labeled and stored in answer boxes specific for each project or building site. Existing classification systems, BSAB, are implemented together with a dynamic limited vocabulary and task-oriented headings to form an efficient knowledge communication and retrieval system.

Key Words
knowledge capture; taxonomies; multimedia; knowledge transfer; database

WHAT IS THE CUBE SYSTEM?

The Cube system is a protype that captures and stores knowledge from the building process (see Figure 1), and delivers it to the right person at the right time. The current contents of the Cube database deals with the construction stage of the building process. Knowledge is stored as pairs of questions and answers. Each answer is divided into three sections: Direct answer, answer with alternatives and answer with references.

The Cube system is designed to be an aid in the process of generating and maintaining knowledge relevant to the building process, and to make this knowledge easily accessible to the participants of this process.

[1] *Associate Professor, PhD KBS-Media Lab, Lund University, Box 118, 22100 Lund, Sweden.*
[2] *Researcher, KBS-Media Lab.*

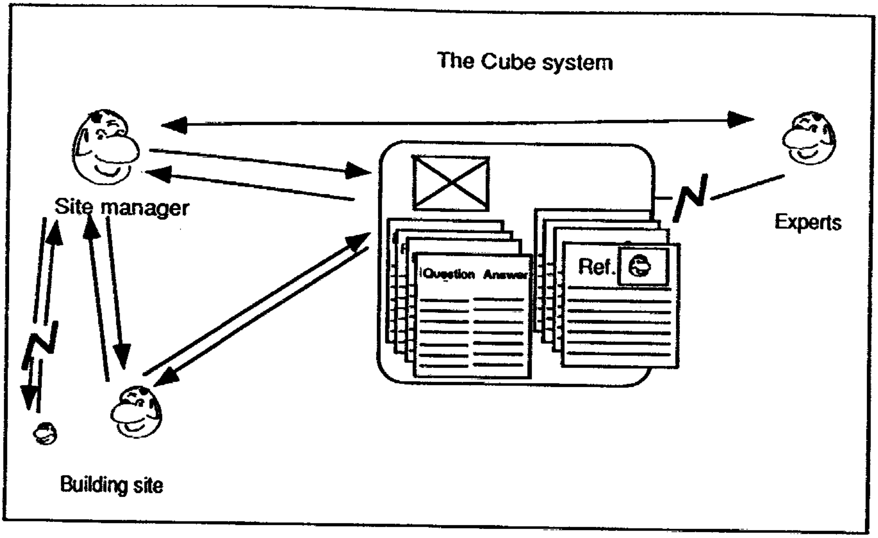

Figure 1. The role of the Cube system in a Dynamic Knowledge Net, in which the participants interact and receive support via phone, personal contact and the computerized support of the Cube system

OBJECTIVES OF THE Cube SYSTEM

The Cube system was developed to enhance the possibilities to collect and make available knowledge emerging during the construction phase in the life of a building. The demonstrator which was developed is used to:

- develop and evaluate interface design
- develop adapted knowledge representations and knowledge handling processes
- make visible technical solutions for communication
- in itself serve as a communications tool during the development work
- contribute to the conceptual development within the application area.

Further the demonstrator should have these inherent characteristics:

- it should stimulate persons to look for knowledge within the system
- it should be open to store non-company information
- it should stimulate person-to-person cooperation and communication.

DYNAMIC KNOWLEDGE NETS

The Cube system, and other systems born at the KBS-Media Lab, are looked upon as being parts of the Dynamic Knowledge Net. The Dynamic Knowledge Net is a network to connect persons, organizations/processes, and computer tools. Connections in the Dynamic Knowledge Net, DKN, may be established in time and space (Christiansson, 1992).

When we communicate as human beeings we use languages which formalize the information flow (English, symbols, sounds etc). Computerized knowledge also need to be communicated in a more a less formalized way. We could talk about a network of knowledge which dynamically adapts to different needs: a Dynamic Knowledge Net, DKN. In the DKN we connect a mix of knowledge representations to make it easily accessible and easy to augment.

