



**AALBORG UNIVERSITY**  
DENMARK

**Aalborg Universitet**

## **Building Models and Building Modelling**

Jørgensen, Kaj Asbjørn; Skauge, Jørn

*Publication date:*  
2008

*Document Version*  
Publisher's PDF, also known as Version of record

[Link to publication from Aalborg University](#)

*Citation for published version (APA):*  
Jørgensen, K. A., & Skauge, J. (2008). *Building Models and Building Modelling*. Department of Production, Aalborg University.

### **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 -

### **Take down policy**

If you believe that this document breaches copyright please contact us at [vbn@aub.aau.dk](mailto:vbn@aub.aau.dk) providing details, and we will remove access to the work immediately and investigate your claim.

Research report

**Building Models  
and  
Building Modelling**

*Kaj A. Jørgensen*  
*Aalborg University*

*Jørn Skauge*  
*Aarhus School of Architecture*

## Preface

This report has been developed based on studies made in a research project, where building modelling has been carried out in practice. In the project, two building models have been developed and the resulting models have been analysed and compared by use of a number of other software applications. Two building modelling CAD tools ArchiCAD and Architectural Desktop was selected from the beginning and used throughout the project. The two building models were created on the basis of drawings and descriptions from an existing Danish building construction project, "Sorthøjparken".

The objectives of the report are firstly to describe in depth the fundamentals of building modelling and building models, secondly to develop some general and specific guidelines for building modelling and thirdly to make comparison between the two CAD tools. By these objectives, it has not been the intention to analyse and address specific problems regarding current design and modelling methodologies.

In the relatively theoretical parts of the report, a number of proposals are developed about concepts, abstractions and modelling approaches. These proposals are based on general considerations about systems theory, system models and systems modelling. The presented guidelines for building modelling in practice are partly related to the proposed approaches and partly to the CAD tools, which were available in the first phases of the project.

Although building models can be seen in a wider context, the limitation of this report is to cover only core data about buildings – the construction components and spaces. Consequently, building modelling is only limited to this content. As a further delimitation, the modelling activities only relates to the early phases of building construction projects. In this report, building models are regarded as a united representation, which

obviously is ahead of time compared to the possibilities of current building modelling tools.

The two selected CAD software products have primarily been selected because they at the start of the project where used already by the authors and not because of any preliminary evaluation. Further, the decision not to use other CAD tools was arbitrary and not based on any kind of analysis.

The report aims primarily at people in building construction, who have already some knowledge about building design or building modelling. By reading the report, it is the author's wishes that building modelling is regarded as an approach, which is based on a solid theoretical foundation. It is further wished that the presented proposals are considered thoroughly argued and that personal modelling approaches can be developed with reference to these proposals. Finally, it is the aim that the report can be useful for students, who participate in building construction educations.

The report is divided into chapters and appendices. The chapters include the coherent presentation of building modelling and building models while the appendices partly support with theoretical issues, partly describe and compare the two selected modelling tools and partly describe the "Sorthøjparken" building models and various modelling issues about these. Examples and figures are created primarily by use of these building models.

The research project has gained support from many parties and persons. Among all, the Danish "Boligfonden Kuben" has offered a most valuable financial support. All support is appreciated.

May 2008

Kaj A. Jørgensen

Jørn Skauge

## Abstract

In the introductory chapter of the report, the primary concepts related to building models are described and some fundamental characteristics about computer based modelling are stated. Further, the differences between drawing software and building modelling software is presented. The essentials of computer-based building models are described and the primary potentials of model-based building design are indicated.

The next two chapters describe fundamental issues of building modelling and building models. Both chapters are based on the theories in appendix A, which are applied to building modelling and building models. It is underlined that modelling should be performed on multiple abstractions levels and in two dimensions, e.g. the modelling matrix. Based hereon, the primary building modelling phases are identified. Further, the basic characteristics of building models are described. Included is the clarification of object-oriented software and object-oriented models and it is stated that the concept object-based modelling gives a sufficient and better understanding. Finally, the image of the ideal building model regarded as one united model throughout the entire life time is described. This model is gradually extended by use of multiple modelling tools and data from the model is extracted and used by a variety of additional tools, e.g. visualisation, economic analysis and technical simulations.

The following chapter is considered the main chapter of the report. In this chapter, a framework for building modelling is developed for the two modelling phases: design modelling and detail modelling. The key initial modelling activity in the framework is termed integral modelling, a highly iterative and integrated design approach, where all design proposals are united and tested as a whole. The result of integral modelling is the first prime version of the building model. The following two modelling activities in the framework are identified as activities

that can be performed concurrently. This is essential for building modelling because spaces and construction are complementary to each other. Detailed descriptions of the contents of these modelling activities are included in the chapter.

Based on the building modelling framework, a set of general guidelines are presented. These guidelines are considered independent of the functionalities of currently available modelling tools. The guidelines cover e.g. modelling approaches, identification of model objects, subdivision of objects and other issues about detailing and specification of model objects.

The remaining two chapters include the application of the general guidelines to more specific guidelines, which can be followed by use of the currently available modelling tools. First, the characteristics of the selected two CAD tools are described and, subsequently, the specific guidelines are presented.

Appendix A, about systems, system models, and system modelling, forms the theoretical basis for the chapters with the theoretical content. The following appendices B-D include more specific characteristics about the two CAD tools ArchiCAD and Architectural Desktop and a comparison between the tools. In the remaining two appendices, the special modelling issues related to modelling of the two "Sorthøjparken" models are described and the resulting models are presented and evaluated.

# Contents

<b>Preface</b> .....	<b>2</b>	<b>5. Building Modelling – General Guidelines</b> .....	<b>29</b>
<b>Abstract</b> .....	<b>3</b>	5.1 Modelling Approach.....	29
<b>Contents</b> .....	<b>4</b>	5.2 Identification of Model Objects.....	30
<b>1. Introduction</b> .....	<b>7</b>	5.3 Subdivision of Model Objects .....	30
1.1 Computer-Based Building Modelling Tools .....	7	5.4 Horizontal Layers.....	31
1.2 Computer-Based Building Models.....	9	5.5 Horizontal Layers and Storeys .....	31
1.3 Model-Based Building Design .....	10	5.6 Other Detailing Guidelines.....	33
<b>2. Building Modelling Fundamentals</b> .....	<b>12</b>	5.7 Identification and Further Specification of Objects .....	33
2.1 System Modelling Applied to Building Modelling .....	12	5.8 Application of the General Guidelines .....	33
2.2 Primary Building Modelling Phases .....	14	<b>6. Characteristics of Current Building Modelling Tools..</b>	<b>35</b>
<b>3. Fundamentals of Building Models</b> .....	<b>17</b>	6.1 Building Model Organisation .....	35
3.1 Object-Oriented Models and Software .....	17	6.2 Templates and Library Object Types .....	36
3.2 Building Model Objects and Structures .....	18	6.3 Support for Add-On Modules .....	37
3.3 Creation of Building Model Objects.....	20	6.4 Support for Sketching.....	37
3.4 Utilisation of Building Models .....	21	6.5 Creation and Detailing of Model Objects .....	37
<b>4. Framework for Building Modelling</b> .....	<b>23</b>	6.6 Graphic Presentation of Building Models .....	39
4.1 Integral Modelling – Spaces .....	24	6.7 Data Extraction for Other Purposes.....	40
4.2 Integral Modelling – Construction .....	24	<b>7. Building Modelling – Specific Guidelines</b> .....	<b>41</b>
4.3 Integral Modelling – Performance .....	25	7.1 Limitations Regarding Modelling Approach .....	41
4.4 The Result of Integral Modelling .....	25	7.2 Creation of the First Model .....	41
4.5 Subsequent Design and Detail Modelling.....	26	7.3 Detailing of Objects.....	42
4.5.1 User Spaces.....	26	7.4 Visualisation and Model Publishing.....	42
4.5.2 Construction .....	26	7.5 Data Extraction .....	43
4.5.3 Objects Belonging to Both User Spaces and Construction .....	28	<b>Conclusion</b> .....	<b>44</b>
4.5.4 Model Evaluation, Consistency Checking, Constraint Specification, etc. ....	28	<b>Glossary</b> .....	<b>46</b>
4.5.5 Levels of Detail .....	28	<b>Literature</b> .....	<b>51</b>
		Web Sites .....	51

## **Appendix A – System, System Models and System Modelling ..... 52**

Analysis and Synthesis .....	52
Fundamental Abstraction Mechanisms .....	53
Generic Model Component .....	54
Identification of Multiple Abstraction Levels .....	55
References .....	57

## **Appendix B – Modelling with ArchiCAD (AC) ..... 58**

Line of action using AC .....	58
Space Modelling .....	58
Construction Modelling .....	59
Particular possibilities with AC .....	59
Tools to cut and isolate objects.....	59
Creating Reports from the Model.....	61
2D drawings .....	61
Particular issues .....	62
Cavity Closure.....	62
Special Designed Windows and Doors .....	63
Wall Ends. ....	63
Design regarding Cavity Closure and Wall Ends .....	64
Wall intersections .....	64
Wall Composites .....	64
Slab composites .....	64
Roof composites – flat roof.....	64
Free standing Windows and Doors.....	65
Development of individual Doors and Windows.....	65
Layout and Print of 2D drawings .....	65
IFC export / import in AC.....	65
Conclusion.....	65

## **Appendix C – Modelling with Architectural Desktop (ADT) ..... 67**

Line of action using ADT .....	67
Space Modelling .....	68
The wall components.....	71
Free standing Windows and Doors.....	71

