View Integration in Building Design

SANDER VAN NEDERVEEN*

ABSTRACT

The design process of a building is a process of co-operation of many different participants. Each participant has his own perception of the building and uses his own building model. Such participant specific building models are called (actor) view models. The combination of multiple view models and multiple companies involved, makes it very difficult to manage communication in the design process.

This paper presents an approach for the management of different view models. Following this approach, discipline specific information is defined in view models. View models can communicate to each other via a model kernel, which is formed by the overlapping of the view models. The view model structure is illustrated with a case study which develops two view models of a wall. The integration of these two models illustrates how the kernel model can be constructed. Models for other disciplines can be added to the view model structure. Two models for conceptual design views are presented and discussed briefly. The presented view model structure can function as a basis for integration of computer applications. However, regarding CAD systems, results are expected to be more modest.

Key Words

view integration; building information models; product modelling; integrated design; management of design information

INTRODUCTION

First, the role of views in the building process is described. Next, the notion of views in information technology is discussed and compared to the needs of the building industry.

Views in Building Design

In a building project, many partners from different companies work together. Every partner uses existing information and creates new information. A lot of the information used is provided by other partners. As a result, a lot of information is exchanged between partners. Fortunately, none of the

* *TNO Building and Construction Research, Computer Integrated Construction, Delft University of Technology, Civil Engineering, PO Box 49, 2600 AA Delft, The Netherlands.*

partners needs all project information. To carry out his specific task, each partner only needs a subset of the project information. For example, a structural engineer needs information only on the structural characteristics of the building and its parts. On the other hand, a partner usually does not process the provided information directly, but transforms it into information he can process. To continue the example, a structural engineer may get shape and material data and transforms this into a structural schema with loads and so on, or into a finite element model. To carry out his calculations the engineer creates a specific model of the building which accommodates his specific view. This kind of view specific models are created by every partner throughout the building process. These so called view models allow the partners to work with a considerably smaller amount of information which is appropriate for them to do their job.

An issue which is related to the existence of participants with different views, is the management of responsibility. Partners not only have a specific task in the project, they also have specific responsibilities. As a result of this, partners have to meet agreements on who is responsible for what information, who is allowed to create and edit what information. The distribution of responsibility has a strong influence on the creation of view models. If a partner is responsible for certain aspects, these aspects will be included in his view model. Aspects, which are beyond the responsibility of a partner, will be excluded. In this way, the view mechanism can help to manage responsibility (Van Dam *et al*, 1992). Summarized, the view mechanism helps to provide partners with only the information they need, and to manage the responsibility for different design aspects.

Views in Information Technology

In many recent projects researchers have been working on the conceptual specification of integrated building information models. Well known models in this context are the General AEC Reference Model, or GARM (Gielingh, 1988), the AEC Building Systems Model (Turner, 1989), the RATAS model (Björk *et al*, 1989) and the EDM (Eastman, 1991). The aim of these models is to provide an information structure in which all building data can be stored in an integrated way, including information on the building and its parts, and their properties and relations during the different building life cycle stages. This approach is known as product modelling, and is closely related to the development of the ISO standard STEP (STEP, 1992). In all of the mentioned models important abstraction mechanisms, such as decomposition and specialization are worked out extensively.

Following the previous section, the view mechanism also seems to be very important for the management of building information. In database technology, the notion of views is already quite common for many years. This goes back to the ANSI/X3/SPARC reference model for database management systems (Burns *et al*, 1985) in which a distinction into internal, conceptual and external views was proposed.

The importance of the notion of views was also recognized in the building information models discussed earlier. In several models, model structures are used which focus on certain aspects of the building. Turner introduced the use of separate models for different building systems, for example the spatial system, the circulation system and the structural system. These systems are described in separate models, each of which focuses on one aspect. A similar approach is used in the RATAS model. The EDM does not define different building systems, but offers mechanisms on a more abstract level. An EDM mechanism which distinguishes different building aspects is the accumulation mechanism. This mechanism focuses on the accumulation of properties of a single aspect in an assembly. In the EDM decomposition is defined as the result of multiple accumulations applicable to the same assembly.

These examples show that the use of aspect specific model structures, is already quite common. However, the discussed models do not explicitly distinguish between discipline specific information. In this paper, this distinction is the starting point, which leads to the formulation of separate discipline view models.

