

A feature based approach to modelling architectural information

Citation for published version (APA):

Leeuwen, van, J. P., Wagter, H., & Oxman, R. M. (1995). *A feature based approach to modelling architectural information*. (BIT Notes; Vol. 9502). Eindhoven University of Technology.

Document status and date:

Published: 01/01/1995

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

A FEATURE BASED APPROACH TO MODELLING ARCHITECTURAL INFORMATION

Jos P. van Leeuwen
Harry Wagter
Robert M. Oxman

BIT Notes 1995 / 2

Building Information Technology Notes

(BIT Notes)

This is a series of notes of the section Bouwinformatica of the faculty of Architecture, Building and Planning Eindhoven University of Technology. Since many of these are preliminary versions, or may be published elsewhere, they have a limited distribution only and are not for review.

ISSN 1381-3129

editors:

ing. Jan Dijkstra
ir. Jos P. van Leeuwen
prof. dr. Robert M. Oxman
prof. ir. Harry Wagter

Copies can be ordered from:
Secretary of the section Bouwinformatica,
Eindhoven University of Technology
Faculty of Architecture, Building and Planning
Postvak 20
P.O.Box 513
5600 MB Eindhoven
The Netherlands

© 1995
Eindhoven University of Technology,
faculty of Architecture, Building and Planning,
section Building Information Technologies.

Other issues in the series of BIT Notes are:

- 94/1 Future Organisation of the Building Process.
Information Technology in the Building Industry.
Geert T.A. Smeltzer.
- 94/2 Future Studies in Construction.
Information Technology in the Building Industry.
Geert T.A. Smeltzer, Jan Dijkstra, Rob H.M. van Zutphen.
- 94/3 The Application of Virtual Reality Systems in Architectural Design Processes.
Geert T.A. Smeltzer, Jo M.M. Mantelers, Walther A.H. Roelen.
- 94/4 NOBI / BBB Project Bouwkosten en bestek.
Rapport prototype.
Rob H.M. van Zutphen, A.J. Jessurun, P. de Jong.
- 94/5 NOBI / BM Project Het Intechno Bouwkundig Model.
Eindrapport.
Erik P. Deiman, J.C. Hubers, Jos P. van Leeuwen, Geert T.A. Smeltzer,
Harmen C.A. Tacke, Bauke de Vries, Rob H.M. van Zutphen.
- 94/6 Architectural Product Modelling.
A Case Study.
Jos P. van Leeuwen, Rob H.M. van Zutphen.
- 95/1 A Proposal for a distributed Virtual Reality system for Architectural simulations.
A.H.A. Brouwer, W.A.H. Roelen, J.A.G. Jess

A Feature based approach to modelling Architectural Information

Jos P. van Leeuwen
Harry Wagter
Robert M. Oxman

Eindhoven: Technische Universiteit Eindhoven, Faculteit
Bouwkunde, sectie Bouwinformatica.

Keywords: Computer aided design; Product modelling;
Feature based modelling.

© 1995 Technische Universiteit Eindhoven,
faculteit Bouwkunde, sectie Bouwinformatica.
Alle rechten voorbehouden. Niets uit deze uitgave mag worden
verveelvoudigd en/of openbaar gemaakt door middel van druk,
fotocopie, microfilm, elektronische media of op welke andere wijze
ook, zonder voorafgaande schriftelijke toestemming van de
uitgever.

All rights preserved. No part of this publication may be reproduced,
stored in a retrieval system, transmitted or utilized in any form or
by any means, electronic, mechanical, photocopying, recording or
otherwise, without the prior written permission of the publisher.

