DESIGN AND MODELLING IN A COMPUTER INTEGRATED CONSTRUCTION PROCESS – THE BAS•CAAD PROJECT

ANDERS EKHOLM and SVERKER FRIDQVIST *CAAD, School of Architecture, Lund University, Sweden.*
E-mail: Anders.Ekholm@caad.lth.se and Sverker.Fridqvist@caad.lth.se

Abstract

A new approach to product modelling in a design context is proposed. CAD-software must not only enable product modelling, but must also support product design. This is not fully achieved in the traditional 'enumerative' approach to product modelling. We discuss how product design and modelling can be based on a 'facetted' approach to information modelling, and how a data model that supports the design process can be based on a framework for system information. The background for our research is the current development in the construction industry towards a computer integrated construction process. A first prerequisite for this is the use of computer based models. Another prerequisite is that CAD-software can support the design of the results of the construction process, including construction works, user organisations, and the production and facility management processes. A third prerequisite is that computer based models are built with standardised concepts and terminology to enable exchange of information between different actors and computer systems during different stages of the construction process. Principles for organising frameworks for user organisation and construction works information are presented in an appendix.

1 Introduction

1.1 DISPOSITION OF THE PAPER

The introduction of the paper discusses prerequisites for a computer integrated construction process; of specific interest in the design context is the shift from computer aided draughting to computer based modelling. The second part of the paper discusses principles for structuring product model information and presents principles for structuring frameworks for user organisation and construction works information. The third part of the paper presents some requirements on a data model for a product design tool.

1.2 PREREQUISITES FOR A COMPUTER-INTEGRATED CONSTRUCTION PROCESS

The current development towards a computer-integrated construction process is characterised of both an increasing use of computers and an integration of the different stages of the process (Björk 1995:12). The use of computers considerably increase the possibilities of handling the huge amounts of information that characterise the construction process. A computer-integrated construction process will enable a faster, safer and more complete transfer of information between actors and stages in the process. For example information from the brief development and the design proposal stages will more easily be made available and utilised during the production and the facility management stages.

The realisation of a computer-integrated construction process depends on a multitude of factors beside the development of computer technology. Many obstacles depend on lack of theoretical knowledge of both the building design process and the development of CAD-programs for design. Three main prerequisites for the realisation of a computer-integrated construction process are here considered as a background for the BAS•CAAD project.

A *first* prerequisite for a computer integrated construction process is the use of computer based models, so called product models, of buildings and other objects of design. Computer based models enable simulations during different stages of the design process, for example for cost- and quantity analysis, energy calculations, evacuation studies, and studies of activities during production and use of the building (Eastman 1991).

A *second* prerequisite for a computer integrated construction process is that CAD-software can be used not only as a product representation tool but also as a product design and modelling tool. This would enable the use of CAD already in the earliest stages of the design process. CAD-software must enable design and modelling of all the systems and processes that are formed, not only the building, but also the building user organisation, the production process, and the facility management process (Ekholm et al 1994).

A *third* prerequisite for carrying out a computer integrated construction process is the establishment of standardised and scientifically well-founded principles for structuring information in product models. During the design process a multitude of models are developed and different actors and computer systems must be able to interact and exchange information. In order to enable communication among actors and computer systems and to ensure conceptual consistency and compatibility among models, it is necessary that the models are built as well with scientifically well-founded concepts as by use of common classification systems and technical standards (ISO 1994a).

1.3 CAD – FROM COMPUTER AIDED DRAUGHTING TO COMPUTER AIDED MODELLING

An essential factor in realising a computer integrated construction process is a design process that develops and utilises computer based models, so called product models. CAD-software is mainly developed for a traditional design process where it is used as a draughting tool for producing production drawings in the later stages of the process. Currently research and technical development within CAD-software are directed towards modelling tools, where data in the computer can be structured to build a computer based model of an object (Galle 1995).

The introduction of building product models represent a revolution to information handling in the construction process. Information can be exchanged between computer systems and different actors without the need for human interpretation, thus accelerating the speed of information transfer and eliminating sources of human misunderstanding. Among the major research questions that have emerged, and which also have been given the most attention, are those concerning the structure of building product models and the transmission of building product data between different actors and computer systems throughout the construction process, e.g. GARM (Gielingh 1988), RATAS (Björk 1989), STEP (ISO 1994a), and COMBINE (Augenbroe 1995).

This research has concentrated on questions regarding buildings and other construction works, while additional systems and processes, that also result from the construction process, so far have been given less attention. Among such results are the building user organisation and the building management process. That also these results can be represented as computer based models is essential in order to achieve the overreaching goal of a computer integrated construction process.

The earliest stages of the construction process, the brief and the design proposal stages, deal not only with the building but also with the user organisation as an object for design. Every construction project has a brief-formulation stage which includes a description of the user organisation and its requirements on the building. The design proposal stage includes spatial layout and co-ordination of the user organisation and the building. CAD-software that could represent the user organisation would be useful not only in the brief-stage but also to facility management.