Information is captured, stored, manipulated, transferred and delivered through more and more efficient media. This increased efficiency of media can be seen both in transfer speed, storage capacity and improved multimedia interface. The senders and receivers of this information may be man and/or machines. At the same time we store greater and greater amount of information in digital form in computers at a steadily increasing speed. The information when communicated carries knowledge and implicitly, or explicitly contains representations or models of our reality. Knowledge is mainly stored in people but may also be stored in machines and networks of machines. These networks or highways for information makes possible high accessibility to knowledge both in man and machines.

Very high demands are put on the DKN in terms of bandwidth, compression and decompression routines, parallel cooperating processes etc. For example; two persons operating in virtual reality on the same damaged power station together with a muscle and eye extender (robot) or a design team collaborating with the building site on a design alteration. We already face problems in distributed multimedia networks.

The access points of the net will in many cases be location independent. Today the mobile telephone system is rapidly extended (and there are fixed stations with personal access codes). This also implies that the network has to 'know' where you are, to be able to establish connection when you are searched for. At Xerox PARC, (Weiser, 1991), the researchers have 'everywhere'/ubiquitous access to tools like 'liveboards' (electronic black-boards) and smaller 'pads' and 'tads' as opposed to the personal computer.

CONCEPTUAL MODELLING

This paper deals with the conceptual models of the Cube system. A computer is a universal machine and can through programming host many different kinds of specialized machines. Conceptual modelling can be viewed as the process of designing such a "machine inside the machine". The computer can in this respect be likened to a box of mechanic pieces ('meccano') and the conceptual models we create can be likened by constructions of 'meccano': in the process of creating a computer program, we make decisions on where the joints should be, the length of the levers, etc.

A conceptual model may or may not be well suited to the application. It may turn out that the model cannot cope with change and cannot be developed further. To use the 'meccano' metaphor: the joints simply are not in the right place to allow motion. When the conceptual modeling is successful one has created an improvement that is irreversible, something that can be used as the new standard for performing the task at hand.

THE DEMONSTRATOR METHOD OF SYSTEMS DEVELOPMENT

The Cube system was developed using the demonstrator method. The demonstrator method of systems development is a form of prototyping. A rough sketch/prototype of the system is the starting point for communicating opinions and insights on what the final system should look like and how it should behave. This sketch is then continually refined until it becomes a valid model of the final system. One could say that the demonstrator starts as an simulation of the final system and ends as the real thing (see Figure 2).

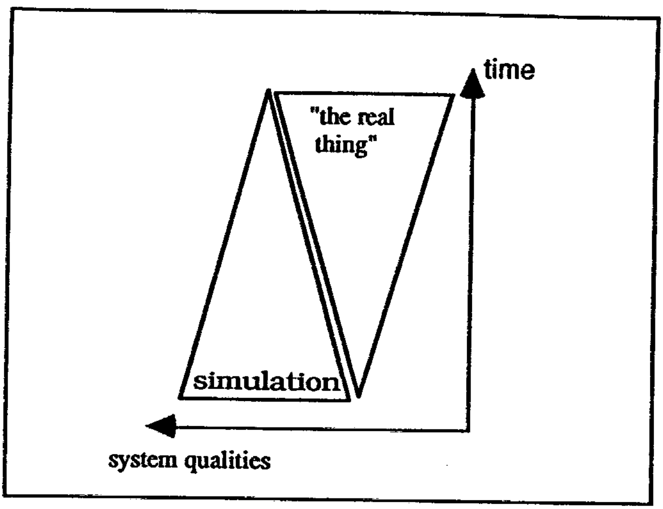

Figure 2. The Demonstrator is More and More Approaching the Final System

The Cube system was demonstrated to the production group that evaluated and criticized it during meetings. The demonstrator was refined and changed until the next meeting. Often changes in the "look" (the user interface) could be made during a meeting to give immediate feedback on proposed ideas. Occasionally the requests from the production group could not be met by the demonstrator. In these cases the demonstrator either had to be fundamentally redesigned or the requests of the production group had to be lowered or altered. The demonstrator method requires rapid protyping tools. In the Cube project we used HyperCard 2.1 and extensions to the Macintosh OS (Apple Computer, 1987).