Contents

1.	Introduction.	5
1.1	Current approaches in Product Modelling.	5
1.2	What is lacking in Product Modelling?	5
1.2.1	Dynamic modelling.	5
1.2.2	Adequate support for early design stages.	6
2.	Feature Based Modelling concept.	6
2.1	Introduction to Feature based approaches in Mechanical Engineering.	6
2.1.1	Classifications of Features	7
2.1.2	Processes of Feature modelling	7
2.2	Concept of Features in Mechanical engineering.	8
3.	Feature Based Modelling of architectural information.	8
3.1	Current applications of Feature Based Modelling in construction industry.	8
3.2	Extensive application of Feature Based Modelling in Architecture.	8
3.2.1	The relation between FBM approaches and PM issues.	8
3.2.2	Feature types for Architecture.	9
3.3	Definition of a Feature.	9
3.3.1	Contents of a Feature definition.	9
3.3.2	Standardisation of Features.	10
3.3.3	User-defined Features.	10
3.4	Modelling with the Feature Based approach.	10
3.4.1	Advantages of the FBM approach.	10
3.5	Communication of Feature models.	11
4.	Discussion and future research.	11
	References	12

Abstract.

Formal representation and communication of building information are main issues in the development of Building Information Technology. The development of standardised information models starts to find practical application in the construction industry. However, a major problem in these developments is formed by the conflict between static information models and the dynamic nature of Architectural design. The research presented in this paper investigates the approach developed in the field of Mechanical Engineering on Feature Based Modelling. This approach offers the formalisation of design information in a flexible manner that allows designers to change and extend the information model according to their needs. The paper presents the concepts of applying the Feature Based Modelling approach in the area of modelling Architectural information, and discusses the requirements for standardisation and system-development related to these concepts.

1. Introduction.

Modelling architectural information is a task that can be generally characterised by the complexity of the information and processes involved in architectural products. This complexity is contrived by the complex nature of architectural products, including a great amount of interrelated aspects of information, and by the many participants involved in the processes concerning the product, the many different views on the product. Modelling this complex information aims at improving the representation of the information and at enhanced communication. In order to model information unambiguously, it is necessary to define how information can be represented formally, which is the main problem addressed by today's research in the area of Product Modelling [van Leeuwen & van Zutphen 1994].

In this section we discuss the current approaches of Product Modelling research and indicate what we think is lacking in these approaches. The next section reviews the concepts of Feature Based Modelling, used mainly in Mechanical Engineering to model product parts. We then show how these concepts, if integrated in a Product Modelling approach, can be used to address the problems indicated below.

1.1 Current approaches in Product Modelling.

"Product data is data that describes the function and physical characteristics of each unit of a product from its requirements at inception to its configuration at time of retirement" [Wilson 1993]. This definition of product data delineates the kind of complexity as described above. The purposes of Product Modelling (PM) include the representation of product data throughout the life-cycle of a product. This involves modelling multiple aspects of the product, such as aspects of design, costs, assembly or manufacturing, planning, etc. These aspects can in turn be seen from different viewpoints of parties involved, which makes product data view-dependent over many stages in the product's life-cycle. For the purpose of enhanced communication, product data needs to be either exchanged between parties [de Vries & Wagter 1995] or integrated in shared models. Standardisation of data definitions then becomes very important.

Numerous Building Product Models have been defined in research projects, some are being used in practice. Early research projects resulted in models that are either generic for a type of building or specific for a particular domain. In the first group we find projects such as the RATAS project which presented a rather generic information hierarchy for buildings [Björk 1989]. More specific models represent a subset of the life-cycle of a building, for instance for the domain of HVAC engineering or structural engineering. Current research moves into the direction of complete life-cycle models, mostly defining a generic or kernel model to which view-specific models are related, as for example in the COMBINE project [Augenbroe 1993].

Approaches of PM research can be generally described as comprising some kind of information hierarchy that defines entities or objects as parts of the complete product. These hierarchies are established by means of abstraction mechanisms, such as decomposition and specialisation, which allow information to be handled at different levels of abstraction. Decomposition is used to define a 'has a' relationship between entities of different levels in the hierarchy. Specialisation is used to define an entity as a subtype, with special characteristics, of one or more entities that are found one step up in the hierarchy, an 'is a' relationship. Eastman & Fereshetian [1994] describe these abstraction mechanisms and relationships in detail.