Today’s practice overlooks the fact that architectural drawings provide very little representation of the building’s occupants or the environmental context: ”One complete half of the ‘ensemble’ is almost completely missing” (Steadman 1979:184). Neither commercially available CAD-software for building design, nor software for product modelling developed in a R&D context provide explicit representation of the user organisation and consequently have limited use at the earliest stages of the design process.

Another essential issue which is not yet really approached in the building research community is the question of product design. Until now, work within building product modelling has presupposed that the essential properties of the building are determined before constructing the product model. The development of software to be used as an environment for both product design and modelling is still in its infancy and represents a challenge both to construction research and commercial software development (Junge 1995).

1.4 THE BAS•CAAD PROJECT

This paper gives an overview of some of the results of the BAS•CAAD research project at the division of Computer Aided Architectural Design, School of Architecture, Lund University. BAS•CAAD is an acronym for Building and User Activity Systems Modelling for Computer Aided Architectural Design. The BAS•CAAD project has the overall aim to contribute to the development of tools for computer aided design in the early stages of the construction process including the brief development stage.

Today there is a lack of knowledge about how CAD-software can support product design and modelling. The BAS•CAAD project addresses this problem by dealing on the one hand with principles for structuring product information, and on the other hand with requirements for a computer based information system for product design and modelling.

In the project we are developing a basic data model, a generic framework, from which different domain specific frameworks and product models can be built. This will ensure conceptual consistency and enable exchange of information between different models. The data model is based on a philosophically and scientifically well-founded property theory developed by Mario Bunge (1977 and 1979).

A prototype CAD-software that enables product design and modelling is currently being developed within the project. This prototype shall also be developed to enable spatial design and co-ordination of a building and a user organisation. The approach concerns questions of architecture and building science including questions of IT.

The results from the BAS•CAAD project are expected to be applied in different ways. The systemic principles for the data model may be of general interest for conceptual modelling in information systems development. The CAD-software prototype for product and process modelling may be further developed in applications, e.g. for building design and architectural programming. The theoretical foundation in the project may be of interest in other contexts like classification and standardisation.

2 Principles for structuring product model information

2.1 FRAMEWORKS, THEORIES AND MODELS

A representation of a thing, that resembles the represented thing in some way, is in everyday language called a model. A concrete model is a thing that physically resembles the modelled thing, while a conceptual model is a mental representation of a thing, made up of concepts that represent the real thing. A computer based model is a concrete model built into a computer, generally with the help of modelling software. A concrete model, be it in a computer or in clay, is a representation of a mental model, it is not a direct ”footprint” or a mould of reality. We do not model things directly ”as they are” but as we ”see” them in everyday praxis, expressed in a scientific theory, or in an artists vision. The consequences of this for computer software for design will be discussed further on.

In the context of computer based information handling a ”conceptual model” is also called an ”information model” (Schenck and Wilson 1994:10). Referring to computer based models, e.g. product models, we normally mean the information model and not a concrete representation in the computer.

It is of interest in this context to distinguish between frameworks, theories and models. A framework, or context, loosely defines the key concepts of a domain (Bunge 1983:323). A theory is more precisely defined, it is a logically organised context (ibid:331). A model concerns a specific member or group of members of a domain and is based on either a framework or a theory. If the model should represent the dynamical behaviour of the represented thing it must be based on a theory and not a framework. The fact that the term ‘model’ is often used both for models, frameworks and theories gives rise to an ambiguity in terminology. For a discussion see (Björk 1995:8) and (Ekholm 1996:2).

A computer based information system consists of a conceptual schema, an information base and an information processor (ISO 1985). The *conceptual schema* is a generic conceptual representation of the part of the real or formal world we are interested to handle information about. This part of the world is also called the *universe of discourse*, UoD. The conceptual schema is either a framework or a theory. If the modelling purpose is to simulate the behaviour of a product, then the specific relations among entities are of interest and the schema must be a theory. On the other hand, if the modelling purpose is only to transfer information about entities, then it is sufficient that the conceptual schema is a framework. The *information base* in the information system describes the state, i.e. the values of the attributes, of the UoD at a certain time. The *information processor* is a software tool that makes it possible to query and update the conceptual schema and the information base. A specific information model consists of a conceptual schema together with the attribute values in the information base.

In a traditional information system a *computer based model,* e.g. a product model, is defined on the basis of a predefined conceptual schema. The information processor allows the user to assign values to the attributes in the schema. An information system for product design is different in that the information processor also enables design operations on the conceptual schema in order to develop a conceptual model for a new and beforehand unknown product.