Endcaps .....	72
Creating reports from the model.....	72
2D Drawings.....	72
Particular issues.....	73
The slab components .....	73
The Roof Components – Flat Roof .....	73
Corner windows .....	73
Stairs .....	74
Layout and Print of Drawings .....	74
IFC export and import in ADT.....	74
Conclusion .....	74

## **Appendix D – Comparison between ADT and AC ..... 76**

Modelling .....	76
Parametric objects.....	76
Stories.....	76
Analysis and editing during modelling.....	76
Modelling Support .....	77
Modelling tools.....	78
Reports.....	80
IFC .....	81
Conclusion .....	82

## **Appendix E – Modelling Sorthoj ..... 83**

Focus was based on the following issues.....	83
The modelling process .....	83
Modelling phases.....	84
Learnings from 3D modelling .....	85
Modelling the exterior walls .....	86
Roof in relation to storey / level.....	86
Roof construction .....	86
Corner windows .....	86
Generic objects .....	86
Detailing of walls, slabs and roofs.....	87
Product specific objects .....	87
Detailed Roof construction .....	87
Conclusion .....	87
Literature.....	88

<b>Appendix F – Sorthøjparken - Presentation and Evaluation .....</b>	<b>89</b>
Reports of the Product Data Model .....	89
Visualization and animations .....	89
Visualization in AC .....	89
Internal rendering engine in AC .....	89
Lightworks rendering engine .....	90
Rendering in 3DS format .....	91
Visualization in ADT .....	91
Rendering in VIZ. ....	91
Virtual Reality .....	92
TurnTool VR.....	92
VR4Max VR.....	93
Quantity calculations.....	94
Construction analysis in ADT2005 .....	95
Static calculations of the balconies .....	96
Adjustments .....	96
Loads.....	97
Construction .....	97
Numbering of elements .....	97
Beams and frames .....	98
Thermal analysis .....	98
The DDS Viewer for IFC model analysis .....	100
Using the DDS Viewer for 3DS export .....	100
Solibri Model Checker .....	100
Reports – 2D working drawings. ....	101
Conclusion.....	102

# 1. Introduction

Building design is an important part of building construction. In design activities, primary decisions are made about the building that is meant to be constructed. The result is data that describe the building as it should be constructed and to some extent also describe how the construction activities should be performed. The resulting description should be as precise as possible in order to avoid misinterpretations and errors during the construction tasks.

A description of a building is generally termed a *model* and, traditionally, such a model most often consists of a set of drawings and a verbal description. However, it is important to remember that a building model can be of different nature, can be expressed in different ways and can contain different selections of data about the building. In addition, during the last decades, computer technologies have added new dimensions to the ways, how building models can be developed and presented. Today, many sorts of appropriate software tools are available for this and the production of drawings is predominantly performed by use of computers. This means that the building models are created by computers and the physical drawings and descriptions are generated as output from computers. Furthermore, building models are somehow represented internally in computers and stored in computer memories so that it is relatively easy to operate, present, exchange and transform models. *In the following, only computer-based building models<sup>1</sup> will be considered.*

## 1.1 Computer-Based Building Modelling Tools

The primary type of software that is currently used in building design is *drawing software*, where the primary ability of the software is to assist the users with creating and handling of different kinds of lines and text on ordinary two-dimensional drawings. The internal representation of such drawings is only a data structure of objects representing drawing components and with attributes defining e.g. coordinates, line type, line weight and line colour [See 2007]. However, drawing software is becoming more advanced and, in addition, new sorts of building modelling software are now available.

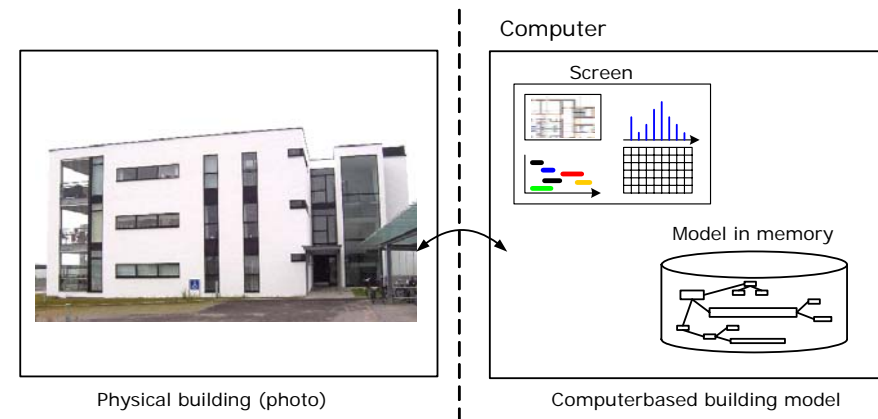


Figure 1 – Building modelling software represent building components as software objects

Characteristically for modern *building modelling software*, the internal representation of the building model is fundamentally

<sup>1</sup> The term *building model* is used in this report in contrast to other often used terms like *Building Information Model* or *Virtual Building*.



different. In contrast to drawing software, the components<sup>2</sup> of buildings are modelled as software objects in the computer memory e.g. walls of buildings are modelled by wall objects in and doors are modelled as door objects [Froese 2002] [See 2007]. In general, such *model objects* are the basic substance of computer based building models.

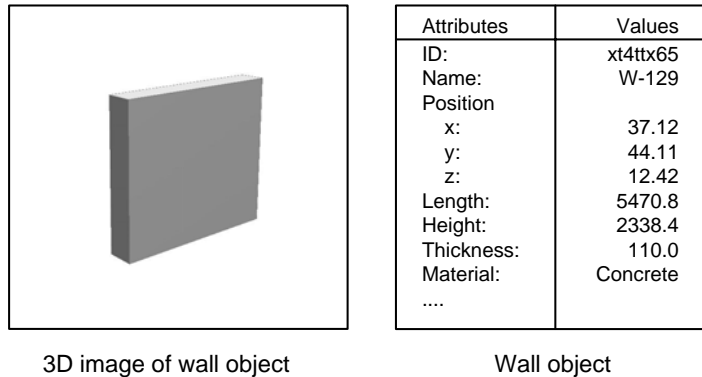


Figure 2 – Attributes of a wall object

Model objects contain a set of *attributes* for specification of the building components (see Figure 2) and *relationships* between components of the building can also be represented in the software, i.e. as structures with objects (see Figure 3). When an object is created, values are given to the attributes either by default or by specification via the software. Normally, these values can later be changed as required and, therefore, objects can also be characterised as *parametric objects*.

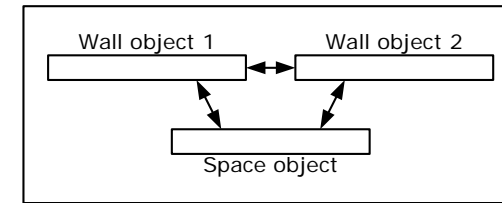


Figure 3 – Relationships between objects

This way of representing building components is also fundamentally different compared to many simple free-form drawing tools, in which primarily visible surfaces of building components are represented. The content of the building components is not modelled. In contrast, building model objects makes it possible to produce building models that can be enriched with various specifications. Thereby, it is possible to extract traditional data like drawings from the building model as well as other types of data extractions, e.g. quantity take off, economic calculations, visualisations, heating and ventilation analysis, acoustic analysis, estimated activity plans for construction, procurement plans and logistical requirements (see Figure 4). Such building models may also take input, which can be useful at later stages, e.g. for facility management.

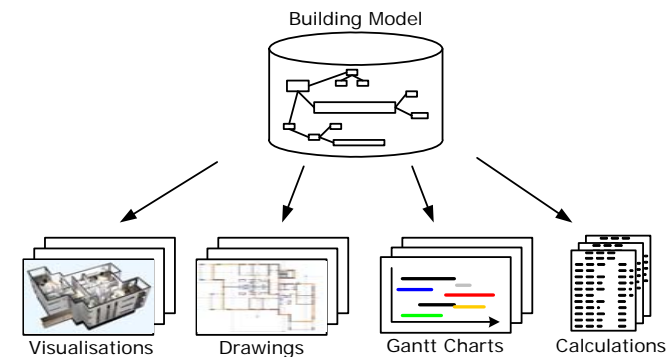


Figure 4 – Various forms of data extraction from building model

<sup>2</sup> The term component is used as the general term for all parts of a building in contrast to another often used term *building element*. See also section 3.2.

Software for these issues is most often available as separate tools. Hence, it is often appropriate to distinguish clearly between modelling software and software for the various kinds of data extraction from building models. Thereby a wide range of additional applications could be developed for specific purposes. Such applications could focus on explicit mapping of data from building models. For instance, the specifications of line weight and colours etc. could be mapping data for a drawing application instead of data attached to the model objects.

## 1.2 Computer-Based Building Models

Ideally, a building model should contain data produced in the entire *lifetime* of the building from the time, where the idea of the building is born, through the construction period and the utilisation period to the time of removal of the building and even after that. In this scenario, the model is supplied and extended with data through the design activities, the construction activities, the utilisation activities, etc. The contained data should be carefully maintained so that they can be reused as much as possible in all life phases.