Overview of Paper

In this paper, the notion of views and view integration is worked out in conjunction with the product modelling approach as used by the models mentioned earlier. In order to do this, the paper starts with an introduction of a number of basic concepts of product modelling and view integration. Next, a case study is presented, in which models of a simple wall are worked out for (1) the structural engineer's view, (2) the energy engineer's view, followed by an integration of both models using a kernel model. After that, the structure of the kernel model is further discussed. Within the kernel so the called system models are identified, which describe different building systems.

Then the relationship between views and applications is discussed, while making a distinction into technical applications such as calculation programs, and CAD systems. The paper finalizes with remarks on evaluation and future work.

BASIC CONCEPTS OF PRODUCT MODELLING AND VIEW INTEGRATION

In this section the basic concepts used for product modelling and view integration in building design are described.

Product Model and Product Type Model

A product model is an information model of a product, in which product data is stored in an integrated way, including information on the product parts, their properties, relations and behaviour, during different product life cycle stages. A product model describes an occurrence of a product, not a class or type. Example: a product model of the Eiffel tower.

A product type model is an information model of a product class, in which general information of the product class is defined in an integrated way, including information on the product parts, their properties, relations and behaviour, during different product life cycle stages. A product type model describes a class of products. Example: a product type model for buildings, or for office buildings.

In the case of real life products, the distinction between product model and product type model is clear: a model which describes an existing building is a product model, a model which describes a class of existing buildings is a product type model.

In the case of product design information, the distinction between a product and product type model is not always clear. In most product model literature, and also in this research, it is assumed that a design of a product is described in a product model, in which all design parameters are specified during the design process, including position and orientation. However, the use of such a design model may not be limited to one product occurrence. It can be used for many other products as well (as in the car industry), or as a prototype for a new product model. Thus a single design may serve as a specification for a set of products, which only differ from each other in serial number and in position. In such cases product models can be defined as mere references to a product type model, with a position specification (comparable to symbol instances in a CAD system).

On the other hand, existing models such as GARM, the RATAS model and the EDM are clearly not describing product occurrences. Such models describe general information of buildings, or even more general, of AEC products. Therefore, this kind of models are considered as product type models, not product models. The ideas presented in this paper are also dealing with product type model information.

View Model and View Type Model

A view model describes a discipline specific view of a product. A view model is part of a product model. A view model may overlap with other view models.

A view type model describes a discipline specific view of a product class. A view type model is part of a product type model. A view type model may overlap with other actor view type models.

The goal of these concepts follow from the function of the view mechanism stated earlier: provide partners with only the information they need, and manage the responsibility for different design aspects.

These concepts can be refined by a distinction between disciplines and actors (Tolman *et al*, 1993). In this way, the fact that actors may play different roles can be supported. This distinction is absolutely necessary to allow organizational differences in building projects, at least between different countries and ages, but certainly also between projects of almost the same place and time. For example, an architect may play the role of the project manager in one project, but not in another one.

Kernel Model and Kernel Type Model

A kernel model is formed by the overlapping parts of view models, (Figure 1).

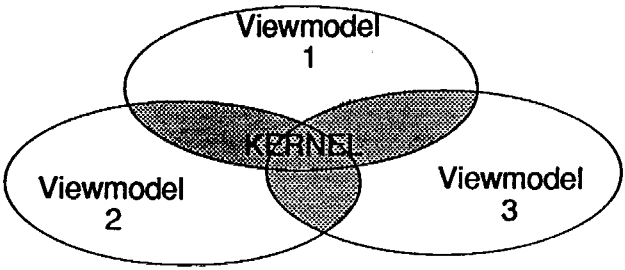

Figure 1. A kernel model is formed by the overlapping parts of view models. The union of all models is the product model.

The function of the kernel model is to provide a basis for communication between the different disciplines. As a result of this, the kernel includes exactly all information that is communicated between disciplines.

A kernel type model is formed by the overlapping parts of view type models.

CASE STUDY: SIMPLE WALL

In this section the concepts introduced in the previous section are illustrated using examples from a case study on a simple wall. The case study fully concentrates on the usage of the view mechanism. Therefore many other aspects of product modelling of buildings will remain out of scope.
These aspects include:

- the decomposition of buildings,
- the relationship between spaces and physical objects,
- the usage of generic concepts which can be applied to all objects, for example modelling constructs defined in STEP resource models.

These aspects will be discussed briefly after the example.

The Structural Engineer's View

First, a model is presented which describes the view of the structural engineering discipline on walls: a structural engineer's view type model. Such a model contains the information, which is needed or produced by structural engineers. This includes:

- an idealized shape description,
- an idealized material description,
- load information, and
- information on mechanical quantities (moments, forces, elasticity parameters) (since decomposition is out of scope, the wall is considered as a single object, which is not decomposed further).