2.2 THE NEED FOR STANDARDISED AND SCIENTIFICALLY WELL-FOUNDED FRAMEWORKS

A product information model or, for short, *product model* is defined as ”an information model which provides an abstract definition of facts, concepts and instructions about a product” (ISO 1994a). In the context of computer based information transfer it is necessary that product models are based on scientifically wellfounded standardised frameworks in order to fulfil the requirements for unambigousness and exchangeability. Of interest to this study are three main abstraction levels of frameworks:

1. The ontological level, concerning very generic properties like object, property, thing, system, space and time common to every description of reality.
2. The domain level, concerning properties common to a certain technology or trade, e.g. construction or shipbuilding.
3. The application level, concerning domain specific artefacts, e.g. buildings, masts or dams.

That these levels are relevant for standardisation of frameworks is confirmed by the work within the STEP Project (Standard for Exchange of Product Model Data). STEP is an international standardisation activity within the International Standardization Organization, ISO; the objective of STEP is to enable ”unambiguous representation and exchange of computer-interpretable product data throughout the life of a product” (ISO 1994a). For example both the formal language EXPRESS, which is used for developing conceptual schemas, and the so called “Integrated resources“ in STEP, that are common to all or most product information models, belong to the ontological level. Models for different kinds of domain specific artefacts like high-rise buildings, steel structures or curtain walls, belong to the “Application protocol“ level (ibid). The domain level is not formally distinguished in STEP, but the Building Construction Core Model, BCCM, within the AEC-domain is a typical example of a framework at the domain level (ISO 1996).

Standardisation has a long standing international tradition in the construction context, for example through the international applications of the Swedish SfB-system[1].

---

1 SfB is an acronym for ”Samarbetskommittén för Byggnadsfrågor”, a Swedish organisation that developed the SfB-system for building classification (ByggAMA 1950).

Currently a framework for classification within the construction industry is under development as an ISO International Standard based on the Technical Report TR 14177 (ISO 1994b). The purpose of this standard is to be a basis for the development of national and international classification systems.

Among the objectives of the BAS•CAAD project are to develop principles for structuring frameworks as a basis for standardisation of information on the ontological level, especially the data model for the design tool, and on the domain level for buildings and user organisations.

2.3 AN ONTOLOGICAL FRAMEWORK FOR DESIGN

In order that frameworks and standards shall be mutually compatible and stable it is necessary that they have a sound scientific basis. In this study such a foundation is provided by a scientific[2] ontology developed by Bunge (1977 and 1979). A short presentation of the concepts of thing, property and system in Bunge's ontology follows here. A slightly more extended presentation is given in Ekholm (1996).

A *thing* is a concrete object, among its most basic properties is its existence (Bunge 1977:160). In order to exist, the thing must interact with its environment and therefore a thing has environment and structure (Bunge 1979:6), see Figure 1.

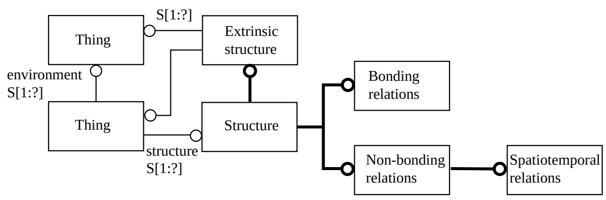

Figure 1. Thing[3]

2 In his "Treatise on Basic Philosophy", Mario Bunge discusses different views on what ontology is about. His own view is that the ontologist should "stake out the main traits of the real world as known through science, and that he should proceed in a clear and systematic way" (Bunge 1977:5).

3 The framework diagrams in this paper are developed in EXPRESS-G, the graphical counterpart of EXPRESS, a formal language which was developed in its present form to be used within STEP (Schenck and Wilson 1994). In the EXPRESS-G schemas the circle shows the direction of a relation towards the entity pointed at by the circle/arrowhead. A broader line indicates a subtype relation. The cardinality of a relation is indicated by the figures within the square brackets and expresses the number of entities that occur in a relation (ibid:316). The syllable letter S expresses that the entity at the arrowhead is a set in which the order of the instances is unimportant and that each instance may only be related once (ibid:145).

The environment is the set of things with which the thing interacts. A thing does not necessarily have parts, but it has an external structure which is the set of relations between the thing and its environment. A relation is a mutual property of the related things. For example the property to leave a graphite trace is a mutual property of a lead pencil and a paper. A mutual property is often described as an attribute of each of the related things; the term used for this attribute in information modelling is ”role”, e.g. the role of the pencil is ”marker”, and the role of the paper is”marking surface”. Relations may be bonding and non-bonding; bonding relations affect the state of the related things, while non-bonding relations do not. Relations to reference frames like spatial and temporal relations are non-bonding. The state of a thing is the values of its properties at a specific time.