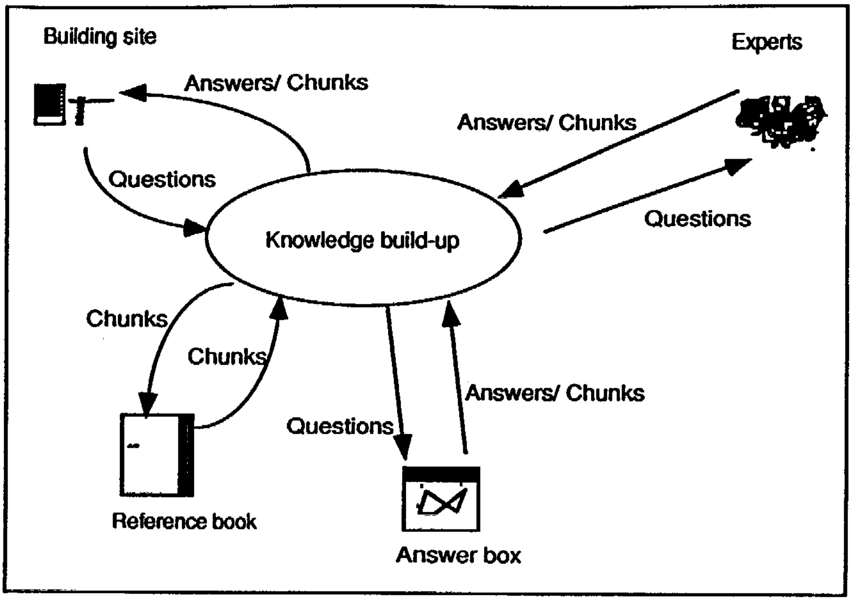

Figure 3. The Data Flow in the Cube System

CONCEPTUAL MODELS OF THE CUBE SYSTEM

Overall Structure and Behaviour

The Cube system contains facilities for capturing, storing and retrieving knowledge. The user enters questions into the cube system with the aid of mark-up systems. The answers to the questions are then entered into the cube system by an expert, or provided immediately by the Cube database itself. The Cube database consists of two parts: the answer box, which contains question-answer pairs, and the reference book that contains knowledge stored from previous projects. When a user enters a question the Cube system tries to display information from the databases that is relevant to the question. If the user decides there is no relevant answer the question is stored in the answer box and experts are notified of the new question. As the database grows a substantial amount of knowledge in the form of question-answer pairs will be stored in the answer box. The answer box is tied to the specific project. When the project is finished the answer box is stored for future reference, and knowledge of general interest is transferred to the reference book (see Figure 3).

This transfer becomes a process of refinement. The reference book is read-only and is accessible to all projects in the organization.

It was regarded as important that the users, when putting a question, should have as rich ways as possible of expressing themselves. The user should be able to take a shot of for example a building structure and include the resulting electronic photograph in a question. The system was therefore designed to be used with a (relatively) cheap electronic still video or video camera. A user-interface convention was established which indicates links to photos from words by drawing these words in bold italic text style (see Figure 4). A paint program was also integrated for making drawings or pasting/linking in drawings from other programs into a question. The drawings are called up by clicking at words drawn in bold non-italic.