This common approach to PM is also reflected by the STEP developments on standardised exchange of product data, ISO 10303. In STEP information requirements are collected and formalised at different levels. At a relatively generic level the so-called Integrated Resources (IR) provide data definitions which are application (or view) independent, e.g. data definitions for geometry, shape representations, materials. Based on the IRs, but more domain specific are the data definitions provided in the Application Protocols (AP). APs define the interpretation of the data definition in the IRs according to a particular domain. Examples of APs are the AP for Building Explicit Shape Representation (AP 225) and the AP for Building and Construction which is currently under development. The formal definition of the data models in the IRs and APs is established using the data definition language EXPRESS, which encompasses mechanisms for specialisation and decomposition. The actual modelling of products and exchange of data conforming to the STEP standards are based on the definitions given in the Application Protocols.

1.2 What is lacking in Product Modelling?

1.2.1 Dynamic modelling.

The developments of Application Protocols for AEC and for Building and Construction in particular have again raised the discussion in the STEP community whether it is possible to define a global schema for buildings. A global schema should not impose any limitations on the information requirements of buildings throughout their life-cycle. How would such a schema handle the evolution of construction-technologies, the

development of new products and materials? Moreover, design itself has a very dynamic nature, not only as a problem-solving process in which the search for solutions has many dynamic aspects. Also because design as a long term, ongoing development process results in new methods for design and new construction-techniques. These future developments are unlikely to be anticipated in a global schema which therefore will prove to be a limitation on design. The discussion in the STEP AEC group has been moderated and summarised by Wittenoom [1995].

The issue raised here is not the one that has mainly been addressed by PM research, namely the integration of multiple views into a common conceptual information model, but rather how changes of this conceptual model in time can be dealt with. In fact, this question is preceded by the question if at any time a satisfactory, generic definition of data can be given to cover every technique, product or method that is devised at that time in architectural design. We contend that maybe this would be possible at a very abstract level, but not at the level of detail that is required for adequate use in practice. An analysis in [Ramscar 1994] demonstrates that PM approaches cannot achieve a complete model that is necessary for an integration or exchange standard.

If we cannot define an adequate global schema, we will need to find a way to define a schema that can be extended and updated according to the changes of its requirements. In other words, a global schema for AEC, or for Building & Construction in particular, needs to be flexible enough to allow changes and additions of entities and structure in order to fulfil other or future requirements that are not considered when this global model is defined.

1.2.2 Adequate support for early design stages.

Another drawback of current PM approaches is found in early design phases. While we recognised before that design as a long-term process changes construction techniques and develops new methods and products, this innovative character of design is also found on a smaller scale in every design-task. Design is a dynamic process in which information is constantly re-interpreted and restructured, especially in early phases which concern ill-defined or poorly structured information. This makes modelling the information very hard. One approach to solving this problem is the development of intelligent sketch based CAD software, using Artificial Intelligence techniques such as frames [de Vries & Wagter 1992]. Another approach is to use case based reasoning mechanisms and formalisation of design knowledge, as in the cognitive models presented by Oxman & Oxman [1994].

If definitions and structures of information deriving from more definite stages of design are used in supporting early stages, there is the risk of hindering the creativity of design. A static information-model, such as defined in many Product Models, can hardly be expected to form an adequate support for this dynamic process. The inflexible structure and rigid definitions in which information needs to be specified would rather limit than stimulate the creativity of a designer. The designer has no control over the definition of either information format nor information structure and therefore constantly needs to translate the available information to the format and structure defined in the Product Model, and vice versa for requested information. This will be experienced as a strong limitation on the practical usability of Product Models. Designers need formal definitions of information so they can use computers to reason and communicate about it, yet they also want to handle this information in a very dynamic way and do not want to be confined to static definitions.

In addition to this, design phases often concern abstract concepts such as architectural function, aesthetic aspects, ambience, etc. This type of information is hard to formalise, but nevertheless plays an important role in design. Current PM hardly addresses modelling information of this type.

2. Feature Based Modelling concept.

2.1 Introduction to Feature based approaches in Mechanical Engineering.

In the 1980's, feature modelling came into development in the field of Mechanical Engineering to supply geometric modelling tools with a higher level of information. High-level models were necessary to extract information for generating process plans, GT codes (Group Technology), and NC programs (Numerical Control). Solid geometry modelers provide an incomplete product definition, consisting of only nominal geometry and low-level product definition, higher level information is missing. Solid modelling alone can not be used for process planning or manufacturability evaluation, because a higher level of information is required [Shah & Rogers 1988a]. Also in the early design-stage, high-level information is required for a full support of the process. The needs of the designer are to construct and reason about a model in terms that are derived from design-practice and that have particular functional significance to design.

