COMPUTER AIDED CONFORMANCE CHECKING

M. de Waard

Delft University of Technology, Faculty of Informatics, Department of Information Systems, P.O. Box 356, 2600 AJ Delft, The Netherlands.

TNO Building and Construction Research, Department of Computer Integrated Construction, P.O. Box 49, 2600 AA Delft, The Netherlands.

Abstract

In the building process a number of problems exist with regard to building regulations, causing the conformance checking process to be an island in the building process. This paper discusses an approach that enables us to perform computer aided conformance checking and to integrate the conformance checking process in the building process. The approach is based on the use of product models. This paper discusses theoretical as well as implementation aspects.

1. INTRODUCTION

The conformance checking process of building designs against building regulations is an activity that has to be performed during the building process. It is the point in the building process where a building permit is to be obtained, so that the actual construction of the building can begin.

The conformance checking process is still done by hand by the building authorities. If the building authorities use computers at all, the computers are not integrated with CAD systems or the like, making the information transfer from the design team to the building authorities and vice versa still done by means of paper documents. Reasons for not being able to integrate CAD systems with regulations software, if they exist, are given by Eastman [2]. According to Eastman the functionality of CAD systems is limited, due to what he calls a lack of 'architectural semantics'. Meaning that CAD systems should be able to define and manipulate architectural objects like rooms, walls, doors, etc. instead of lines, points, etc.

The approach proposes the use of product models [9]. The approach requires:

- an information model of residential buildings,
- an information model of building regulations, and
- a link between both information models.

The information model of residential buildings contains architectural semantics Eastman speaks of. This information model can also be used as a neutral exchange medium for informaton exchange between the design team and the building authorities and vice versa. The information model for building regulations considers building regulations as consisting of two things. A model of the objects that are subject of the regulations and constraints which limit the attributes and characteristics of, and relationships that can hold between, objects. The building regulations considered are those stated in the new national building code called the Building Decree, where only the requirements for residential buildings to be build are considered. The link between the information model of a building and that of the building regulations describes the mapping of the first information model on the second. The model of the objects that are subject of the regulations form a building regulations view on the building model. By describing how this view can be derived from the view independent model, computer aided conformance checking can be achieved.

2. INFORMATION MODEL FOR RESIDENTIAL BUILDINGS

There are two ways of looking at a building, also seen by CIB-W74. The one is to look at a building as a spatial system consisting of spaces, the other is to look at a building as a building elements system consisting of building elements. Both systems are related to each other, because part of the building elements system, like walls, floors and ceilings, make up the spaces from the spatial system. This is why an integration of the two systems is to be achieved. This is done by introducing an artificial entity type called 'space boundary'. A space is now always bounded by a space boundary and a space boundary can be formed by a separation structure. The kernel of the information model for residential buildings looks as depicted in the NIAM [8] model of figure 1. The kernel can be further specialized, by specializing the entity types from the kernel and as such gradually introduce more and more architectural semantics. The kernel as depicted in figure 1 can also be used for other buildings, as can be seen in Nederveen [7]. Next to the kernel of the product type model the GARM [4] is used as a reference model to describe residential buildings. For further detail on the information model for residential buildings, please refer to [11].