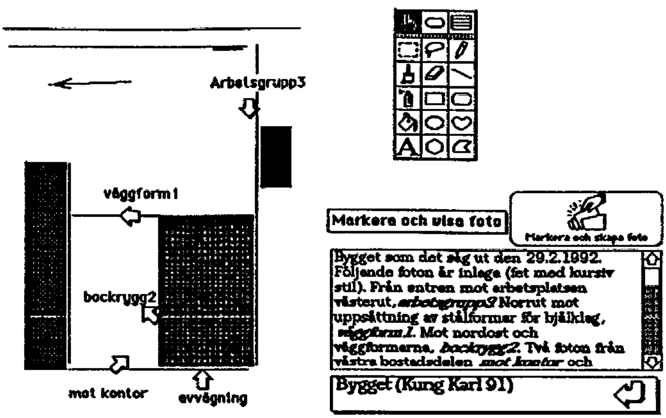

Figure 4. It's possible to integrate photos into sketches, as in this example that depicts a plan of the building site. The sketch is in its turn tied to a knowledge chunk. Clicking at the bold-faced, italic text brings up photos taken from angles indicated by the arrows in the drawing.

Modeless Information Seeking

The interface in the Cube system consists of graphical objects that respond to the user's input from mouse and keyboard (like most programs for Mac™ and Windows™). Since there are several objects the user can choose from at one time it is not possible to regard the entire program as a linear flow. The program flow is shaped in small pieces in the dialogue between the user and the program. It is therefore important to keep track of what information is available, *ie,* what state the program is in.

One can think of the screens as forms that are filled in by the user (see Figure 5), and as the user traverses the program, it gets into different states. When the user has filled in at least one search criterion he is able to perform a search. He is informed that a search is not possible at the moment, if there is no criterion.

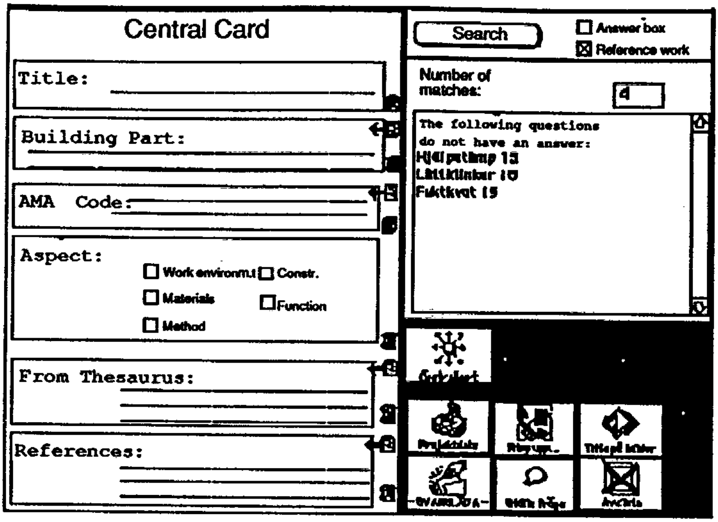

Figure 5. The user states what he is looking for by using the different mark-up systems available on the left half of the screen. The system returns a list of chunks that match the search criteria in the field to the right. You go to the chunk by clicking at the line with the chunk name (Some of the text in this picture has been translated to English for clarity).

States are not the same thing as modes. The user rarely has to select a certain mode in the Cube system. A task can be performed at any time as long as the system has sufficient information to fulfill it. This is what is commonly referred to as a non-modal interface. There are however for security reasons three global modes; user, expert, and systems maintenance mode. User mode is for the every-day user, expert mode is for subject-matter specialists providing the answers, and systems maintenance mode is for the developers of the system.

In the initial versions of the Cube demonstrator the user was welcomed by a screen that asked him whether he wanted to enter a question or search for information. It turned out that this did not make much sense to the user. From the view of the programmer it is a logical choice since the activity of entering text and the activity of searching the database are computationally different tasks. The user though does not know if the information he is looking for is in the database. Therefore stating a search query or putting a question is from his point of view the same activity, *ie,*to search for information. There is only a difference in response time.