Information entities called Features were introduced to fulfil the need for high-level information related to solid geometry models. Initially, from the view of manufacturing, Features represented shapes and

technological attributes associated with manufacturing operations and tools. From a geometric modelling point of view, Features were defined as groupings of geometric or topological entities that need to be referenced together. As the application of Feature modelling has broadened from manufacturing evaluation to product design, the definition of the term *Feature* in literature has changed accordingly: "Features are elements used in generating, analysing, or evaluating designs" [Shah 1991a]. A commonly acceptable definition of Form Features seems to be given by Shah [1991b]: "Features are generic shapes with which engineers associate certain properties or attributes and knowledge useful in reasoning about the product".

2.1.1 Classifications of Features

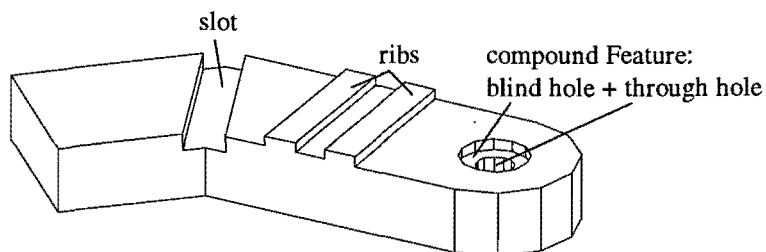
Many attempts have been made in literature to classify Features. One distinction can be made on the basis of the Feature's purpose, as by Shah and Rogers [1988a]:

- Form Features: elements related to nominal geometry.
- Precision Features: acceptable deviations from nominal form or size.
- Technological Features: related to performance and operation.
- Material Features: material composition, treatment, condition.
- Assembly Features: assembly sequence, orientation relative to the part.

or by van Emmerik [1990]:

- Form Features.
- Pattern Features: regular pattern of similar entities.
- Connection Features: geometric constraints.
- Property Features: properties not related to explicit geometry.
- Application Features: related to process planning requirements.

Concentrating on Form Features, several classifications are based on the shape of Features rather than on their application, distinguishing passages, depressions, protrusions, deformations, etc. The figure below shows some examples of Form Features defining a part.



Some examples of Form Features defining a part.

Although Form Features play a dominant role in the different phases in Mechanical Engineering (i.e. design, evaluation, manufacturing, etc.), other types of Features as mentioned in the above classifications are very important for product design support and automated reasoning.

2.1.2 Processes of Feature modelling

With the increasing application of Features, the processes of modelling Features have evolved. Initially, human interpretation of geometry was needed to specify the high-level information as Form Features. This process is called human-assisted or interactive Feature recognition. In automatic Feature recognition, high-level information could be automatically recognised and extracted from a geometric model [Henderson & Chang 1988] [Laakko & Mäntylä 1993] [Meeran & Pratt 1993]. The Feature model resulting from this process contained high-level information for evaluation purposes for instance in manufacturing, but the level of information was still limited. From pure geometric data, only a limited set of information can be extracted. Additional information is required to model for instance material properties and tolerances of parts of a product. Another disadvantage of Feature recognition is that a completed design and model is required before the recognition process can be performed. Valuable information available during design cannot be modelled and is lost.

The logical next step was to construct the model of the product not on a geometry-basis, but directly on the basis of Features, the entities of high-level information. This approach is called design-by-Feature [Shah & Rogers 1988a] [van Emmerik 1990] [De Martino et al. 1994]. Libraries of Features have been developed, but it

is commonly agreed upon that Feature modelling requires an extensible set of Feature definitions, allowing designers to define their own set of Feature types [van Emmerik 1990] [Shah 1991b]. The meaning of a Feature is therefore dependent on the context in which the Feature has been defined. This makes Features necessarily view-dependent in the complex life-cycle of a product. The communication of view-dependent information, however, requires translation of this information from one view to another. This translation of Features is devised by mechanisms for Feature mapping [Shah 1993] [Chamberlain et al. 1993].