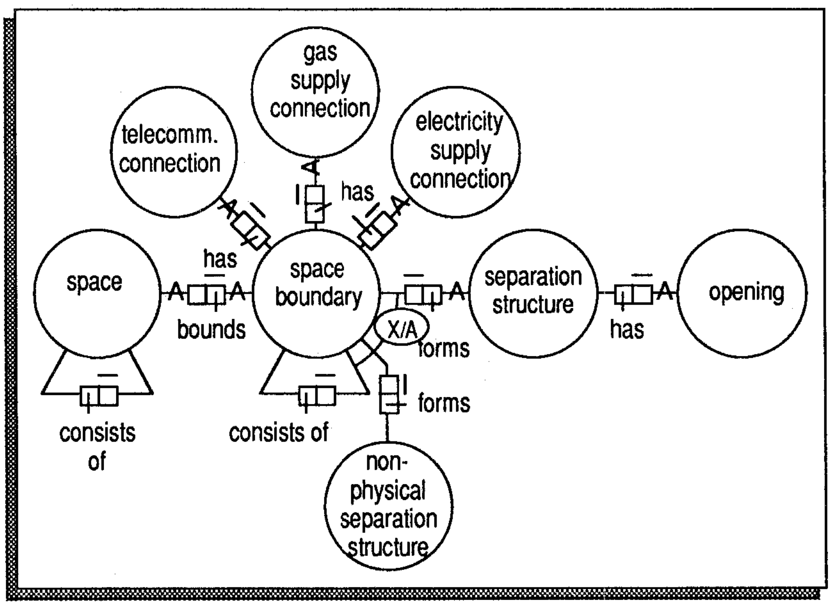

Figure 1. Kernel of the product type model for residential buildings

For the presentation of the product type model for residential building a choice is made for NIAM for the global models. The detailed models will be described in Express [5], as NIAM has some limits as to what can be modelled with it.

3. INFORMATION MODEL FOR BUILDING REGULATIONS

There are different ways to look at building regulations. Looking at building regulations from a GARM point of view, it can be said that building regulations limit the admissible technical solutions for a certain functional unit. The Building Decree gives requirements for functional units that their technical solutions must meet. The properties from the technical solution should either directly conform to the requirement or the result of an analysis, where a property can be used as input, should be conform to the requirement. This is presented in the NIAM model of figure 2.