A new model was designed around the concept *to search for information*, suggesting a non-modal interface where the process of searching information and entering a question becomes one seamless process. The same form with its interface elements is thus used to look for chunks and to enter new chunks. The user states what he is looking for by using the different mark-up systems available. The system returns a list of chunks that match the search criteria. If the user decides there are no good answers he puts a question and the system uses the entered criteria as the mark-up for the new question.

Marking Chunks

One of the questions the development team had to face from the early versions of the demonstrator was: "How should information retrieval be performed in the Cube system?". One way would be to have access to free text search where the user states the words he thinks will be useful for retrieving knowledge chunks. There might though be a discrepancy between the user's vocabulary and the one used in the relevant chunks. It was therefore decided that the knowledge chunks should be tagged, with labels that could only be filled with a limited set of terms that are familiar to the building personnel. The users should be able to retrieve information by pointing and clicking at appropriate terms.

A limited set of terms can be derived from the BSAB classification system that is commonly used in the Swedish building industry. It is a system consisting of two hierarchies (an additional third hierarchy is under development). A hierarchy is referred to as a table in the BSAB system. See also '*The BSAB system*' below. A flat vocabulary was also introduced that can be customized to fit the vocabulary of the specific organization.

Lastly, five "check marks" were introduced to indicate the overall aspects of the question: Work environment, materials, method, construction, and function. See also Figure 5. Compared to classification systems for libraries, one could say that the BSAB system corresponds to an enumerative hierarchical system as the UDC, the limited vocabulary to a flat thesaurus and the aspects of the building process to general facets of a colon classification system.

Conceptual Models for Communicating the BSAB System

The BSAB System

The Swedish building classification system (BSAB, 1972) has been extensively used during 20 years in Sweden. The BSAB system, which was a follow up to the CI/SfB classification system, is now under reconstruction.

In the Cube system we use two BSAB tables to mark up question-answer pairs:

- Product table 2; pointing at building parts and installations expressed mainly from a functional view point. Example; 3.1 - Walls in a Building. 'Building part' in Figure 5.

Product table 1; pointing at technical solutions and activity results. Example; X2.112 - Windows and window doors made of wood. 'AMA-kod' in Figure 4.

In addition the answer/question pairs may be marked up for domain:

- Work environment
- Materials
- Methods (related to Product table 1)
- Construction (related to Product table 1)
- Function (related to Product table 2).

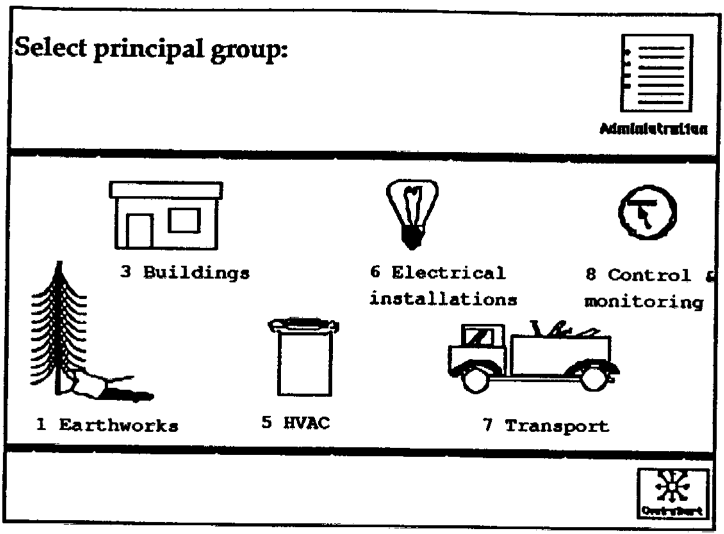

Figure 6. Main Menu for the BSAB System Pproduct Table 2 (Translated to English)

Graphical BSAB Browser

A graphical browser was introduced early in the project. Figure 6 shows the browser top level for the BSAB Product table 2 (building parts and installations). If the user chooses to mark up for 'Buildings' the next level of graphic browser will appear according to Figure 7.