The processes of Feature modelling today include both Feature recognition and design-by-Feature. Each of these approaches has certain benefits and systems are being developed that combine the two approaches [De Martino et al. 1994]. However, most of the current developments in Mechanical Engineering focus on either Feature recognition and extraction from geometry or design-by-Feature.

2.2 Concept of Features in Mechanical engineering.

In analysing the developments of FBM in Mechanical Engineering, we conclude that the concept of Features lies in the definition of collections of information with a semantic meaning to a specific view on a part of a product. These collections of information form independent units which, during the process of design and modelling, are used to define a part. The information concerned can be related to the shape of a part (Form Features), but the concept also includes any non-geometric information which is of significant meaning to a given engineering view in a product's life-cycle.

Systems for FBM in Mechanical Engineering allow users to define Feature types, which can be stored in libraries of Features, and to create instances of Feature types for modelling product parts. The structure relating the Features that compose a part is not pre-defined: the relationships between the Features are context dependent and defined by the user. On higher abstraction levels in the definition of a product, the aggregations of parts and modules, Features are not applied.

3. Feature Based Modelling of architectural information.

3.1 Current applications of Feature Based Modelling in construction industry.

The approach to information modelling as developed in Feature Based Modelling is applied in areas of the building industry that resemble Mechanical Engineering. Some PM researches report the utilisation of Features in structural engineering, where parts of a building's construction are modelled by Features [Watson, Crowley, et al. 1994]. These parts represent the construction as designed, and the describing Features are mainly Form-Features defined for manufacturing purposes. Although this application of the concept of Features is limited to Form-Features, it does demonstrate the relevance of Feature modelling for Architecture.

3.2 Extensive application of Feature Based Modelling in Architecture.

3.2.1 The relation between FBM approaches and PM issues.

The main aspect of FBM, as concluded in the previous section and which is of interest for Architecture, is the definition of collections of information with a semantic meaning to a specific view on a part of a product. Translated directly to the domain of building, a Feature can be defined as a collection of information with a semantic meaning to a specific view on a part of a building. For a specification of this very generic definition, we need to return first to the issues of Product Modelling discussed in the first section of this paper.

We concluded that the support of dynamic modelling is an important condition for Product Models. Dynamic modelling requires (1) that the definition and structure of information is not formed by a finite set of entities, but is extensible, and (2) that the structure of information is flexible so models can be defined to reflect closely the intentions of a specific point of view. These requirements are important for two reasons: first, we cannot assume any schema of information definitions to be complete, and second, we cannot presume a static, generic information structure to reflect sufficiently the individual intentions of parties in the processes of design and modelling.

However, we can try to identify a common basis for the definition of information entities and a generic mechanism for structuring these information entities. Here we find the common ground with the approaches of Feature Based Modelling.

3.2.2 Feature types for Architecture.

In Feature Based Modelling, without knowing in advance what product parts look like, the characteristics of parts have been formalised and defined in Features as high-level, independent units. During modelling, these Features are instantiated and related to each other in order to define a part. We can think of something similar for Building information in different ways.

Definition of physical parts.

Feature definitions would concern the characteristics of building parts, such as shape, function, physical properties, requirements, application constraints, planning aspects, etc. These definitions would be either generic for the entire domain of Architecture or provided from the viewpoint of the specific sub-domain in which the information is needed, e.g. design, manufacturing, assembly, construction, planning, etc.

The definition of these Features would be most similar to those in Mechanical Engineering, although Form Features would probably not play a similar dominant role.

Definition of non-physical concepts.

Information in Architecture is not always related to physical parts of a building. This is particularly true in early design, when the physical parts have not yet been decided upon, but also in later stages, when we still need to reason about the building in terms of spaces, functions, relations, constraints, which cannot be defined as attributes of physical parts. However, in a way similar to physical parts, we can define information describing the characteristics of these concepts by means of Features.

Information related to different levels of abstraction.