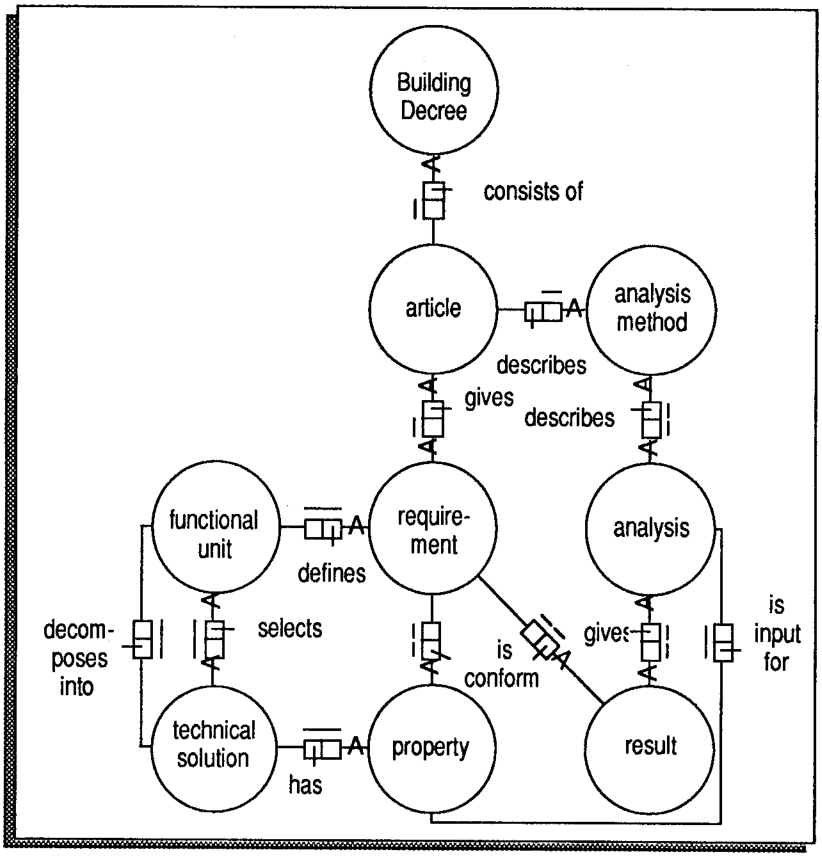

Figure 2. Relation between the Building Decree and the GARM

Seen from a database point of view, one can say that building regulations are constraints over the building model, see Fenves [3]. From an information point of view, it can be said that building regulations, apart from giving constraints over the building, also decribe a building. This is also seen by Cornick et al. [1] when they state: *'Building regulations such as the Building Code can be separated into two conceptual entities. Models which describe the objects that are subject of the regulation and constraints, which limit the attributes and characteristics of the objects defined in the regulations. Constraints also place limits on the relationships that can hold between objects'*. In modelling building regulations, one should not only model the constraints, but also model the objects. Cornick et al. state [1]: *'At present, regulations only implicitly contain models of objects within their scope and subsystems. These models need to be made explicit for incorporation into intelligent design systems'*. The way of modelling building regulations proposed here intends to do just that. In modelling building regulations four steps are proposed:

- make a NIAM model of the entity types involved in the requirements under consideration;
- translate the NIAM model to an Express model;
- refine the Express model by adding attributes to the entity types;
- refine the Express model by adding functions, procedures and extra constraints.

A NIAM model can be constructed by looking at the requirements under consideration and collecting all the nouns in these requirements. Model these nouns in a NIAM model and give the interrelations between these nouns. In GARM terminology, these nouns constitute functional units, for which technical solutions should be found. The next step is to translate the NIAM model to an Express model. This is done because not everything in the requirements can be modelled in NIAM. However, NIAM has a good presentation that can be used for discussion about the model. The next step is to refine the Express model by adding attributes, necessary for the determination of satisfaction or violation of the considered requirements. The last step is to add functions, procedures and extra constraints. The functions and procedures are used to describe the rules to determine whether a requirement is satisfied or violated. Some requirements can not be modelled by the use of functions and procedures. They are modelled by adding extra constraints to the model. Functions and procedures can also be used to model constraints. For an example of the way of modelling building regulations, please refer to [10].

The four steps described before can be applied to each requirement in the building regulations. This way an Express model for each requirement is generated. If necessary, the models for all the requirements can be combined, so that an overall model of the building regulations develops. The model that then arises is in principle a 'simplified' product type model for residential buildings, the building regulations view on the product type model for residential buildings. It is possible to combine the three viewpoints, the GARM viewpoint, the database viewpoint and the information viewpoint, in one model. The NIAM model, describing the entity types subject of the regulation, can be expressed in GARM concepts. The subjects of the regulations are the functional units in the model. Technical solutions for the functional units can easily be defined. The attributes that are added to the Express model after the translation of the NIAM model to Express, are properties of the technical solutions. The constraints given by the regulations are modelled by a combination of Express constraints, functions and procedures, where the latter two are mainly used for describing the analysis method defined in the requirement.

The building regulations can now be modelled apart from the product type model, which gives advantages as described by Fenves [3]. Cornick et al. [1] even state: *'By separating the models from the constraints imposed by the codes we hope to provide a system which can be adapted to changes in the regulations by changing the constraints on the models, rather than having to make changes to the models themselves. Over time, the models in regulatory codes are stable and the changes that do occur are generally associated with constraints on the models'.*

4. COMPUTER AIDED CONFORMANCE CHECKING

In the previous sections an information model both for residential buildings as well as building regulations was described in global terms. To be able to achieve computer aided conformance checking the two information models must be related to each other. In this section the relation between the two models will be described and how that helps in achieving computer aided conformance checking.

4.1. Relating the information model for residential buildings and the information model for building regulations

To describe the relation between the two information models we will first have a look at the three schema framework for database management systems, as defined by ANSI/SPARC. The framework proposes to distinguish three schemas for a database management system, see figure 3. The most important is the conceptual schema. It comprises a unique central description of the various information contents that may be in the database. The second important schema is the external schema or rather the external schemas. Users and application programs may view the data in a variety of ways, each described by an external schema. Each external schema is therefore derived from the common conceptual schema. The last schema is the internal schema which describes the physical storage structure. The product type model for residential buildings is a model that can be found on the level of the conceptual schema. The product type model seen from the viewpoint of the building regulations is found on the external schema level.