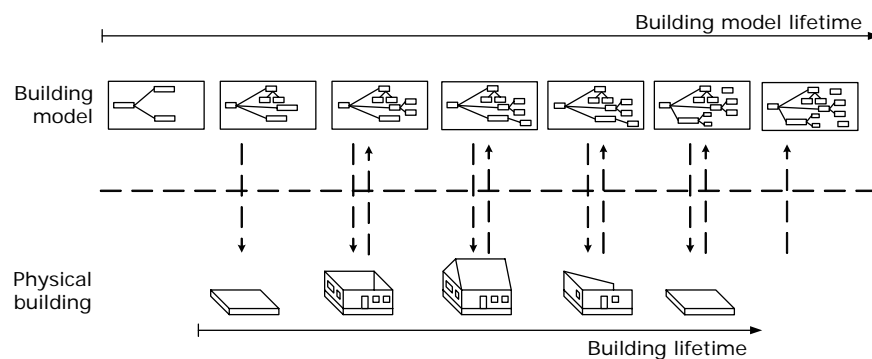


Figure 5 – Building model life time and the life time of the physical building

Related to this ideal view, it is important to look at representation of the building model. Predominantly, each software vendor has developed their own internal data representation and file formats and the choice of attributes in the data objects varies much in the software tools. Therefore, it is difficult to exchange building models between software tools and, when it can be done, it normally leads to loss of data.

In order to provide a common data representation and thereby enable easy exchange of model data, international standardisation work is carried out by the *International Association for Interoperability (IAI)*, who has published the neutral data model *Industry Foundation Classes (IFC)*. This model is very comprehensive and is until now the best attempt made to provide support for the idea of collecting all data of a building model in a united representation.

IFC concentrate on representation of the core data about building components as model objects and independently of the modelling applications. So, data about how the building is presented is secondary, e.g. surface colours, line weight and line colours.

In addition, the IFC standard includes also specifications of how model data can be represented in data files<sup>3</sup>. Thereby, software vendors can develop interfaces, which can read and write files, where building models are represented by IFC.

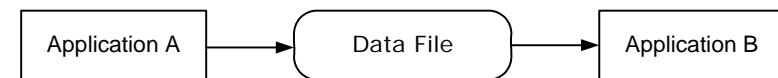


Figure 6 – Data exchange based on file

<sup>3</sup> Originally, the ISO 10303 Part 21 file format was developed and specified but lately a file format based on XML has also been created. The Part 21 file format is usually referred to as the IFC file format.

A few software vendors have implemented the complete IFC data model as a *database*, which can then accept and store all data objects that comply with IFC. A computer system with such a database is termed an *IFC model server*. Such servers obviously provide means for model input and output but it is important that some special services for extraction, versioning, merging and concurrency are also offered. Ideally, applications should be able to exchange data directly with model servers.

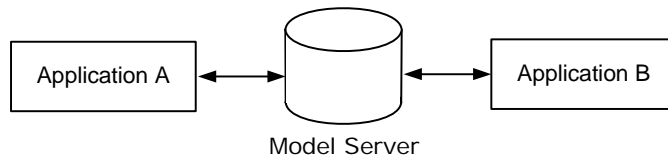


Figure 7 – Data exchange with model server

Hence, the current situation is that a comprehensive international standard for representing building models is available and corresponding model servers can store the models. But, when it comes to software tools, there are serious limitations. Some tools can import IFC formatted files, some can export to IFC formatted files and some can both. However, the way this is done differs quite a lot. Because each tool has its own internal representation, it is sometimes impossible to perform exact transformations and, in such cases, all data are not exported or imported.

Therefore, the image of the building model as one single file or database accessible from a range of interoperable software tools may still seem rather ideal. But, with a few more functionalities in model servers and some better import/export/merge features in key software tools, the situation can be improved a great deal.

### 1.3 Model-Based Building Design

Traditional projects, which are based on the use of drawing software, have often demonstrated considerable loss of resources. It is difficult to control the production of drawings, including version management and distribution of new versions of drawings. There are many examples showing that misinterpretations and use of obsolete drawing versions have had serious consequences. Consequently, it is a well known fact that the cost of making changes to already made decisions increases dramatically as the design activities progresses (see Figure 8).

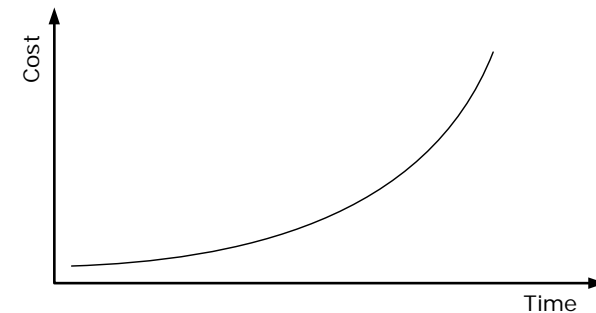
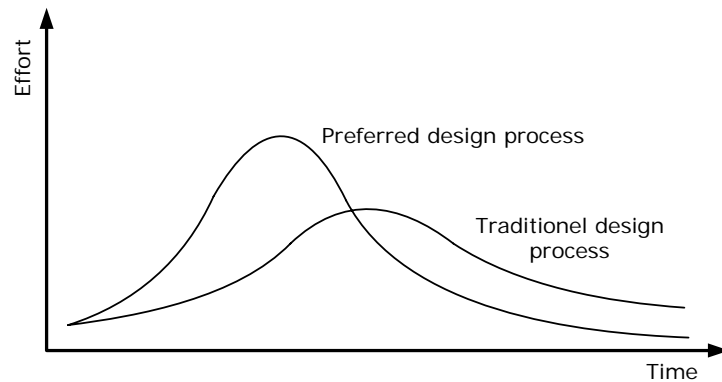


Figure 8 – Cost of making changes increases over time

In addition, there is a general tendency to delay decision making in design activities so that the design effort is spread relatively much over time (see Figure 9). Many decisions are often postponed until it is absolutely necessary for other certain tasks to be performed. Furthermore, design tasks are normally carried out in stages, which are prepared according to approvals from the authorities. However, considering Figure 8, the emphasis should be concentrated on the first stages of a design process and how to move the right decisions towards these early phases (see Figure 9) so that unnecessary waste of resources can be avoided.



*Figure 9 – Design effort variation of traditional design process and preferred variation for building modelling*

Modelling makes it possible to distribute the design effort differently so that efforts can be shifted to other more important issues. Model based design can be seen as an important response to the desire for better communication during the design process and can make it possible to get a precise impression of the resulting product. Before making decisions about the details, it is possible from models to e.g. simulate reality, calculate cost estimates, visualise and estimate functionality etc. Furthermore, it is easier to produce alternative proposals and, finally, production of drawings can to greater extent be performed automatically.

However, because modelling tools are relatively easy to use, they also encourage to postponement. Therefore, it is important to develop good modelling methodologies, to make careful implementation of the methodologies and to control that they are followed.

## 2. Building Modelling Fundamentals

Modelling is typically characterised as an iterative process with development of proposals and subsequent testing against the specified requirements. In the beginning, the proposals are made as sketches and analysis is made rather roughly. As the proposals become more concrete and detailed, testing should be performed as more exact analysis and here new opportunities with computer-based modelling tools have become available. Such tools have become more useful and with an increasing number of functionalities.

Often, the modelling tools dictate certain modelling methodologies with a number of limitations. However, modelling can be performed in many ways and can have different meanings. The emphasis can be set on many subjects, decisions can be sequenced in many ways and resources can be allocated variously.

In appendix A, a short description of key concepts and issues about systems and system modelling is given as a basis for the following sections. These fundamental issues are applied to building modelling and addressed more specifically. Especially, it is clarified, how abstraction can be used to manage the complexity of modelling and what is included in building modelling on multiple levels of abstraction.

### 2.1 System Modelling Applied to Building Modelling

As stated about system modelling, it is important to distinguish between *analytic modelling* and *synthetic modelling* and thereby between *analytic models* and *synthetic models*. Building modelling, as it is introduced previously, is therefore regarded as synthetic modelling and the result is a synthetic model, which serves as a foundation for the construction work.

However, analytic modelling is often performed in order to establish an important basis for developing such models.

As also stated in appendix A, models can be characterised by the degree of abstraction with physical as the lowest degree of abstraction and mental as the highest degree of abstraction. Rooms, for example, are in building modelling often not identified before surrounding walls are created, a relative physical view. Mentally, rooms can be identified long before their spatial positions and dimensions are determined.

Modelling is often rather limited with respect to abstraction level and modelling approach, i.e. the process is regarded as relatively analytic and relative physical oriented (illustrated in Figure 10). A similar characterisation can be stated for models. It is, therefore, important to consider a suitable balance, i.e. a suitable level of abstraction and a suitable approach.

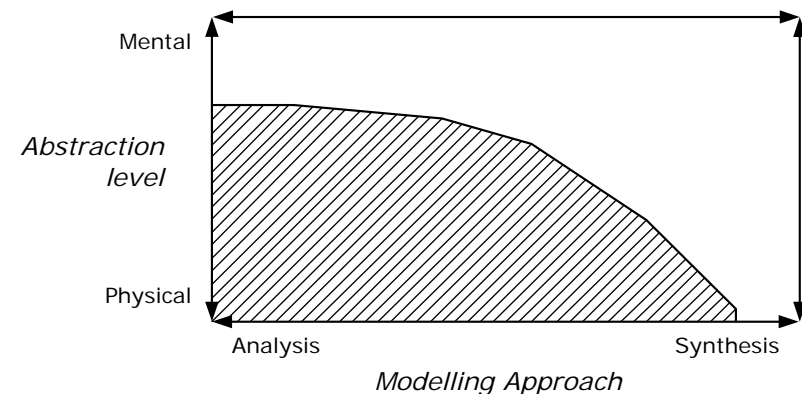


Figure 10 – Typical characterisation of modelling

Two abstraction mechanisms are also described in appendix A; they are termed *composition* and *classification*. Composition is shortly described as the development of hierarchies of building

components determined by *whole-part relationships*. Classification involves development of hierarchies of *classes*, termed *taxonomies* showing *general-special relationships*.