A *system* is a complex thing with composition, environment and structure, both extrinsic and intrinsic, see Figure 2. In order to be a system it must have integrating, bonding, relations among its parts otherwise it is just an *aggregate* (Bunge 1979:6). The structure is the set of all the systems relations, both intrinsic and extrinsic. Intrinsic relations hold among the parts of a system; they are mutual properties of the parts. Intrinsic properties are basic to external properties, e.g. the pencil mechanism must hold and feed the lead in order that the pencil shall fulfil the role as a marker.

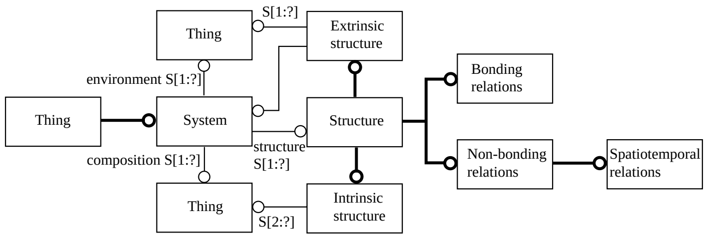

Figure 2. System

3 Frameworks for user organisation, construction works and construction space information

In order to accomplish a computer integrated construction process, principles for standardised computer based representations must be developed. In this section is shown how such principles can be based on the systems framework. The schemas in this section also illustrate how the basic representation of a system with composition, environment, and structure can be extended to represent domain specific systems by attaching domain specific system objects and property objects. In a CAD-program these domain specific objects would be found in a property library.

3.1 A FRAMEWORK FOR USER ORGANISATION INFORMATION

An organisation is normally described and organised in a planning process for administrative, business or production activities. The purpose of such a description is to understand the functioning of the organisation. If the organisation needs a building for its activities, the organisation's relations to the building are described in a building brief. The brief design stage starts with a description of the user organisation concentrating on those aspects that are of importance in relation to the use of buildings. Together with other information, e.g. requirements emerging from town planning, this description is used as a basis for defining the requirements on the building.

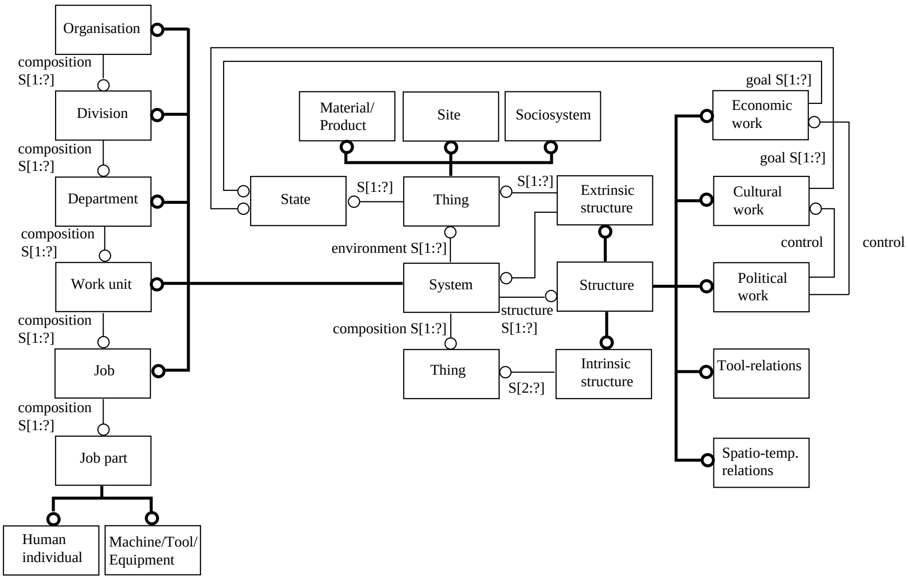

Figure 3. A framework for organisation information

In Ekholm and Fridqvist (1996) we define an *organisation,* see Figure 3, as a system with:

1. A *composition* of organisation units. The composition has a level structure with a lowest level of jobs. A job is composed of human individuals, and/or machines, tools or other equipment. Jobs combine into higher levels like work units, departments and divisions (Child 1984:85). Among the subsystems are an economic or cultural system, and a control system that manages the former (Bunge 1979:189).
2. An *environment* of things with which the system interacts and which is partly transformed by its work. To the environment belong the material resources transformed by work and the resulting products. Also sociosystems towards which the organisation directs its cultural work belong to the environment.

Finally the organisation has a physical site, for example a building or the natural environment.

3. A *structure* including an intrinsic social network based on work, with tool-relations to the artefacts in the system. Some members of the organisation manage others with respect to their work in the organisation. To the extrinsic structure belong the economic and cultural work as well as relations to the site. To the structure of an organisation belong its spatiotemporal properties.

3.2 A FRAMEWORK FOR CONSTRUCTION WORKS INFORMATION

The framework for system information has been applied for development of a framework for standardised classification of construction works in (Ekholm 1996). This section presents the basic structure of that framework. The concrete functionally distinguishable things that are produced as results of the construction process are *infrastructure units*, *construction works*, *construction work elements*, *element parts*, and *spaces* (ibid). The construction results are composed of resources of the construction process, e.g. *construction products* (ISO 1994b).