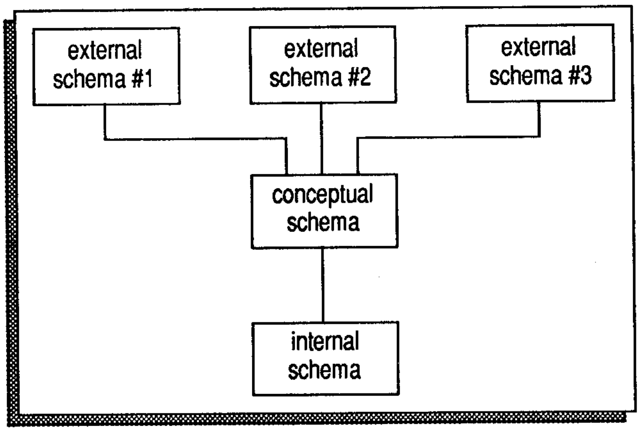

Figure 3. The three schema framework defined by ANSI/SPARC

Seen in terms of Nederveen [7], it can be stated that the information model for building regulations consists of many aspect models. An article or a combination of articles from the Building Decree, deals with one aspect of the building. Aspects dealt with in the Building Decree are safety, usability, health and energy. If the models dealing with the same aspect are combined, different aspect models develop. Modelling an article from the Building Decree is modelling a sub-aspect model.

A question that remains to be solved is how the two information models can be related to each other. The product type model for residential buildings is used as a view independent model, see [7]. The model is described using GARM concepts. The information model for building regulations also describes a product type model for residential buildings, only this time the model is view dependent. The view is that of the building regulations. The model is also described using GARM concepts. As the product type model defined by the building regulations is a view dependent model, it must be possible to derive the view dependent model from the view independent model. The view dependent model differs from the view independent model with respect to entity types and relationships between entity types considered.

The Building Decree describes a product type model for residential buildings in more global terms than the view independent model does. An example of this is that in the view independent model entity types like 'living room' and 'bed room' are found. In the building regulations view, these spaces are both called a 'residing space', which is still a specialization of the entity type 'space' from the kernel model. For these cases a simple generalization/specialization tree can be build, where all the entity types from the view dependent model are referred to by the view independent model as in figure 4.

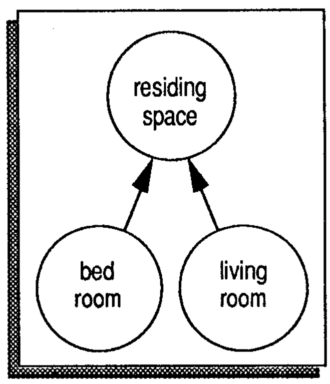

Figure 4. Example of mapping entity types from the view independent model on the view dependent model

Another change seen in the view dependent model are the relationships between entity types. Some relationships have to be redefined, because an entity type relating two other entity types to each other is not of interest in the view dependent model. Some relationships can be redefined. An example of this is i.e. a 'gas supply connection'. In the view independent model, as can be seen in the kernel depicted in figure 1, a gas supply connection is related to a 'space boundary'. However, building regulations don't state that gas supply connections should be on a certain wall, but they state that a certain space must have a gas supply connection. So the relationship between a gas supply connection and a space boundary in the view independent model, can be replaced by a relationship between a gas supply connection and a space in the building regulations view model. The relationship is however also present in the view independent model, only indirectly.

4.2. The computer aided conformance checking process

The steps in the computer aided conformance checking process are now as follows. The design team either work directly with the product type model for residential buildings, or make sure that whatever model they use can be translated to that product type model. The model is used as a neutral exchange format for exchange of information between the design team and the building authorities. The building authorities receive the model in neutral format. The neutral format is translated to the building regulations view on the neutral format. Within the view dependent model, the constraints as defined by the building regulations are found as constraints, functions and procedures. Each requirement is now checked for applicability to the design at hand. If the requirement is found applicable a consistency check is performed. If the model is found to be consistent, the requirement is satisfied, otherwise the requirement is violated. Doing this for all the requirements in the building regulations comprises a conformance check of a building design against building regulations.