Under '3. buildings' we can reach one more level for example 3 called 'loadbearing structure' (see Figure 7). If we want to mark up for walls we can either press 3.1 'walls' in the drawing or click in the list at the upper right. We have then selected 3.3.1 as a mark-up for our search.

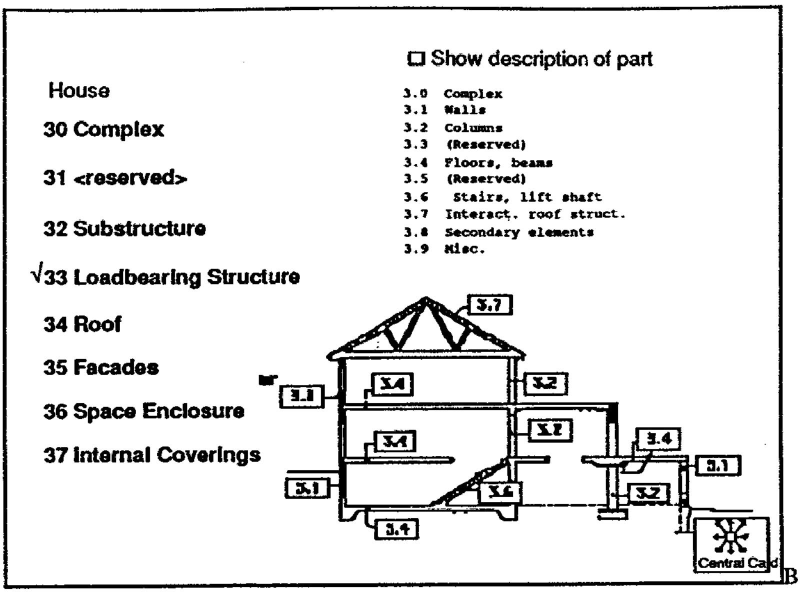

Figure 7. This screen is displayed when clicking at 'Buildings' in the screen depicted in figure.9 The user feels comfortable because drawings and accessible help text is copied from the familiar book (BSAB systemet, 1987).

Hierarchical Browser

Because it is a considerable effort to put the entire BSAB system into the form of the picture browser, a simpler interface was designed that can swallow any hierarchical system arranged in an ASCII file. Three different hierarchical user interfaces were considered:

1) An outline interface as used in text outlining programs as More™ (Fraase, 1987).

2) An interface consisting of a number of text fields arranged side-by-side with the highest hierarchical level to the left. This interface solution is used in the file browser of the Next™ computer and the class browser of the Smalltalk™ programming environment.
3) An interface with clickable objects that opens into windows, as in the Apple Macintosh™ Finder, *ie,* part of the operating system, see (Apple, 1987).

The second alternative was chosen, since it allows much information to be displayed in a small space and it utilizes the landscape (horizontal) orientation of the Apple 13" Monitor. The testers regarded the pictorial browser to be the more intuitive of the two BSAB browsers.

Information Refinement

What is the nature of knowledge? Here follow some viewpoints:

- Knowledge does not exist outside its context.
- Knowledge that lasts longer over time is of higher value.
- Knowledge is used to solve a problem. This may be a practical problem that involves a physical action or it may be a problem of *eg* maintaining a knowledge base. The latter is often referred to as *metaknowledge*. The border between knowledge and metaknowledge is vague and depends on the context.