Construction works are specifically designed to be used in activities that require for example a loadbearing ground structure, servicing installations, or æsthetic and symbolic expression. Examples of construction works are buildings, streets, canals, bridges and parks. Buildings are a specific kind of construction works, they are built to accommodate user organisations, and have an enclosed space for control of climate or protection against intruders. Construction works are aggregated into larger so called infrastructure units, characterised by a common location and by being used by a social organisation for a purpose, e.g. the construction works of a university campus or an airport.

The functionally defined parts of construction works are elements and element parts. An element like a gypsum wall with a space enclosing function may be composed of element parts like loadbearing scantlings and enclosing gypsumboards. An element part does not have the complete function required to be classified as element. An element part is composed of one or many assembled and transformed construction products, and constitute the lowest level of construction artefacts, they do not have the complete function required to be classified as elements (Ekholm 1996).

The internal bonding relations between the parts of a building can be caused by gravitation or by fixture devices. Among the external relations are the functions to the users and the transfor-mation-relations to the site, both of which are bonding relations. Among the non-bonding relations are the spatial relations, and indirectly, the interpretation-relations to those who experience the building as a system and a

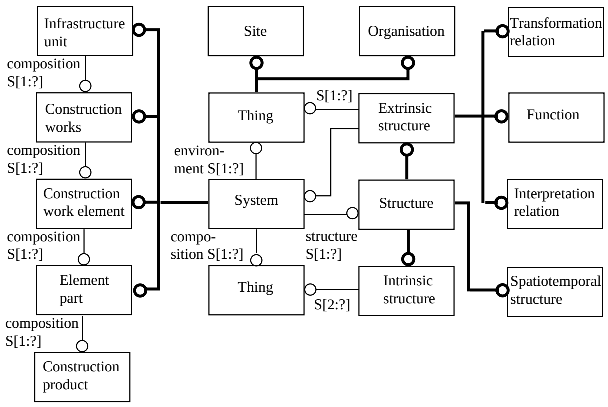

Figure 4. A framework for construction works information

sign, in order to appreciate its architecture and history. These concepts are related in a framework for construction works information, see Figure 4.

1.1.1A framework for construction space information

In the construction context, space is treated as a factual material thing. For example, a space function program developed in architectural programming contains requirements on the buildings spaces, e.g. for surface materials, fire resistance and sound reduction levels. The properties of spaces in buildings are designed for occupancy of users, machinery and equipment. Spaces are classified by their basic function in relation to the users and other agents, for example office and communication spaces, and climate- and fire-zones (ISO 1994b).

In order to develop a framework for space in the construction context, a generic definition of space is necessary. In Bunge's Ontology, a spatial relation is a mutual property of things; spatial relations are non-bonding relations. Using the convention of naming things after their properties, the term *space* denotes a collection of spatially related things (Bunge 1977). The referent of the concept *space* are things, and the concept represents their spatial properties. This concept of space is practical in construction, but contrasts to our perception of space as an intangible object enclosed by surfaces of things.

According to this definition any collection of things may be regarded as a space; both a constellation of stars, an alley of trees and a room of building elements are spaces. In the construction context spaces are constituted by construction works and their parts. Of specific interest to the users are spaces that can be used for occupancy and that have enclosing functions. Ekholm (1996) has proposed the following definition of space in the construction context: ”A space in the construction context is an aggregate of construction works, their parts and other

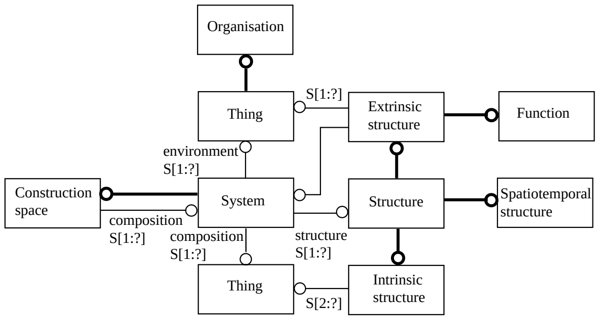

Figure 5. A conceptual schema for construction spaces

things defined only with regard to their materially or experientially enclosing properties”. See Figure 5.

The user organisation that uses the building’s spaces belongs to the environment of the spaces. The question of which parts belong to the construction spaces and which belong to the organisation depends on the context. For example, in one case a bookshelf may be considered a building element belonging to a construction space, while in another case the bookshelf is considered part of the organisation.

Spatial layout design deals with the spatial co-ordination of spaces and the user organisation located within the construction spaces.