Both the shape description and the material description should be idealized for the structural engineers' needs. This means that these descriptions do not include information which is irrelevant for the structural engineer. For example, a complex shape may be simplified to something like a mechanical schema. In this schema information such as colour data or detail design information, which have no impact on the structural design are left out. As a consequence, the shape and material description in this model is different from descriptions used by other disciplines.

The information discussed here, can be described using the modelling technique NIAM. The symbols in NIAM diagrams must be interpreted as follows:

- circles represent object types,
- double boxes represent relationships,
- arrows represent subtype-relations

For further description of the NIAM modelling language see (Nijssen *et al*, 1989).

The NIAM diagram for the structural engineer's view on a simple wall is shown in Figure 2.

The Energy Engineer's View

In the same way, a model can be defined for the energy engineering discipline. An energy engineer's view type model contains information, that is needed or produced by energy engineers. This includes:

- an idealized shape description,
- an idealized material description, and

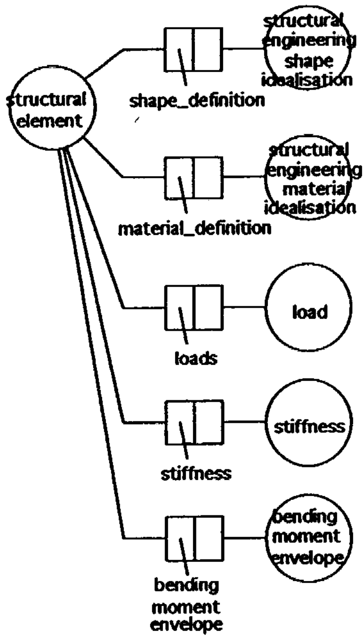

Figure 2. Structural Engineer's View Type Model of a Simple Wall

- information on energy quantities, such as transmission coefficients, u-values, etc.

Once again, the shape description and the material description are idealized for the discipline's needs. In this case information such as reinforcement details, are left out, since they are irrelevant for the energy engineer. Obviously the shape and material descriptions will differ from the descriptions in the structural engineer's view type model.

The information discussed here may be modelled in NIAM as in Figure 3.

Integration of the View Type Models

The next step is the integration of the two view type models. Since actors such as the structural engineer and the energy engineer need to cooperate in the design process, it is essential that they can exchange information between their models. This means that the relationship between the models must be specified.

The diagram in figure 4 shows how this can be done.

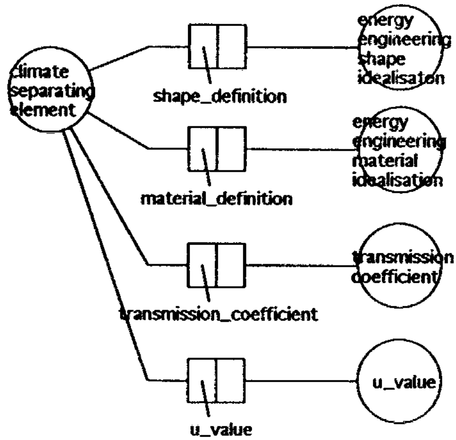

Figure 3. Energy Engineer's View Type Model of a Simple Wall

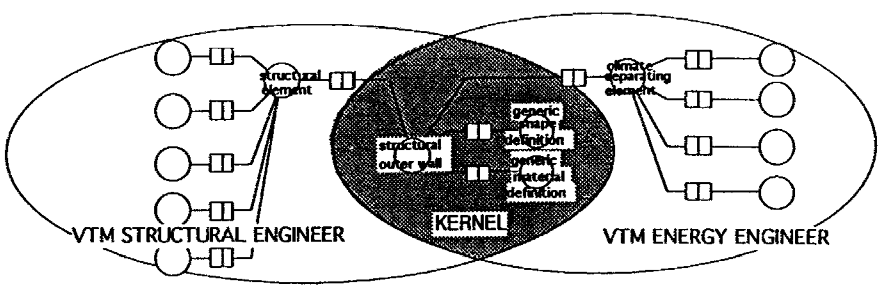

Figure 4. Integration of the View Type Models

The diagram shows that the view type models are integrated through a general entity 'structural outer wall', which has a 'generic shape definition' and a 'generic material description'. These generic descriptions are used in the communication between disciplines. The view specific descriptions of shape and material, may be derived from this generic description.