In connection with building modelling, these abstraction mechanisms are used very often. The well known relationships that building components can be divided into smaller components, sub-components and that a set of components can be assembled to a new component, super-component are simple examples of composition. Buildings of various types have many similarities regarding their composition, compared to modelling of other kinds of products, e.g. walls generally consist of material layers and have openings with windows and doors. Consequently, it is possible to create a number of basic composition structures of general value for the building sector.

Classification and development of general taxonomies for the building sector are of even greater value. Such taxonomies can support the identification of model components. Whenever a possible solution must be considered, it is often suitable to have a set of relevant taxonomies available. Thereby, specific solutions can be selected in a systematic way. Simple forms of taxonomies are catalogues with types of building components classified by certain criteria, e.g. bricks by colour, windows by form, walls by composites, tiles by dimension, rafters by form and concrete by strength.

Like for systems modelling in general (see appendix A), the identification of multiple abstraction levels for synthetic building modelling contains considerations about the two corresponding dimensions: *modelling of attributes* and *modelling of structure*. The combination of these two dimensions forms the *modelling matrix*. Modelling of attributes means gradually identification, definition and specification of attributes of model components. Modelling of the building structure include identification of sub-components and their mutual relationships. For each sub-component, both modelling of attributes and modelling of sub-

components are applied recursively. How deep, it is necessary to go in the structure, differs from project to project.

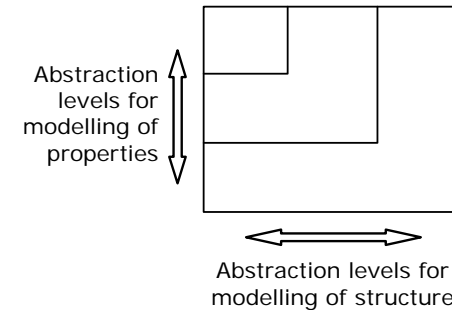


Figure 11 Modelling matrix: modelling of attributes and modelling of structure

All modelling activities start in the upper left corner of Figure 11 and they are supposed to end in the lower right corner. In the beginning, the work is only performed on one component, the model component representing the entire building, and the set of attributes belonging hereto. At the end of the projects, all components have been created and the attributes of these components have been specified.

Which specific route to follow through the matrix and how fast it is done depends on circumstances and will typically differ from project to project. In some projects, it is important to allocate more work on a high level of abstraction before the details are defined. But in other cases, it is possible to go faster to identification of details.

Thus, it is important to consider, which route to follow through the matrix. Modelling of attributes of the entire building as well as of sub-components regards the ability to perform functions. In synthetic modelling, this mostly includes working with appearance attributes, which characterises and represent each function. These attributes are also termed performance attributes because they represent the performance of the

building. To illustrate two different routes through the modelling matrix, one approach would be to work separately with the performance attributes in order to set requirements for the building's performance before specific solutions are found. Another approach could be to seek alternative solutions and estimate the performance.

For buildings, it is important to realise that the primary components are two kinds of *spaces*: the *user spaces*, which are utilised by the users, and the *construction spaces*. Hence, these two kinds of spaces are complementary to each other (see Figure 12). User spaces are divided into *exterior spaces* and *interior space*. The building construction spaces contain the building *construction components* and, typically, they separate the interior user spaces from each other and act as closure of the building.

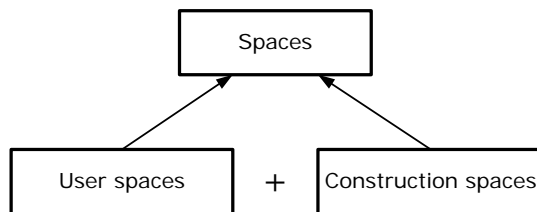


Figure 12 – User spaces and construction spaces are complementary to each other

Composition structures of spaces and construction components are defined by aggregation and separation. As result of aggregation, larger spaces and construction components can be identified and, as result of separation, minor components can also be identified. For instance, a hall is defined as a space consisting of a number of sub-spaces and a roof is defined as a collection of many individual building components. In another kind of composition, a space can be subdivided into a mixture of user spaces and construction spaces, e.g. a space with free-standing walls.

## 2.2 Primary Building Modelling Phases

As indicated above, modelling on higher abstraction levels is typically characterised as work on data about the building before the physical substance of the building is identified. Therefore, when initial considerations about creating a building takes place, then, above all, the purpose of the building must be expressed and it must be determined, what the primary functions of the building should be. This include overall long-term considerations about how the building should function in its future environment with users, owners, authorities, landscape, etc. (see Figure 13) [Kiviniemi 2005]. Examples of basic functions of a building are to provide spaces, to shelter from the weather, to provide heating, to dispose waste water and to secure property.

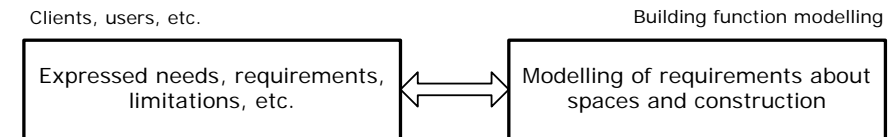


Figure 13 – Requirements and building function modelling

This kind of modelling primarily uses the function modelling approach and is termed *function modelling*. It is important to perform this modelling as a reliable foundation for the subsequent modelling activities so, based on these overall requirements, secondary functional requirement can be identified. If possible, high level function modelling and simulations should be performed in order to balance contradicting requirements. For instance, total economy calculations are often performed on multiple abstraction levels and thereby on different grounds.

When results of function modelling are obtained, modelling is more oriented towards possible solutions. The second primary kind of building modelling is termed *design modelling*. Ideas of solutions about user spaces and construction components are

generated and tested against the requirements, i.e. the function model. The building is designed by modelling to a suitable level of detail. It is characteristic for design modelling that all primary structural decisions are made so that only detailing with minor and limited structural implications remains to be decided. It is important to consider the economic relationships illustrated by Figure 8 and Figure 9.

In order to make durable decisions, i.e. to avoid change of decisions, it is important that these modelling activities are iterative and take all requirements into consideration. Therefore, it is important that analysis methods are available on different levels of abstraction.

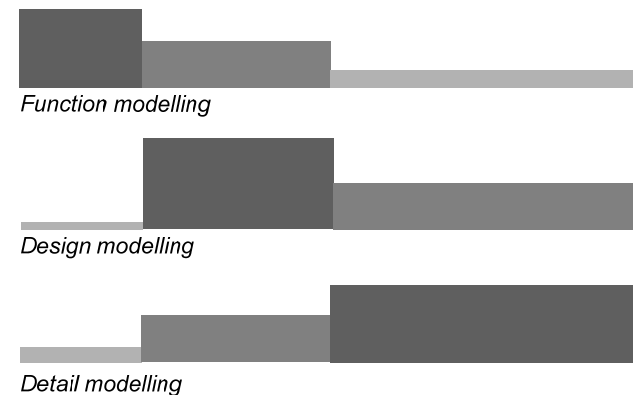
A third kind of modelling is *detail modelling*, where the building model is detailed down to the required level. Of course, this modelling is based on the results from design modelling and, as stated, it should not affect the primary structural decisions, which are already made. In all modelling projects, it is always necessary to consider how far detail modelling should be performed. It depends on the modelling purpose and what the building model should result in, i.e. what it should be used for.

Consequently, there are three primary kinds of modelling: function modelling, design modelling and detail modelling. The three kinds of modelling are different and, in all modelling projects, they must be combined and a suitable balance between them must be considered, when resources are allocated to the included activities.

It must be underlined that these three modelling approaches must also be seen in combination with the two modelling dimensions in Figure 11. For instance, each time a new model component is to be created, primary attributes, e.g. geometry attributes, are generated and their values must be specified. In addition, the structure of the component must be considered.

In the modelling period of the project, the work is spread over the total period but, typically, the work load is unevenly distributed as illustrated by Figure 14.

#### Primary kinds of modelling



#### Modelling phases:

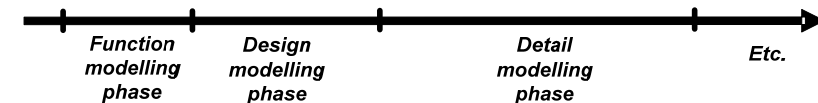


Figure 14 – Identification of modelling phases

Hence, three main *modelling phases* are also identified: *the function modelling phase*, *the design modelling phase* and *the detail modelling phase*. As illustrated in the figure, these phases can not be precisely delimited as the modelling activities overlap each other. Further, it must be emphasised that the modelling phases to some extent can become iterative.

Ideally, modelling tools should be available to support all phases or it should be easy to combine the use of different tools. Typically for building CAD tools, they are mainly focused on the construction and from a rather physical viewpoint (see Figure 10). Modelling of overall functional requirements is



rarely possible in these tools and the features for space modelling are often rather limited.

*In the following, only the design modelling and detail modelling phases will be considered.* Consequently, it is assumed that the function modelling phase has been carried out and the functional requirements about spaces and construction are identified and specified.