As opposed to the application of Features in Mechanical Engineering, we propose to apply Features at multiple levels of abstraction. Much the same to the Features that are used to model the properties of product parts, Features can be used to define the characteristics of the entities distinguished at any level of abstraction in a hierarchic Product Model. This way we can define information in terms of Features for the complete building, for spatial systems, for each floor in a building, etc.

Feature Based Building Model.

The eventual building model will exist by means of interrelated Features at different levels of abstraction, for different views on the product, and defining both physical and non-physical aspects of the building. This structure of Features is user-definable since a high rate of flexibility is required for the building model to be an adequate representation of the concepts of the design and the actual building.

3.3 Definition of a Feature.

Concluding from the above, we propose to define the term Feature in the context of Architecture as *a collection of high-level information defining a set of characteristics with a semantic meaning to a particular view in the life-cycle of a building*. These collections of information will be defined as units in what can be regarded as a vocabulary for information modelling. This vocabulary can then be used to define a Building Product Model of which the structure is determined by the particular view for which the model is defined. Features are defined as high-level units of information that may include knowledge such as the formal specification of their semantic meaning and procedural knowledge for modelling and evaluation. The actual domain of design and construction is rendered more closely by this high-level vocabulary for information modelling.

3.3.1 Contents of a Feature definition.

Features are defined to improve the representation of information. This means that Features need to reflect closely the intention of the provider of this information. Without presuming the actual contents of this intention, we argue that the information contained in a Feature will accommodate the following aspects:

- Factual information, the information that specifies the actual characteristics that need to be modelled. This includes numeric values, geometric data, referential data (i.e. references to other data in the model), etc. It also includes less formally defined characteristics, such as construction directives, descriptions of functions or requirements, etc.
- Descriptive information is the information that describes what is contained in the factual information, it specifies the semantics of the factual information, for instance the semantic meaning of numeric values including their units, range, constraints, tolerances, etc.
- Procedural information includes procedural knowledge related to the Feature. For instance how or where to obtain the information contained in this Feature, validation of the information, dependencies related to this information, behaviour of the Feature in relation to other information, etc. Also knowledge for Feature mapping can be contained within a Feature, as well as other evaluation procedures.

Features defined according to the above will have an independent character, meaning that they form entities of information that stand alone, independent of a specific context in an information structure. This means that characteristics such as material specifications, e.g. tensile stress, colour, permeability, etc. are no longer defined as attributes for a specific type of material, but as unambiguously defined units of information called Features. Another example: part properties such as thickness and height of a wall are instances of Feature types for thickness and height.

A major advantage of defining this type of information as autonomous units is that users, designers, are allowed to decide (1) which information they like to model, and (2) how this information will be structured. Also, we can begin to standardise this type of information without the limitations of static information structures that cannot be changed or extended.

3.3.2 Standardisation of Features.

Types of Features will need to be standardised, similar to the standardisation in the Integrated Resources of STEP, but at different levels of specialisation. Very generic Features can be defined for the complete domain of AEC, others more specific for the sub-domains of Architecture and Building & Construction. This is different from the standardisation in the Application Protocols in STEP because the APs define complete information structures, whereas the Features we propose to define do not impose a static, standardised structure.

3.3.3 User-defined Features.

Besides the standardised Feature types, users will need their own set of Feature types when the standards do not reflect well their needs, or simply because a new type of information needs to be modelled. The definition of Feature types should be based as much as possible on the standardised Feature types, using mechanisms of specialisation and aggregation. Specialisation can be used to define specialised versions of Feature types, by inheritance of knowledge from standardised types. Aggregation of standard types will lead to more complex Feature types that can be used to model compound sets of Feature information. Basing user-defined Feature types on standardised types will make future standardisation of new Feature types easier, and can enhance the translation of one Feature type to another.