4 A facetted approach to design and modelling

4.1 ANALYSIS AND SYNTHESIS TOOLS

When discussing problems in technical and scientific research, Bunge recognises two main types of problems, analysis and synthesis: ”The analysis problem is: Given a system (i.e. knowing or assuming its composition, environment, and structure), find its behaviour. The inverse problem is that of synthesis: Given a behaviour, find or design a system that realises it” (Bunge 1983:274).

This fundamental difference between analysis and synthesis is reflected in the design tools that can be used to assist these processes. In (Fridqvist and Ekholm 1996) we suggest a division of design tools into two groups: analysis tools and synthesis tools. *Analysis tools* are used for evaluating designs, i.e. problem solutions obtained through synthesis. Examples of analysis tools are software for cost estimation, stress analysis and energy balance calculation. Typical for analysis is that once the system is known, the task of finding solutions to problems regarding the system will follow given rules. Thus analysis tools can be highly automated, provided the necessary data is at hand in a suitable form.

Bunge's definition of synthesis has as a consequence that properties must be possible to define and describe separately from the things that have them. A *synthesis tool* is used for creating designs. In (Fridqvist and Ekholm 1996) we propose that a synthesis tool has three characteristics:

1. It assists the designer initially to describe the desired behaviour or functions of the thing to be designed.
2. It supports composing and describing things that have these functions.
3. It presents data in ways that are suitable both for analysis tools and for the realisation of the design.

A consequence of the first characteristic is that a synthesis tool not only must be able to represent the domain objects, but also the designer's way of thinking about the objects.

4.2 DETERMINATION OF PROPERTIES IN THE PROCESSES OF CLASSIFICATION AND DESIGN

In a design process the main objective is to find a solution consisting of parts that together satisfies the requirements on the resulting whole. During the process the understanding of the requirements on the whole as well as the necessary properties of the parts develops gradually and the conceptual model gets increasingly more specific (Lawson 1990:91). However, the process does not unfold simply by subclassing from generic to specific, instead it is a search process characterised by addition and removal of attributes from the conceptual representation . For example in building design an internal wall may initially have the function loadbearing, but later in the process the loadbearing properties may be removed and finally the wall properties may be reduced to a separation function held by a textile curtain.

A starting point for the BAS•CAAD project is that a design tool must enable a free combination of properties. How such a system would be organised may be conjectured by making an analogy to principles for classification applied in the context of information retrieval. Two different approaches to classification can be distinguished, the enumerative and the faceted (Hunter 1988).

The principle behind an enumerative classification system is that all relevant classes, simple and compound, are listed. An object is classified, according to its properties, in one of the available classes in the classification system. A problem with this kind of classification is that objects that have new kinds or new combinations of properties, which are not represented in the classification, cannot be classified properly. In a design system based on similar principles as the enumerative classification, the designer would have to choose among objects with a predetermined selection of properties. The advantage of this is that the design system can be optimised for handling well known combinations of properties, as for example concrete slab or steel truss. The drawback is that only things corresponding to the predetermined classes can be modelled.

The principle behind a facetted classification system is that a thing is classified by a combination of properties collected from different facets. A facet is a set of properties of similar kind like characteristic function, material, shape etc. To design according to this principle would mean that the designer determines the properties of an object by combining properties from different facets. For example the object may be determined to have a certain 3D-shape listed in a shape facet. The shape can be further specified through the interface of a geometric modeller. Among other properties it would be possible to decide whether the object is composed or homogeneous, or to determine its material, etc.

By enabling a ”free” combination of properties the designer would be able both to start on a high abstraction level and to create new kinds of parts. In a practical design system a combination of these two principles would be favourable. Sometimes it would be necessary to build a combination of properties and at other occasions it would be practical to choose a part with a predefined compound set of properties from a library.

4.3 CLASS-CENTERED AND OBJECT-CENTERED APPROACHES TO DATA MODELLING

The enumerative and the facetted approaches to product classification and design are seemingly equivalent to the class-centered and object-centered approaches to data modelling, respectively, discussed by Garrett and Hakim (1994). In the following, information modelling refers to the development of conceptual representations of the UoD, while data modelling refers to the implementation of the conceptual model into objects in the database, see Björk (1995:26) for a discussion of this distinction.

In a *class-centered* approach an information model is implemented as computer based classes. Commonly a representation of a specific domain object is created as an instance of one such class. For example, the concept of physical parts of the building with a loadbearing wall function is implemented as the class ”loadbearing wall”. A representation of a specific loadbearing wall is created as an instance of the class.

A problem arises when the class-centered approach is used for implementation of a modelling tool for design. Since an instantiated object inherits all the characteristics (attributes and methods) of the class, addition of a new attribute may not be consistent with the class definition and may require a transfer of the instance to another class. For example a design change from loadbearing wall to non-loadbearing wall would require a reclassification of the instantiated wall-object. Problems of pre-determined classification of product model objects has been discussed by Junge (1995), Garrett and Hakim (1994) and in papers concerning different software systems developed for general representations of design data like EDM-2 (Eastman et al 1995), and Feature based modelling (van Leeuwen et al 1993).