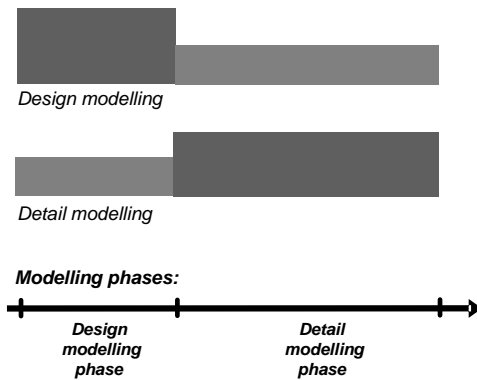


Figure 15 – The following covers only design modelling and detail modelling

*Further, the subsequent phases in the building lifetime will only be considered to the extent that it is briefly shown how building models can be utilised.*

### 3. Fundamentals of Building Models

Computer-based models are stored in computer memories as *data structures* (illustrated in Figure 16). Data structures consist of *data objects* with *data attributes* and *references* to other data objects. The data object references are used to implement the previously introduced relationships among objects (see chapter 1). The data structures are permanently stored in a persistent memory, e.g. disk storage, and accessed and manipulated by software applications, separated from the data structures. When such applications work with the data structures, temporary data structures are built in working memory.

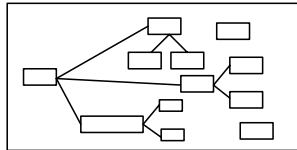


Figure 16 – Structure with data objects and references

Modelling tools can be based on an already developed data structure, e.g. IFC, or they can work on their own proprietary data structures.

The specification of a data structure is termed a *data model*, a synthetic model with identification, definition and specification of object types. An *object type* is a description from which individual objects can be generated (see Appendix A, figure 4 and 5). Each object type has a description of a set of attributes and a set of *relations*, i.e. rules or constraints, which specify how data structures can be created. One of the most important requirements for a data model is that it is non-redundant so that no data value is stored more than once. In order to ensure that this requirement is fulfilled, the model has to be developed with identification of the data and the meaning of data, the semantics. Information is data and the meaning of data (see Figure 17). Therefore, the foundation for a data model is an

*information model*, created with semantics from the domain, which the model is addressing.

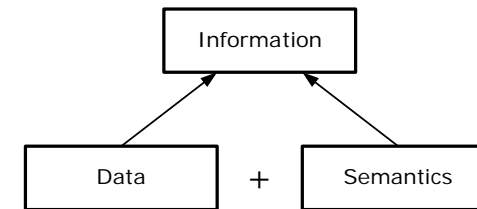


Figure 17 – Information = data + semantics

#### 3.1 Object-Oriented Models and Software

Modern software is often characterised as *object-oriented*, which means that, besides the data attributes, objects also have behavioural attributes, i.e. *methods*, which can perform operations on the data attributes. For illustration, see the generic model component in Appendix A, Figure 3. Thereby, such objects can better resemble real life living organisms<sup>4</sup>. This view is underlined by the conception that objects can send *messages* to each other.

Hence, if a computer-based building model is regarded as object-oriented, the model objects also include behavioural attributes (methods). An object, which should represent a door in a building, will include method attributes, which e.g. can take care of changes to its measurements and control that certain constraints are obeyed. To enable interaction with wall objects, other method attributes can be included. For instance, when a door is inserted in a wall, the corresponding two model objects

<sup>4</sup> Because of this, objects are sometimes considered intelligent. It must be remembered, that this is purely *artificial intelligence*.

are related to each other and some derived operations may be carried out automatically, e.g. in response to cutting a hole.

As indicated, the concept object-oriented is primarily related to information models and implementation of these models in software. For most modelling considerations, it is not necessary to imagine objects with methods. Therefore, in the following, the concept *object-based* will be used as characterisation of software and building models.

### 3.2 Building Model Objects and Structures

As previously stated, spaces and construction components are represented as model objects in building models. Besides these objects, building models need also to include objects to represent various structures. As also mentioned, one fundamental kind of structure is the hierarchy structure. Spaces can be organised in hierarchies and, similarly, the construction components can be structured by forming hierarchies.

In a composition hierarchy, each component can be considered a part of a larger component (super-component) and each component can be divided into smaller components (sub-components). The top level building component is the total building, which of course is not a part of a super-component. The components at lowest levels of the hierarchy are the components, which are not considered subdivided. They are also termed *elements*<sup>5</sup>. This delimitation of elements is project dependent and for instance varies with the degree of prefabrication. For instance, if a wall is prefabricated, this component may be regarded as an element and no subdivision is carried out.

<sup>5</sup> The term element is used in many modelling tools and also in IFC. This indicates clearly a bottom-up approach, which is typical for current modelling tools. As previously stated, the term component is used in this report and thereby it is emphasised that modelling has to be considered both top-down and bottom-up.

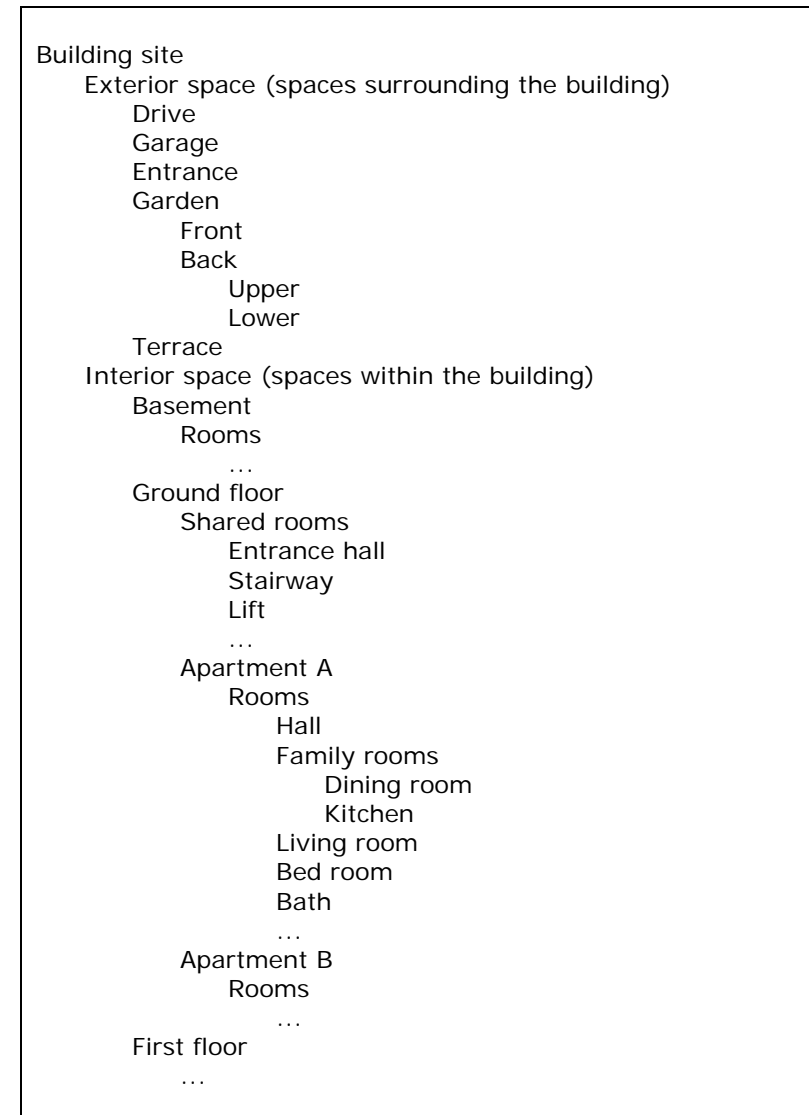


Figure 18 – The outline of a hierarchy of user spaces seen from a functional viewpoint

As construction components can also be regarded as spaces, all spaces can be gathered in one single hierarchy but, in practice, it is more appropriate to work with two hierarchies; one for the user spaces and one for the construction components including the technical installations.

A hierarchy of spaces shows how spaces can be sub divided versus aggregated. By creating a tree graph of the structure (see the example in Figure 18), the hierarchy offers a good overview. Among the user space hierarchy, it is important to identify all the rooms of the building. Normally, they are the primary spaces, seen from a functional viewpoint.

Every building construction can also be sub-divided and form a hierarchical structure of construction components (see the example in Figure 19). In principle, this hierarchy can be detailed down to the smallest component of the building and, in general, the depth of the hierarchy is increased along with on the progress of the modelling process.

Construction components can also be aggregated into larger components, for instance, roof construction, facade, foundation, and technical installations. When this is performed, different structuring criterions can be applied. For instance, the components can also be structured by location, building sections and storeys.

Most appropriate, the construction component hierarchy is structured purely from a building product point of view, i.e. walls, columns, beams, windows, pillows, etc. and in many modelling projects, model objects of these types are considered the basic content of building models.