Note that the entities 'structural element', 'enclosure element', and 'structural outer wall' are all describing the same object. The information of the outer wall is distributed over the three entities. Part of this information are three different shape descriptions, which can be derived from one another. The rules for these derivations are included in the relationships between the view entities and the kernel entity. This type of information is view specific, so the relationships are not part of the kernel.

THE STRUCTURE OF THE KERNEL

In the case study, the contents of the kernel remain small and manageable. However, when other views are added, such as the architect's view and the HVAC engineer's view, the kernel will become a lot more complex. This is due to the fact that the kernel includes all information that is communicated between participants, which follows from the function of the kernel: to provide a basis for communication between the different disciplines.

System Models

For this reason it is necessary to add structure to the kernel. For this purpose, concepts of the existing building models discussed earlier, can be used. For instance, different building systems can be described in separate models, as in (Turner, 1989). Following this approach the following systems can be distinguished:

- the functional system,
- the space / enclosure system,
- the structural system,
- several HVAC systems.

The models of these systems can be regarded as aspect models: the models describe a certain aspect of the building, which corresponds to a certain set of properties. However, since the term aspect models is used with other meanings as well, we prefer to use the term system models here.

The difference between system models and view models may need some clarification. A system model is defined by an aspect of the product. It is considered as part of the kernel. It contains only information, which is used in the communication between partners. Conversely, it does not contain information which is only used by one partner. Finally, a system model uses a common language which is understood by all disciplines.

A view model is defined by the view of a discipline. It partly overlaps with other view models. The overlapping parts belong to the kernel, and may belong to system models. The private part of the view model uses a discipline specific language, or jargon. Nevertheless, system models and view models may have a strong relationship. It is clear that the structural engineer's view model will be strongly related to the structural system model.

The Functional System

Within building information, a subset can be defined which concentrates on the function of the building and its parts and the relationships between these functions. This subset is called the functional system. The functional system is limited to space functions. The reason for this is that physical objects usually have multiple functions (a load bearing wall, for example, also separates two spaces), which are easier to describe in other system models. Thus the functional system can be seen as the subject of functional analysis in a design process.

The functional system model starts with a general specification of the function of the building. This may be something like 'an office housing facility for 200 employees with facilities to have meetings with groups up to 50 people...'. The general building function can be decomposed into space functions. Furthermore, space functions can be decomposed into smaller space functions, but the number of decomposition levels cannot be fixed. These considerations lead to a rather abstract model, with a recursive decomposition construct, see Figure 5.

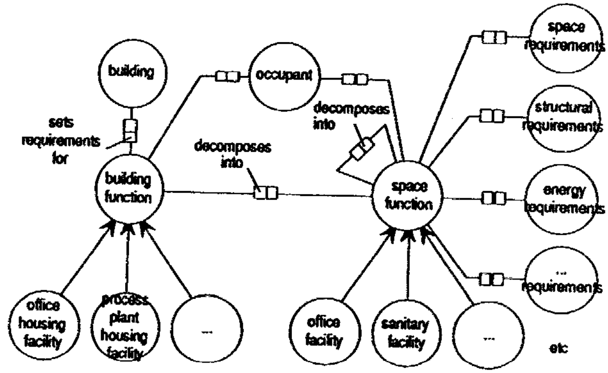

Figure 5. Overview of the Functional System Type Model

One could argue that building function is in fact also a space function, which accidentally is fulfilled by a building. This idea would lead to an even more abstract model, with only one entity space function.

The requirements (on the right side in the NIAM diagram), which are derived from the function of a space, form an interrelation with the other system models, such as the structural system model, and the space/enclosure system model, which is discussed below.

The Space/Enclosure System

The space/enclosure system of a building is the collection of spaces, space boundaries and enclosing structures of the building. The relationships between these entities have already been worked out in a number of models. Björk (1992) has recently published an overview of four of these models (the RATAS model by VTT, the Integrated Data Model by CSTB, the House Model by De Waard and the Synthesis Model by GSD), and has added a new model in which the other models are synthesized. In this study many important aspects of the space/enclosure system are identified and worked out quite extensively. The synthesized model may not be perfect, but it seems to be a good starting point for the space/enclosure system model. Figure 6 shows an outline of the synthesized model in NIAM. In this outline the assembly entities are left out, as well as subspaces (spaces which are only partially bounded by physical elements), and the component layer information.