The class-centered approach does not comply with the requirement that a synthesis tool must allow a free specification of properties, separate from the things that have them. For example to be able to represent the design of a brick wall and its change from loadbearing to non-loadbearing, an object would at a certain time be associated to attributes like ”enclosing”, ”vertical plate”, ”loadbearing”, and ”brick”, and at a later stage have ”loadbearing” removed. In analogy with the facetted approach to classification this may be called a facetted approach to design.

According to Zhao and Biliris, such a functionality would be achieved by an object-centered approach to data modelling (Zhao and Biliris 1993). Furthermore, object-oriented data models can be grouped into two main categories: class-based and prototype-based. Class-based models focus on classes while prototype-based models focus on objects. The *object-centered* approach proposed by Zhao and Biliris combines these two methods and ”achieves dynamic object evolutions, flexible object classification, and strong typing”.

4.4 THE BAS•CAAD DATA MODEL

The object-centered approach is developed within the context of computer science. In the BAS•CAAD project we have chosen to build a data model that enables a facetted approach to design, on the basis of Bunge’s property theory. The reason for this is our hypothesis that the data model of the design tool and our approach to information modelling must be mutually consistent.

In order to build a data model that can be used for product design, we need to implement a view of the world that conceptually separates the individual from its properties. Such a view is needed in both science and design, for example in science the question is ”what are the properties of this thing?”, and in design the question is ”what properties must this factually possible thing be given?”. This separation makes it possible to question the properties of a thing without questioning its existence.

According to modern science, the world consists of things characterised by their properties. In this view there are no ”bare” things without properties and there are no ”free” properties. In spite of this, it is in both science and design convenient to assume the existence of bare individuals that can be equipped with different kinds of properties. Such an approach is supported in Bunge’s ontology, where a thing is understood as a substantial individual with substantial properties (Bunge 1977:162).

This reflects the way a designer thinks during the design process. Although the properties of a designed object change, the designer may still treat the transformed object as having the same identity through the design process. An example is the situation when a wood-carver works on a piece of wood to make a bowl; the original piece of wood gradually transforms into a bowl, but to the wood-carver it keeps its identity through the transformation.

According to this view it is possible to at the same time maintain the identity of a thing and be uncertain of its properties. In the BAS•CAAD data model this is achieved by introducing the class of individuals possessing the ”existence” property. The members of the ”individual object” class are ”bare” individuals without other properties than a feigned existence; the members of ”property” classes are attributes, i.e. concepts representing real properties of things.

A short presentation of the basic objects in the BAS•CAAD data model is made here as a summary from Fridqvist and Ekholm (1996). The data model distinguishes between three main object classes: System object, Property object and State value object.

System objects alone represent only the identity and existence of a concrete thing. System objects have the attributes *composition, structure* and *environment.* The structure is the set of all the systems relations, both internal and external. The composition and internal structure are basic to the intrinsic properties of the system, while the system, its environment and the external structure are basic to the extrinsic properties of the system. The properties of a system are modelled by associating a Property object with a System object.

Property objects represent properties of concrete systems. A Property object is instantiated only once, to ensure compatibility between different modelled objects. Consider, for instance, the property ‘mass’: In the real world there is only one kind of mass. This should be reflected in a model, so that there is only one instance of property object that represents ‘mass’. A Property object always has a state function with a state space. The *state space* represents the possible states of the associated systems with respect to the property.

State value objects represent the actual value of a property’s state function with respect to a specific system object. For example the property ‘colour’ has a state function for traffic lights that can take the state values ‘red’, ‘yellow’ and ‘green’.

Property objects may be grouped in domain characteristic property libraries, for example a property library for building design will hold properties specific to buildings. The development of property libraries will have to be guided by analyses of the different design domains the tool is intended to be used within.

System objects, Property objects and State value objects are to be determined by design operations through a user interface. Characteristics of design operations and design moves have been discussed in an earlier paper by Fridqvist and Ekholm (1996).

The actual implementation in the BAS•CAAD project of System object, Property object and State value object is made in Smalltalk, an object-oriented development environment. The principles for the design and use of the model design tool are to be further developed and tested in a prototype currently under development. This testing will include the development of a small building model and a model of a user organisation. The aim is to enable a spatial representation and co-ordination of the systems as in a lay-out process.

References

Augenbroe G. (1995). Combine 2. Final report. Delft: Delft University of Technology.

Björk B.-C. (1989). Basic structure of proposed building product model. *Computer aided design* Vol. 21, No 2, pp. 71-78, 1989.

Björk B.-C. (1995). Requirements and information structures for building product data models. Espoo: Technical Research Centre of Finland, VTT Publications no 245.