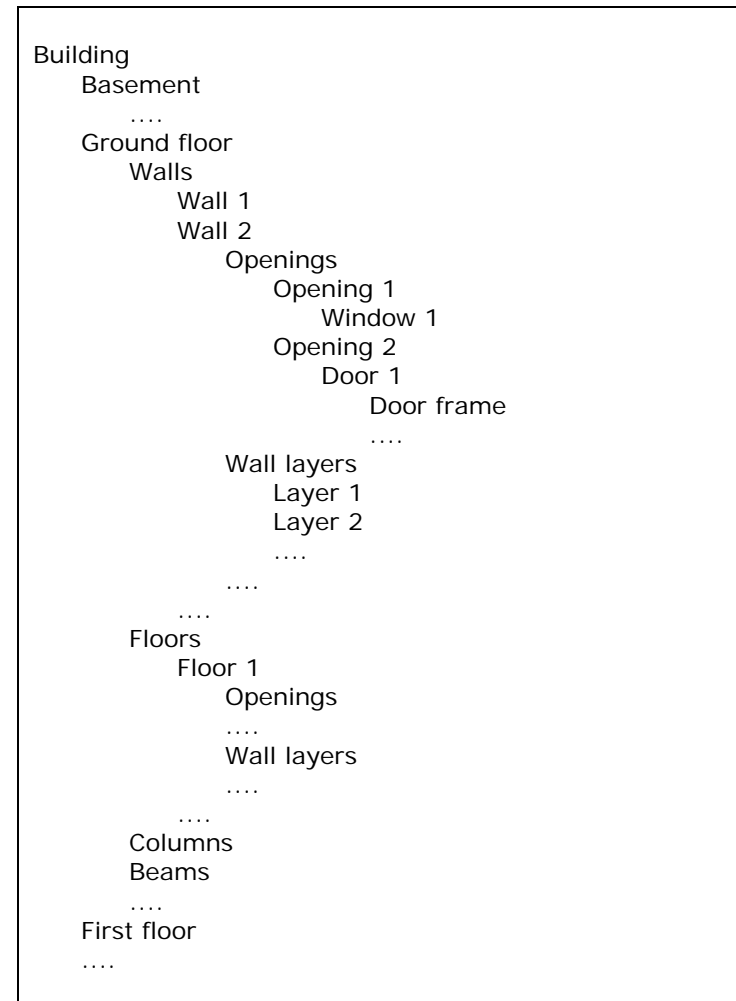


Figure 19 – The outline of a hierarchy of building components

Besides the hierarchical structures mentioned above, components can interrelate in other ways. The most obvious *relationships* are the physical relationships that occur, when components are in physical contact with each other, e.g. a relationship is created between two wall objects if they are supposed to be joined together physically. It is also suitable to create relationships between the space hierarchy and the component hierarchy. Most important are the relationships between rooms and building components.

Furthermore, model objects may be related to each other in collections in a more general and flexible way compared with the super-sub hierarchical relationship. A *collection* establishes a direct access to the members of the collection. For instance, it may be suitable to go across the building model and collect window objects of the same kind. Collections can be defined before or after the objects are created. In the first place, the object can be included in one or more collections when it is created. It is important that modelling tools support creation of collections.

### **3.3 Creation of Building Model Objects**

With object-based building modelling tools, the model objects are identified and defined very early in the process in order to enable specification of attributes and to create the various kinds of relationships. Typically, the modelling tools focus on *geometry* and the model objects are created as soon as the primary geometric data are specified. Modern modelling tools work with *three-dimensional (3D) geometric representation* and all views of the objects on computer screens or on printed drawings are projections of 3D to 2D. Further, this means that all model objects are located in space.

One of the advantages of such modelling tools is that creation of objects can be made very easily because various libraries are available with the tools. The libraries contain types of objects,

which can be selected and inserted in the model. Typically for CAD tools, this can be performed visually by drag and drop operations in the screen interface.<sup>6</sup>

As previously mentioned, building model objects contain *data attributes*, which contribute to a description of the building. These attributes are very important for analysis and simulation purposes. Some analyses can be performed when only the 3D representation is available but normally some additional attributes need to be specified, for instance different kinds of material data for analysis of strength, energy or acoustics.

As also mentioned, a number of proposals have been presented about how building model objects should be represented and many modelling tools have developed their own representation, i.e. data model. Today, the international standard IFC should be considered as a reference not only for building model exchange but also for model representation.

IFC is a data model, which is developed over a long period by the international organisation IAI. The primary content of the data model is a large set of object types, which can be used to describe components of buildings and many other related topics, e.g. activities, resources, and actors. Each object type consists of a fixed set of attributes but an unlimited number of additional attributes can be attached by using, what is termed property sets.

As mentioned previously, this standard is very comprehensive and is prepared to enable representation of a large variety of buildings and building components. Until now, all software tools have only implemented a limited part of the data model. Some tools can read IFC files and will typically extract only the data that are relevant for the tool. The modelling tools naturally set

---

<sup>6</sup> Modelling with this type of CAD tools is often termed *3D modelling* but, as indicated, this modelling approach is only one side of building modelling.

most emphasis on how their internal representation can be transformed to IFC and only secondary on the reverse transformation.

IFC supports the notion of space hierarchies and construction component hierarchies as introduced above. In principle, this means that spaces and construction components may be created with other approaches than with CAD tools, where the geometry is the primary identification.

IFC also supports the idea of the building model as one united model, which is extended and maintained throughout the entire modelling phase as illustrated in Figure 5 and Figure 20.



Figure 20 – One united building model throughout the entire modelling phase

In the present chapter and in the following two chapters, this ideal view of building models is considered as the basis for building modelling. Consequently, the description of modelling and the discussions about modelling methodologies and approaches are based on this ideal view and thereby relatively ahead of what is currently possible.

### 3.4 Utilisation of Building Models

At any time, the building model can be used as the basis for model extractions with presentation, visualisation, analysis and simulation as illustrated in Figure 21. Some of these operations are available in the modelling tools but usually separate applications must be used.

Model extractions are determined by purpose and performed by setting up specific selections from the building model. This means that different parts of the model are selected with the aim to simplify the view. Furthermore, the selected data can be aggregated to reach additional degree of simplification or added to the view as enrichment. Consequently, such extractions are considered *sub-models of the building model*.

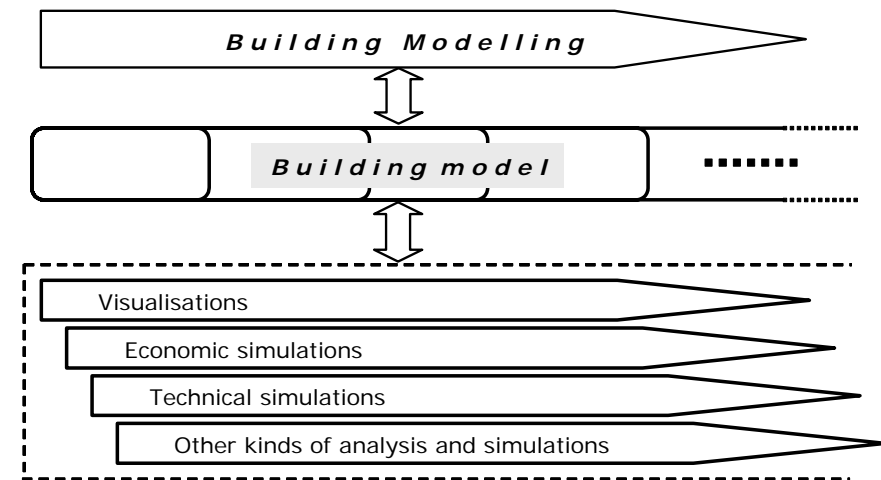


Figure 21 – Model extractions are performed concurrent to building modelling

Models and model extractions (sub-models) can be presented in many different forms depending of the purpose. Thereby, the image of the building model can be very different to the viewer. Of cause, the value of the presentation depends on the content of the building model.

Often used presentation forms are:

Augmented reality  
 Virtual reality  
 Planned flight/tour  
 Picture  
 Scale model – 3D print  
 Graphs  
 Hierarchies  
 Tables  
 Lists

Graphical view forms are well known from drawings, where various 2D projections are produced. But many 3D graphs can also be generated. If for instance the presentation should underline the primary architecture of the building, a graphical view as shown in Figure 22 could be produced.

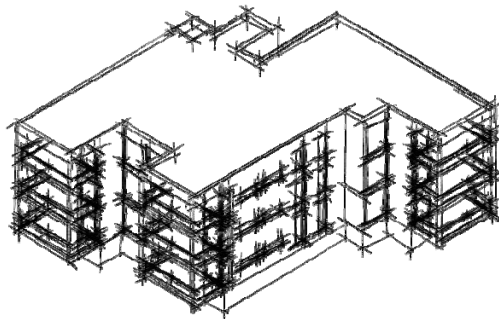


Figure 22– A view of the model, which focus on the primary architecture

Other examples of extractions and selected presentation forms:

<i>Data extraction</i>	<i>Presentation form</i>
Spaces	hierarchy, augmented reality
Rooms	3D boxes, 2D areas
Escape routes	graph on floor plan
Tensions (slabs, beams, etc.)	coloured graph
Electric wire placements	3D "X-ray" graph
Quantities	table with summations
Cost calculations	table, histogram, pie charts
Interior	rendered picture
Bearing construction	beams and columns as lines
Composite layers of walls	2D cross section

Figure 23– Examples of data extractions and presentation forms

As underlined, sub-models presented in various forms are extractions of the building model. Such selections may be formulated as mapping rules and stored independent of the model. For each view, multiple mapping rules could be applied in order to generate views, which are dependent on different levels of detail of the building model.

Stand-alone applications are typically related to investigation of specific primary building functions or system views, e.g. structural analysis of the bearing system, thermal analysis of the building including the heating and ventilation system. In principle, it is important that the data flow is bi-directional so that results from these applications can be feed back to the building model.