3.4 Modelling with the Feature Based approach.

The task of information modelling in architectural design will need to change with this approach of Feature Based Modelling. In Product Modelling, the information to be modelled needs to be translated to the prescribed format and structure of the product model. In the FBM approach, information needs to be modelled by the following steps:

- Identification of Feature types matching the type of information to be modelled.
This can be done by either selection of adequate standard types, or by definition of user-types which can be added to libraries of Feature types. As a result, Feature libraries will represent particular design knowledge formalised by the designer.
- Instantiation of actual Features from Feature types, these instances will represent the particular building information. Since Features are representations of high-level information, modelling this information will be a process that is much closer to the domain of design and construction, as opposed to information models that require substantial abstraction of the domain-knowledge.
- Building the information structure by relating Feature instances into the model. The building model is a composition of inter-related Feature instances at different levels of abstraction, for different aspects of the building, possibly for different views on the building information.

Clearly, computer tools will play an important role in the support of this modelling task, ensuring that modelling does not become a burden for designers but in fact enlightens the design process.

3.4.1 Advantages of the FBM approach.

This new method of modelling has several advantages. Design and modelling with the Feature based approach will become more closely integrated. Design activities can be more precisely translated to modelling activities, since the vocabulary for modelling the design-outcome is not as pre-defined and static as before, but can be updated and adapted to particular needs. Especially the information structure can be adapted to the intentions of design, and can be handled in a flexible manner during different stages of design, allowing adaptation and updates as more information becomes explicitly known.

Another advantage of this approach to design and modelling is that design information is formalised not only within the particular building model, but also in the form of Feature types and libraries of Feature types as they are gradually defined during the process of modelling. Standardisation efforts on Feature types will result

in high-level standards of formalised architectural information that define the domain of architecture without enforcing static structures that conflict with the dynamic nature of design.

3.5 Communication of Feature models.

The exchange and sharing of information is affected in two ways by adopting the FBM approach. The loss of a pre-defined structure of information makes it harder to find a basis for communication: besides the actual building information, also the cohesion of the information needs to be conveyed as well. The process to do this, however, can be significantly supported by the level of information incorporated in the Features. The inclusion of for instance semantics and procedural knowledge takes away much of the ambiguity of data definitions.

The communication of Feature models will depend on:

- Interpretation of a model composed of Features and the determination of the semantic meaning of the interrelationships between Features.
- Translation of the interpreted model into the desired composition of Features. This includes translation of the semantics from one domain to another.
- The ability to translate Feature types of one kind to another. This is required in particular for the communication of non-standardised and unagreed Feature types.

4. Discussion and future research.

For enhancement of the domain of architectural design, by using support from Information Technology to improve the representation and communication of building models, a strict rationalisation of the domain is required. Yet the current approaches seem to fail, due to the incompatibility with the dynamic nature of the domain. In this paper we have presented the techniques of Feature Based Modelling and how we think that the utilisation of Features as independently defined entities of high-level information can be a successful approach to solving this problem.

Future research will be necessary to determine a framework for the standardisation of Feature types. Evidently, the standards will need to be developed in correlation with the STEP developments, in particular the Integrated Resources. However, the concept of Feature modelling requires a different role for these standards in modelling methodologies and a different implementation thereof in modelling environments. Standardisation of Feature types will be required at a generic level of the domain of AEC, but also at more specific levels, now covered by the Application Protocol developments.

Besides the standardisation of basic Feature types, standardisation is also required of the method of Feature type definition. This part of the standardisation is essential for the inter-operability of Feature type standards, for the communication of user-defined Feature types, and for the communication between different Feature based modelling or evaluation environments.

The Feature Based Modelling approach will also require dedicated environments for modelling information. Not only geometric modelers supporting Features are needed, as are developed already for the field of Mechanical Engineering. The approach that we presented in this paper requires an environment that supports defining and modelling both geometry and non-geometry related information. A prototype environment currently under development for Feature Based Architectural Information Modelling will basically support the following tasks:

- Feature type definition, creation of Feature type libraries. This task allows a designer or other user to define a Feature type, and to store it in a library of Feature types.
- Feature instantiation. The actual task of modelling the particular building information, by creating instances of previously defined or standardised Feature types. This task involves specification of values for the Feature instance, validation and evaluation of the instance.
- Feature based modelling, relating Features into a dynamic model. Newly instantiated Features are positioned in the building model as accomplished so far, validation of new Feature instances in relation to the existing model is necessary.

References.