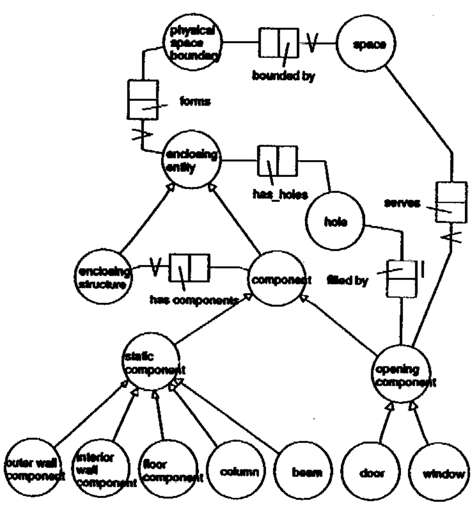

Figure 6. Outline of the Space, Space Boundary and Enclosing Structure Model by Björk

VIEWS AND APPLICATIONS

One of the intentions of the view model approach is the integration of computer applications used in design. For example, information can be interchanged between a structural design application and an energy calculation program through interfaces based on the discussed view models and the kernel model. This approach seems very well feasible for specialized disciplines such as structural engineering, energy engineering, HVAC design and so on. The data structures of the applications used in these disciplines usually fit quite well in the structures of the view type models.

This picture changes when the tools used for conceptual design are considered, such as CAD systems. The problem is that the data structure of a CAD system is usually geometry oriented, and contains little notion of functional, spatial and physical properties and structures. In fact, there is a significant gap between the way designers think, and the way CAD systems work. The ideas presented here can not only be regarded as an approach for meaningful data exchange, but also as a starting point for future design systems, which really work the way designers think.

As long as such systems are not available, we must try to achieve the best possible exchange between the current systems. For this purpose, the view approach is also useful. The models will become more simple, and the exchanged data will not be as meaningful as we think it can be. But progression of the research for data exchange between current systems is already of great value for the building industry.

EVALUATION AND FUTURE WORK

In this paper an approach for the integration of different views in building design is presented. For this purpose, view models and view type models are defined, and illustrated with examples. The models are not yet complete, and it is still a lot of work required to complete them. Nevertheless, the ideas on view integration have become clear in the presented models.

Other future work concerns implementation efforts. The conceptual models as presented here must be tested in implementations to find out whether they are useful for software development. The first implementation efforts have begun, using TNO's product modelling tool PMshell. In the near future, implementation and testing will play an increasingly important role in this research.

ACKNOWLEDGEMENT

Several colleagues have contributed to the ideas presented here, most notably Wim Bakkeren, Bart Luiten and Frits Tolman. Particularly the ideas on view models and system models have been described earlier in (Tolman, 1993). The author also had benefit from discussions in the Dutch research project NOBI and the ESPRIT project ATLAS.

This research is sponsored by the Technology Foundation (STW).

References

Björk, B C, Pentillä, H, Finne, C, Nervola, M, Saarinen, H, and Moisio, J (1989), "A Prototype Building Product Model Using a Relational Database", *Proceedings ARECDAO*, Barcelona.

Björk, B C (1992), 'A conceptual model of spaces, space boundaries and enclosing structures', *Automation in Construction*, Volume 1, Number 3, December.

Burns, T, Fong, E, Jefferson, D, Knox, R, Mark, L, Reedy, C, Reich, L, Roussopoulos, N and Truszkowski, W (1985), "Reference Model for DBMS Standardization," Tech. Rep., NBSIR 85-3173, May.

Eastman, C M (1991), "Use of Data Modeling in the Conceptual Structuring of Design Problems", *CAAD Futures '91*, Vieweg & Sohn Verlagsgesellschaft, Zürich / Braunschweig.

Gielingh, W F (1988), "General AEC Reference Model," Tech. Rep. BI-88-150, TNO Bouw, Rijswijk, the Netherlands, October .

Nijssen, G M and Halpin, T A (1989), *Conceptual Schema and Relational Database Design: A fact oriented approach*, Prentice Hall.

STEP (1992), *Product Data Representation and Exchange - Part 1: Overview and Fundamental Principles*, STEP document ISO TC184/SC4/PMAG.

Tolman, F, Van Nederveen, S and Bakkeren, W (1993), "A Systems Approach to Product Modelling", Tech. Rep. 93-BI-R0054, TNO Bouw, Rijswijk, the Netherlands.

Turner, J (1988), "AEC Building Systems Model", Michigan, January.

Van Dam, P R, Eeltink, M, Plokker, W, Taal, A C and Van Veghel, M L W (1992), *DUS boek*, Tech. Rep., VABI, Delft, April (in Dutch).

---

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