5. IMPLEMENTING COMPUTER AIDED CONFORMANCE CHECKING

The implementation of the two models is done according to a layered structure, as proposed by Willems et al. [12]. Layers distinguished are at least a reference model layer, a product type layer and a product model layer. On the reference model layer, abstract models are found. Typical models found on this layer are the GARM and the kernel of the product type model for residential buildings. Both these models are used as reference models to develop a product type model for residential buildings. Typical models on the product type layer are the product type model for residential buildings, the road design model etc. These are models that are used when product models are being made. The structure of the product model is based on the structure of the product type model.

5.1. Implementing the product type model for residential buildings

The product type model for residential buildings is implemented in a layered way. To be able to make a product model for a residential building based on the product type model at least three layers are distinguished. These are the reference model layer, the product type layer and the product model layer, see figure 5. On the reference model layer models like the GARM and the kernel of the product type model can be found. Also other abstract models useful for developing product type models could be found here. The product type layer inherits from the reference model layer, it uses the reference models described in the reference model layer. Entity types from the product type model are specializations of the entity types defined in the different reference models where the inheritance structure is sometimes a multiple inheritance structure. Entity types inherit at least from the GARM and possibly also from other reference models such as the kernel. On the basis of the product type model, defined on the product type layer, a product model can be made using the structure defined in the product type model. The product model is an instantiation of the product type model. It is not always clear what can be found on which layer. An example of this is the kernel for the product type model. The kernel can be placed on the reference model layer, as is done in figure 5, but the kernel can also be modelled as a sub-layer of the product type layer. However, it is not important on which layer it is defined. Even a number of extra layers can be defined more abstract than the reference model layer, as proposed by Willems et al. [12].

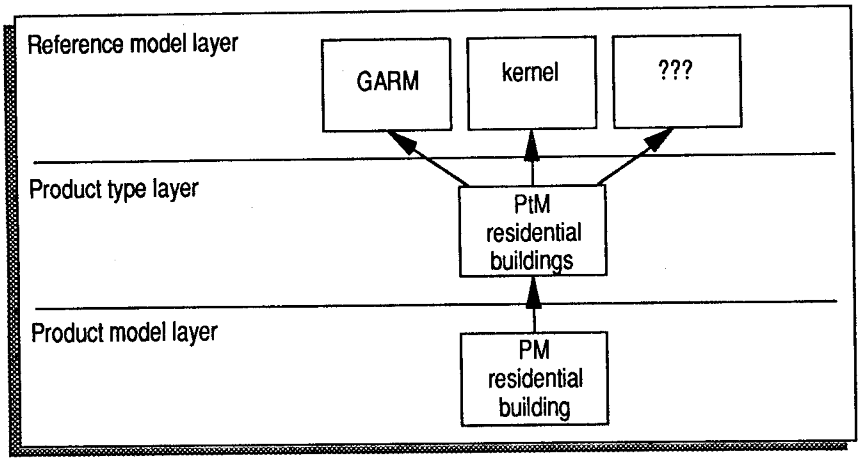

Figure 5. The layered structure of the product type model for residential buildings

5.2. Implementing the model for building regulations

The model for building regulations is implemented in a similar way as the product type model. A similar layered structure is defined, only this time the product model layer is not used by the end-user. The product model layer is automatically filled by an application that translates a product model defined according to the structure of the view independent product type model, to a product model defined following the structure of the product type model seen from the viewpoint of the building regulations. The models on the more abstract levels are defined in a similar way as for the view independent product type model.