- Augenbroe, G.L.M., ed. 1993.** Combine: Final report, CEC-JOULE-RUE report. Delft: Delft University of Technology.
- Björk, B.-C. 1989.** "Basic structure of a proposed building product model". *Computer-Aided Design*. vol. 21 no. 2. pp. 71-77.
- Chamberlain, M.A.; Joneja, A.; Chang, T.-C. 1993.** "Protrusion-features handling in design and manufacturing planning". *Computer-Aided Design*. vol. 25 no. 1. pp. 19-28.
- De Martino, T.; Falcidieno, B.; Giannini, F.; Hassinger, S.; Ovtcharova, J. 1994.** "Feature-based modelling by integrating design and recognition approaches". *Computer-Aided Design*. vol. 26 no. 8.
- Eastman, C.M. & Fereshetian, Nirva. 1994.** "Information models for use in product design: a comparison". *Computer-Aided Design*. vol. 26 no. 7. pp. 551-572.
- van Emmerik, M.J.G.M. 1990.** Interactive design of parameterized 3D models by direct manipulation. Ph.D. thesis. Delft: Delft University Press.
- Henderson, M.R. & Chang, G.J. 1988.** "FRAPP: Automated Feature Recognition and Process Planning from solid model data". *Computers in Engineering 1988*. vol. 1. pp. 529-536. San Fransisco: ASME.
- Laakko, T. & Mäntylä, M. 1991.** "Feature modelling by incremental feature recognition". *Computer-Aided Design*. vol. 25 no. 8. pp. 479-492.
- van Leeuwen, J.P. & van Zutphen, R.H.M. 1994.** "Architectural product modelling - a Case Study". *Proceedings of CIB W78 '94 workshop in Helsinki*.
- Meeran, S. & Pratt, M.J. 1993.** "Automated feature recognition from 2D drawings". *Computer-Aided Design*. vol. 25 no. 1. pp. 7-17.
- Oxman, R. & Oxman R. 1994.** "Cognitive models in design case libraries". *Automation in Construction*. vol. 3 nos. 2-3. pp. 113-122.
- Ramscar, M. 1994.** "Static models and dynamic designs, an empirical impasse vs. an inductive solution". *Proceedings of the first European Conference on Process and Product Modelling 1994*. Dresden.
- Shah, J.J. & Rogers, M.T. 1988a.** "Functional requirements and conceptual design of the Feature-Based Modelling System". *Computer-Aided Engineering Journal*. vol. 5 no. 1. pp. 9-15.
- Shah, J.J. & Rogers, M.T. 1988b.** "Feature Based Modeling Shell: design and implementation". *Computers in Engineering*. vol. 1. pp. 255-261. ASME.
- Shah, J.J. 1991a.** "Assessment of features technology". *Computer-Aided Design*. vol. 23 no. 5. pp. 331-343.
- Shah, J.J. 1991b.** "Conceptual development of form features and feature modelers". *Research in Engineering Design*. 1991 no. 2. pp. 93-108. New York: Springer-Verlag.
- Shah, J.J.; Hsiao, D.; Leonard, J. 1993.** "A systematic approach for design-manufacturing feature mapping". *Geometric Modeling for Product Realization* (ed. Wilson et al.). pp. 205-221. Amsterdam: North-Holland.
- de Vries, B. & Wägter, H. 1995.** "Message development in the Building process". *These proceedings of CIB W78 '95 workshop at Stanford University*.
- de Vries, M. & Wägter, H. 1992.** "The First CAAD Package (sketch based cad)". *Proceedings of CAAD Futures '91 in Zürich*. pp. 497-511. Braunschweig/Wiesbaden: Vieweg.
- Watson, A.; Crowley, A., et al., eds. 1994.** CIMSteel: The Logical Product Model (LPM) version 3.3, ISO TC184/SC4/WG3/N319.ISO TC184/SC4.
- Wilson, P.R. 1993.** "A view of STEP". *Geometric modeling for product realization* (ed. Wilson et al.). Amsterdam: North-Holland.
- Wittenoom, R. 1995.** http://yarrow.wt.com.au:80/~realize/step_aec/aec_spec.html