Bunge M. (1977). Ontology I: The Furniture of the World, Vol. 3 of Treatise on Basic Philosophy. Dordrecht and Boston: Reidel.

Bunge M. (1979). Ontology II: A World of Systems, Vol. 4 of Treatise on Basic Philosophy Dordrecht-Boston: Reidel.

Bunge M. (1983). Epistemology and methodology I: Exploring the world. Vol. 5 of Treatise on Basic Philosophy. Dordrecht: D. Reidel Publishing Company.

ByggAMA 1950 (1950). ByggAMA 1950, Allmän material och arbetsbeskrivning för husbyggnadsarbeten. Stockholm: A V Carlsons Bokförlags AB.

Child J. (1984). Organization: A guide to problems and practice. London: Paul Chapman Publishing Ltd.

Eastman C. M. (1991). The evolution of CAD: Integrating multiple representations. *Building and Environment*, Vol. 26, No 1, pp. 17-23, 1991.

Eastman C. M., Assal H., and Jeng T. (1995). Structure of a database supporting model evolution. In *Modelling of buildings through their life-cycle.* Proceedings of CIB workshop on computers and information in construction (eds. Fisher M., Law K., and Luiten B.) Stanford University, Stanford, Ca, USA, August 21-23.

Ekholm A., Fridqvist S., af Klercker J. and Ljunggren N.-O. (1994). Building and user activity systems modelling for computer aided architectural design. Paper presented at the IT-BUILD conference in Lund 1994. Lund: CAAD, School of Architecture, Lund University.

Ekholm A. (1996). A conceptual framework for classification of construction works. *Electronic Journal of Information Technology in Construction (ITcon)* Vol. 1. Stockholm: Royal Institute of Technology. URL: http://itcon.fagg.uni-lj.si/~itcon/.

Ekholm A. and Fridqvist S. (1996). Modelling of user organisations, buildings and spaces for the design process. In *Construction on the Information Highway*. (Ed. Ziga Turk). Proceedings from the CIB W78 Workshop, 10-12 June 1996, Bled, Slovenia.

Fridqvist S. and Ekholm A. (1996). Basic object structure for computer aided modelling in building design. In *Construction on the Information Highway*. (Ed. Ziga Turk). Proceedings from the CIB W78 workshop 10-12 June 1996 in Bled, Slovenia.

Galle P. (1995). Towards integrated, ”intelligent”, and compliant computer modeling of buildings. *Automation in Construction*. Vol. 4, No 3, pp. 189-211, 1995.

Gielingh W. (1988). General AEC reference model. External representation of product definition data. Doc. no 3.2.2.1, TNO-report BI-88-150, Delft, The Netherlands.

Garrett Jr J. H. and Hakim M. M. (1994). Class-centered vs. Object-centered Approaches for Modelling Engineering Design Information. Proceedings of the IKM-Internationales Kolloquium über Anwendungen der Informatik und der Mathematik in Architektur und Bauwesen, pp. 267-272, Weimar, Germany, March 16-18, 1994.

Hunter E. J. (1988). Classification made simple. Hants: Gower.

ISO (1985). Concepts and terminology for the conceptual schema and the information base. ISO/DTR 9007 (TC97), also SIS teknisk rapport 311. Stockholm: SIS.

ISO (1994a). Industrial automation systems and integration - Product data representation and exchange - Part 1. ISO 10303-1:1994(E). Geneva: International Organization for Standardization.

ISO (1994b). Classification of information in the construction industry. ISO Technical Report 14177:1994(E). Geneva: International Organization for Standardization.

ISO (1996). Building construction core model, BCCM, ISO 10303. ISO TC184 SC4 WG3 N496. Geneva: International Organization for Standardization.

Junge R. (1995). Aspects of new CAAD environments. *Modelling of buildings through their life-cycle.* Proceedings of CIB workshop on computers and information in construction (Eds. Fisher M., Law K., and Luiten B.) Stanford University, Stanford, Ca, USA, August 21-23.

Lawson B. (1990). How designers think. Second edition. Oxford: Butterworth Architecture.

Leeuwen J. P., Wagter H. and Oxman R. (1995). A feature based approach to modelling architectural information. In *Modelling of buildings through their life-cycle.* Proceedings of CIB workshop on computers and information in construction (Eds Fisher M., Law K., and Luiten B.) Stanford University, Stanford, Ca, USA, August 21-23.

Schenck D. A., and Wilson P. R. (1994). Information modelling: The EXPRESS Way. Oxford: Oxford University Press.

Steadman P. (1979). The evolution of designs. Cambridge: Cambridge university press.

Zhao H. and Biliris A. (1993). An Object-centered Data Model for Engineering Design Databases. In *Proceedings of the Third International Symposium on Databases for Advanced Applications.* Daejon, Korea, April 1993, pp 133-140.