5.3. Tools for implementation

For the implementation of this multi layered structure software has been developed at TNO Building and Construction Research, called PMshell (Product Modelling shell) implemented in the object oriented language Eiffel [6]. PMshell works directly on a database. Based on the database the software is able to generate Eiffel class templates, which can be further refined towards an actual implementation. The product type model described in Express can now be translated to an Eiffel implementation in the following way. An Express parser directly fills the database. Because of the layered structure this is done in multiple steps. First the models on the reference model layer are parsed, because the product type models are based on these models and they have to be known before a product type model can be made. The next step is to parse the Express models on the product type layer. The database is now filled with entity types from the product type model. On the basis of the database Eiffel class templates can be generated automatically. This is the first step towards the implementation of a product modeller for residential buildings. By adding functions and procedures to the Eiffel classes a product modeller is constructed. The product modeller is then used to fill the last layer, the product model layer. With the product modeller an end-user is able to instantiate the product type model and as such define a product model.

6. CONCLUSIONS

The paper has globally discussed the different information models developed in the research, that have been described in detail in other papers written. Furthermore, the paper discussed the relation between the developed information models, where the ANSI/SPARC three schema framework was used. Seen from this framework the product type model for residential buildings is seen as the conceptual schema and the product type model for residential buildings according to the Building Decree as one of the external schemas. Within each of the schemas discussed, a number of layers can be distinguished according to which an information model can be developed in an evolutionary way, with the advantages as discussed by Willems et al. [12]. The layered structure is also used for the implementation. If necessary, more layers can be added to the three layer structure described here.

REFERENCES

[1] S.M. Cornick, D.A. Leishman, J.R. Thomas, 'Integrating Building Codes into Design Systems'. In: K. Kähkönen, B.C. Björk, Proceedings of the International Workshop on Computers and Building Regulations, 27-29 May, 1991, Espoo, Finland.

[2] C.M. Eastman, 'Architectural CAD: a Ten Year Assessment of the State of the Art'. In: Computer Aided Design, Vol. 21, No. 5 (June 1989), p. 27-36.

[3] S.J. Fenves, J.H. Garrett Jr., 'Knowledge Based Standards Processing'. In: Artificial Intelligence, Vol. 1, No. 1, 1986, p. 3-14.

[4] W.F. Gielingh, General AEC Reference Model (GARM), BI-88-150, TNO Building and Construction Research, Rijswijk, October 1988.

[5] Exchange of Product Model Data - Part 11: The Express Language, Document CD 10303 - 11, ISO TC184/ SC4 N64, July 1990.

[6] B. Meyer, Object Oriented Software Construction, Prentice Hall, 1988.

[7] G.A. van Nederveen, F.P. Tolman, 'Modelling Multiple Views on Buildings'. In: H. Wagter, Proceedings of the CIB W78 Seminar 'The Computer Integrated Future', 16-17 September, 1991, Eindhoven, The Netherlands.

[8] G.M. Nijssen, T.A. Halpin, Conceptual Schema and Relational Database Design, a fact oriented approach, Prentice Hall, Sydney, Australia, 1989.

[9] M de Waard, F.P. Tolman, 'Implementing the Building and Regulations View on Buildings'. In: H. Wagter, Proceedings of the Second CIB W78 & W74 Joint Seminar on Computer Integrated Construction, 17-19 September, 1990, Tokyo, Japan.

[10] M. de Waard, F.P. Tolman, 'Modelling of Building Regulations'. In: K. Kähkönen, B.C. Björk, Proceedings of the International Workshop on Computers and Building Regulations, 27-29 May, 1991, Espoo, Finland.

[11] M. de Waard, F.P. Tolman, 'A Model to Describe Residential Buildings'. In: Proceedings on the Second International Workshop on Computer Building Representation for Integration, 3-5 June, 1991, Aix-les-Bains, France.

[12] P.H. Willems, P. Kuiper, G.T. Luiten, B.F.M. Luijten, F.P. Tolman, 'A Framework for Evolutionary Information Model Development'. In: H. Wagter, Proceedings of the CIB W78 Seminar 'The Computer Integrated Future', 16-17 September, 1991, Eindhoven, The Netherlands.