## 4. Framework for Building Modelling

Ideally, building modelling should cover the whole lifetime of buildings. However, as previously stated, the following sections cover only building modelling in the design phase and the detail modelling phase. This means that, in principle, all requirements about functions of the building have been modelled and are available as specification of threshold values of the building's performance. Therefore, the following modelling must obey these requirements.

As also stated, design modelling should cover space modelling as well as modelling of the construction components. Design modelling is also characterised as an iterative process with development of proposals and subsequent analysis and testing against the specified requirements.

As a key initial activity of design modelling, it is important to work in a highly iterative and integrated approach with the objective to produce the first model version [Fällman 2003]. In the modelling framework, this activity is termed *integral modelling*, indicating that all requirements are considered in an integrated design process (see Figure 24). This means that all proposals are united and tested as a whole. In integral modelling the focus can be different but, even with a primary focus, secondary areas must also be considered. For instance, if the primary focus is set on modelling of user space the building construction must fit with the space. In contrast, if the building architecture is of primary interest, the user spaces may be considered secondary but the space requirements must be fulfilled as well.

It is characteristic for integral modelling that the use of resources is optimised while a consistent first building model is developed (see Figure 8 and Figure 9). In this model, it must be proved with reasonable probability that the functional requirements can be fulfilled throughout the subsequent

modelling. Primary decisions are taken but many concrete solutions are not yet found. In this phase, where the building model is relatively rough, it is important that some analysis applications are able to work on this basis. For instance, with 3D modelling tools, it is often possible to produce visualisations at any step in the model development. Thereby, performances regarding aesthetics, accessibility, sun lighting, price estimations, etc. can be shown. However, many other forms of data extractions may also be performed.

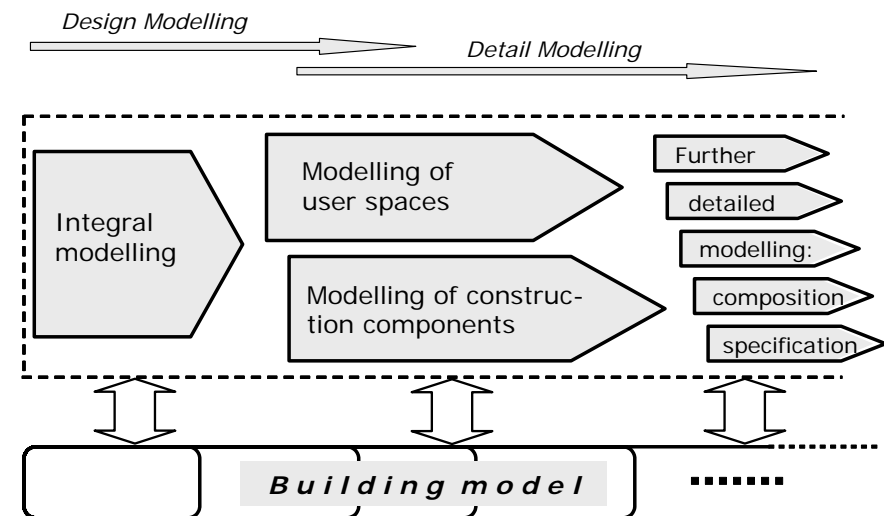


Figure 24 – Key building modelling activities

In the resulting model, the generated solutions are balanced against each other. It is often found that some requirements are contradicting but even with balanced requirements it may be difficult to find solutions, which suits these requirements in a balanced way. So, in this phase, it is important to seek balanced solutions. Consequently, the model must be evaluated



and approved by the building client. Thereby, some unnecessary iterations of the integral modelling can be avoided.

After integral modelling, where the primary construction components and the spaces have been created, the primary emphasis is put on further design and efforts for bringing the model to higher degree of detail so that for instance the analysis tools can work more precisely and more tools can become useful. Characteristically for building modelling, two primary modelling activities can be identified to cover the two complementary spaces user spaces and construction spaces (see Figure 12). Because these spaces are complementary to each other, it is further characteristic that the two activities are only loosely coupled and, therefore, they can be performed with a high degree of concurrency (see Figure 24). At some points, they have to interact with each other and precise delimitations must be defined.

#### **4.1 Integral Modelling – Spaces**

One aspect of integral modelling regards space modelling, where spaces are identified and defined to meet the requirements. As previously stated, the total space of the building site can be divided into two complementary set of spaces, the exterior spaces and the interior spaces, where the building construction can be identified as a special kind of space, which separate user spaces from each other.

As stated, all spaces can be subdivided and thereby form a hierarchical structure (see section 3.2 and the example in Figure 18). The specific form of the hierarchies can be different so, when considering this, some criteria must be defined. For user spaces, the function or utilisation of the spaces is often set as the criterion. For instance, in an apartment building, this criterion determines that spaces for lifts, stairs and spaces for many technical installations does not belong to one single storey or one single apartment but are common to all storeys

and apartments. Hence, they should be placed in branches separated from apartments (like in the example in Figure 18).

It must be underlined that the space hierarchy only shows whole-part relationships. Many other relationships are also important. For example, not all spaces may be directly mutual accessible, access paths are additional cross-going relationships. Such relationships can be identified and shown as separate structures in the space hierarchy and visualised as e.g. a floor plan (see Figure 25).



*Figure 25 – Floor plan of user spaces in a building*

In connection with the space hierarchy, proposals for the specific location of the spaces can be generated. Here the requirements, which are found in the function modelling phase, must be taken into consideration.

As indicated, use of space hierarchies in synthetic modelling is appropriate because a good overview of the model contents can be established and maintained.

#### **4.2 Integral Modelling – Construction**

Another important side of integral modelling concerns the building construction. As previously stated, every building construction can also be sub-divided and form a hierarchical

structure of building components (see section 3.2 and the example in Figure 19). In integral modelling, only higher levels are identified and relatively few sub-divisions are necessary. Modelling tools should of course be able to generate the hierarchies automatically as soon as the model objects are identified and created.

Like for space hierarchies, it must be underlined that the construction component hierarchy only shows whole-part relationships. Many other relationships are also important, physical as well as mental. Technical installations, for instance, are normally interfering with many other components. Such relationships can be identified as separate cross-going structures on top of the component hierarchy.

### ***4.3 Integral Modelling – Performance***

As mentioned, the performance of the building must match the requirements for the building. The overall building performance can be divided into performance on many different areas and, on the other hand, all construction components contribute more or less to these performances.

In integral modelling, it is important to focus on the performances, which are related to the functional requirements. Thus, it should be possible to derive the performance from attributes of the building and its components. In order to analyse that the building will be able to perform appropriately against the requirements, the attributes must be specified to some degree of accuracy. How many and how precise, they must appear, depend on the importance of the requirements and the methods, which are used in the analysis.

Many analysis tools can carry out comprehensive functional analysis based on very rough data about the building. For instance, a building's estimated energy consumption can be calculated from a rough building model with approximate

quantities of the primary components and data about materials. Such tools are often very effective because they include large amounts of empirical data.

### ***4.4 The Result of Integral Modelling***

As mentioned, the result of integral modelling is a building model, where primary decisions about the user spaces and the construction are made. The rooms of the building and the primary construction components are identified and defined. Further, this means typically that the primary geometry is developed and that visible materials are selected, i.e. the building shape is determined and internal rooms are located relatively to each other with separating walls and floors. The bearing parts of the construction are also determined and developed to the necessary degree; important technical installations are considered; etc.

How far the building model is detailed depends also on how precise the functional requirements must be proved and how capable the analysis tools are. Ideally, these tools should be able to analyse the performance of the building at any level of detail – of course with relative precision. It is, however, typical that most tools require many details in order to work properly. Therefore, if certain performance requirements have high priority, it may be necessary to add some further sub-components and/or attributes.

Altogether, a basic model is formed as a solid foundation for subsequent modelling. The primary structural decisions must be made so that the remaining design and detailing can be performed with reasonable certainty within these limits.

Many modelling tools are not suitable for integral modelling. Typically, they can not support modelling on higher abstraction levels and they can not integrate and capture the alternatives and decisions on all areas. Often, they focus heavily on the

geometry of the construction and, for instance, space modelling is only supported vaguely.

#### ***4.5 Subsequent Design and Detail Modelling***

Based on the results of integral modelling, it is fundamental for building modelling that further modelling can be performed in two parallel activities, which are only loosely related to each other (see Figure 24). Thus, besides detailed modelling of the building construction, it is appropriate to model the user spaces as a separate activity. As stated, this distinction follows from the basic fact about buildings that user spaces and construction spaces are complementary to each other. The construction spaces consist of the construction components.

##### *4.5.1 User Spaces*

Modelling of the user spaces includes both outer and inner spaces. Regarding outer spaces, the terrain with coverings of the drive way, terraces and garden areas can be modelled. In practice, however, the primary emphasis is set on modelling of the interior to further detail as illustrated in Figure 26.



*Figure 26 – Modelling of user spaces*

For the interior, modelling will cover placement, size, form, etc. of various components like:

- Openings
  - Windows
  - Doors
- Surfaces, colours
- Fixtures
- Sanitary
- Convectors
- Terminals for electricity, water, gas, waste water, antenna, etc.
- Lightning, equipment
- Acoustic arrangements
- Panels
- Curtains
- Furniture
- Banisters
- Etc.

##### *4.5.2 Construction*

Modelling of the building construction includes further modelling of the identified building components including specification of attributes and subdivision i.e. identification of sub-components and structure.