In the process of increasing the amount of data in the Cube system it became apparent that there is a problem of low density of the knowledge in the system. The reasons for this happening are:

- Some of the knowledge is related to project-specific problems. This knowledge will be of low value outside the project.
- Some of the knowledge will become outdated with the emergence of new building standards, building materials and techniques, etc.
- Some of the questions are not stringent enough to form question-answer pairs of high information content. In this case the subject specialist must reply something as "What you're really asking is...", to get focus and clarity to the chunk.
- Due to the vast possibilities of human language approximately the same question might be posed several times without the subject specialist noticing it. This will lead to unnecessary duplication of information.
- Since there will be several independent databases that may be under growth at the same time redundancy occurs when the same question is put simultaneously in the databases.

Hence there is a need to concentrate the information through clustering similar chunks, sharpen the focus and clarity of the chunks and checking the knowledge-base for outdated or erroneous information.

In the Cube system the process of refining information is performed by one or several persons who transfer the knowledge chunks from the answer box to a reference book. In doing so, they select which chunks are to be transferred. They may also edit the mark-ups and merge several chunks into one. This gives that the refinement of information becomes a process based on judgment. The Cube system does not give advice on what information should be transferred. It just provides the tools for performing the refinement. The tools are gathered in the "transfer palette", see Figure 8.

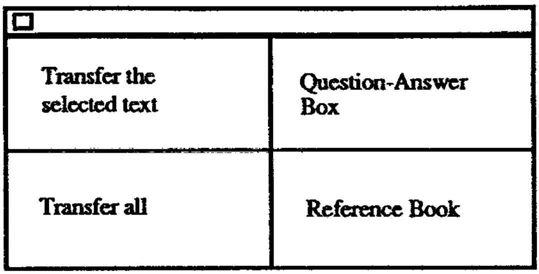

Figure 8. The Transfer Palette

Relevance Ranking

A simple relevance ranking mechanism was implemented along the lines of the ranking mechanism of the WAIS system (Kahle, 1991). A star rating precedes each displayed hit. The rating (number of stars) depends on the number of matches between the search query and the mark-up of the particular chunk.

CONCLUSION

A number of interlinked conceptual models were developed and used in the Cube project. These include:

- A division of the database into a *reference* part and a *project-specific* part.
- The default state of *information seeking* and the non-modal forms-oriented interface that supports it.
- *Mark-up systems* including a limited vocabulary, a hierarchical classification system and five aspects of the building process.
- *Graphical user-interfaces* to communicate hierarchical classification systems.
- The concept of *information refinement* comprising a model with a transfer palette to aid the transfer between the answer box and the reference book.
- *Relevance ranking* using star ratings.

It's too early to judge which of these conceptual models hold a lasting value. The initial reactions from building personnel to the entire system (the sum of the conceptual models) are very positive though.

ACKNOWLEDGEMENTS

The work was financed by the Swedish National IT-Building Programme, the Swedish Council for Building Research (BFR) and the Development Fund of the Swedish Construction Industry (SBUF).

References

Apple Computer Inc. (1987), *Human Interface Guidelines: The Apple Desktop Interface*, Addison-Wesley pp 144.

The BSAB system (1972), The Swedish building-coordination centre, pp 195.

BSAB-systemet, tabeller och tillämpningar (1987), AB Svensk Byggtjänst, pp 144 (in Swedish).

Christiansson, P (1992), Dynamic Knowledge Nets in a changing Building Process. *Automation in construction* Volume 1, Number 4, March 1993 Elsevier Publishers, pp 307-322.

Classification of information in the construction industry. ISO technical Committee 59 SC 13 WG2 (editors Karlsson H, Allott T), August 1992 Draft, pp 56.

Fraase, M (1989), *Macintosh Hypermedia, Volume I, Reference Guide*, pp 271.

Kahle, B (1991), *WAIStation User Guide, Prototype Version v 0.57.* via Internet.

Weiser, M (1991), The Computer for the 21st Century. *Scientific American. Special Issue on Communications, Computers and Networks.* September 1991. pp 66-75.

c. 1993, Management of Information Technology for Construction, K. Mathur et al (Eds), World Scientific Publishing Co., Singapore.