- Architecture
  - Primary construction, bearing construction
  - Secondary construction
  - Roof construction
  - Heating installations
  - Ventilation inst.
  - Water distribution installations
  - Waste Water installations
  - Electrical/electronic installations and equipment

Typically floors and walls are initially modelled as solids without considering their precise dimensions and contents. So, in this

activity, they are gradually modelled to sufficient detail, e.g. with material layers, coverings, etc.



Figure 27 – Modelling of walls and floors gradually to greater detail

It should be easy to subdivide an object into smaller parts of the same type without first having to delete the existing object and then create new objects and need to recreate missing attributes and relationships. In some cases, it would be suitable to be able to refer to the original object also as a collection of the subdivided objects<sup>7</sup>. For instance, a wall object may have been created first from corner to corner and a number of attributes are already added to the object. Then, if this object is subdivided into multiple wall sections, the attributes of the existing wall object should be transferred to the new wall objects and the existing wall object should be changed to a collection of the newly created wall objects (see Figure 28).

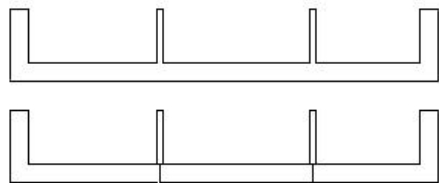


Figure 28 – Subdivision of walls into sections at intersecting walls

Another kind of subdivision occurs when an object is subdivided into objects of other types, e.g. wall objects and floor objects into *material layer objects*. In this case new objects are created

but the existing object should be preserved and relationships should be created between this object and its parts. To consider these material layers as separate objects is necessary, because these objects are very often dislocated compared to its parent wall object (see Figure 28).

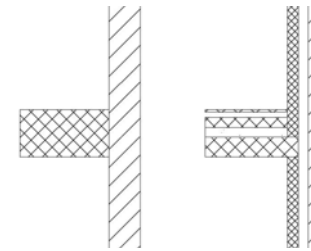


Figure 29 – Subdivision of walls into material layers

Similarly, a roof object may first be created as a simple object and afterwards detailed and subdivided. However, many roofs are rather complicated constructions so many sub-objects must be identified and created.

The relationships between the composite objects and its sub-objects may include relations or constraints between attributes. For instance, the width attribute of a wall object must be the sum of the width attributes of its wall layer objects. Other relationships, however, may not be that simple. For instance, the width and height attributes of a composite wall object may be undefined if the corresponding attributes of the layer objects are not equal. This is often the case for external walls, i.e. walls that face the exterior. When such walls are built on top of foundations, the inner wall layer usually starts higher than the outer wall layer. Similarly, the attributes for horizontal dimensions of a composite floor may be undefined because the corresponding attributes of floor layer objects may be unequal.

<sup>7</sup> In IFC, this is termed nesting.

#### 4.5.3 Objects Belonging to Both User Spaces and Construction

Objects like windows and doors can be regarded as belonging to both activities, so they can be treated differently from project to project. In some projects, they can be modelled as part of the construction but, in other projects, they can be modelled together with the spaces. The design of size, form, colour, joints, etc. belongs most natural to modelling of rooms while modelling of thermal and acoustic properties most appropriately is part of construction modelling.

#### 4.5.4 Model Evaluation, Consistency Checking, Constraint Specification, etc.

Different degrees of detail of models must be identified in connection with the different parties: clients, authorities, advisors, experts, constructors, etc. so that they match the required deliverables to these parties. This graduation must be related to specification of model content and is termed *level of detail*. Consequently, the required deliverables at the end of different intermediate stages must correspond to the content of the model at certain levels of detail.

#### 4.5.5 Levels of Detail

Fundamentally for the building modelling framework, the identification of levels of detail must be divided according to the two parallel modelling activities. Further, the levels can be arranged differently depending on the project. For instance, if visualisation must be prepared early in the modelling process, greater focus may be put on modelling of rooms in the beginning.

Example of identified levels of detail regarding the rooms:

Level 1: Model is the result from integral modelling  
 Level 2: Openings with windows and doors are placed  
 Level 3: Banisters, fixtures, sanitary, terminals, convectors are defined  
 Level 4: Surfaces, colours, etc. are defined  
 Level 5: Lightning, furniture, panels, curtains, etc. are defined

Example of identified levels of detail regarding the building construction and installations:

Level 1: Model is the result from integral modelling  
 Level 2: The primary geometry of the model is defined  
 Level 3: The bearing construction is completed  
 Level 4: The primary installations are defined  
 Level 5: Building products are defined  
 Level 6: Model is completed for exact quantity take off  
 Level 7: Model is completed with details for construction work

## 5. Building Modelling – General Guidelines

Based on the building modelling fundamentals and the framework for building modelling in the previous chapter, this chapter will introduce more specific guidelines for building modelling. These guidelines are developed independent of the functionalities of the current modelling tools. In the next chapters, these guidelines are compared to modelling tools as they are today and more specific guidelines are presented.

The guidelines are developed with the assumption that the building model is kept united as one model throughout the entire modelling process, most suitable as a database in a model server, where all objects of the building model are accessible from multiple users.

### 5.1 Modelling Approach

It has been stated that multiple abstraction levels should be identified regarding modelling of structure and modelling of attributes. These considerations should result in a project dependent modelling approach. If for instance an entirely new building is modelled not based on existing model templates, many abstraction levels would be preferable. In the other extreme, if modelling is based on some well specified requirements and perhaps a draft building form is developed in advance, then modelling is more straightforward – lesser abstraction levels are necessary and identification of model objects representing physical building components can be performed faster. Modelling on each abstraction level should result in presentation of the performed modelling and decisions should be made about the continuing of the modelling on the next level.

One often used approach in building modelling is to put primary emphasis on the outer shape of the building – the architecture.

In this case, abstraction levels are identified in relationship with geometric forms. First the characteristic architecture is modelled and, afterwards, the surroundings and the internal structure are modelled. With this approach, it is important to check that other requirements can be fulfilled, e.g. that the required room can be placed in a suitable way.

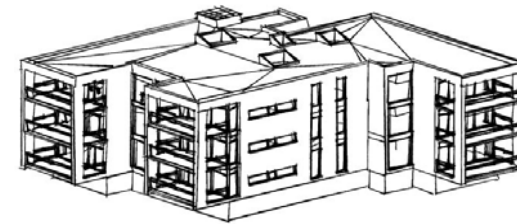


Figure 30 – Focus on the outer shape

Another often used approach in building modelling is to concentrate on the bearing structure of the building. In this case, different abstraction levels can be identified in relationship with the bearing system, i.e. the primary bearing components are modelled first and secondary bearing components are added next. With this approach, it is important to secure that further requirements about for instance the architecture and the rooms are taken into consideration.

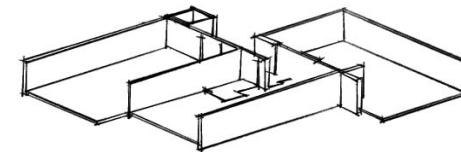


Figure 31 – Focus on the bearing structure

## 5.2 Identification of Model Objects

As stated, working with spaces is very fundamental and one or more hierarchical structures of spaces should be developed and maintained throughout the entire modelling process. When the structure of user spaces is detailed, the rooms of the building are identified and when the construction spaces are detailed and structured, the primary building components are identified.

When the activities design modelling and detail modelling are considered, it is proposed in the modelling framework that integral modelling should be carried out first as a highly interactive and integrated modelling activity. This activity should result in a consistent basic model, where the primary overall design decisions are made and tested against the requirements. Subsequently, modelling of rooms and modelling of the construction can be performed as parallel activities.

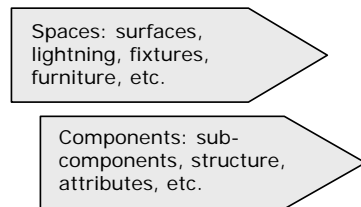


Figure 32– Parallel modelling activities about spaces and construction components

In order to support identification of objects, it is very convenient to have all types of objects available and organised in taxonomies, from which objects can be selected. Thereby, a basic set of attributes are create along with each object. With typical modelling tools, the attributes are about the geometry of the object, but other attributes may be equally important and other types of modelling tools may focus on such attributes. When an object is selected from the hierarchy, some of the attributes are specified, i.e. the values of the attributes are assigned or adjusted, i.e. parametric objects. Typically the

attributes, which are necessary for the definition of the geometry of the object, are specified first and by the modelling tool, this can often be performed in multiple ways, graphically as well as by typing values in fields.

Different approaches can be followed, when objects are identified and created in the model. If a space hierarchy is developed to a suitable degree of detail, then space objects should be created and attributes of these objects should be included. For instance, rooms should be specified with description of what kind of room it is and what the intended use of the room is. If rooms are placed in the 3D space, then initial construction objects may be created automatically. On the other hand, if construction objects are created first, initial space objects may be created automatically.

## 5.3 Subdivision of Model Objects

When construction objects are created from scratch, it may be easiest to create large sections. Later, when such objects must be subdivided into smaller objects, it is important that this can be performed in a fast way as described in the previous section. For instance, walls can initially be created from corner to corner or facades from top to bottom and later such a wall must be subdivided and possibly changed to another type because the internal construction varies. Similarly, a floor object may initially be created as one object and later subdivided because the thickness varies, e.g. in bath rooms compared with living rooms.

Separate modelling modules are convenient for many of these subdivision tasks, e.g. for division of wall objects into material layer wall objects, for division of floor objects into material layer floor objects, or for creating a complex roof object consisting of truss objects, etc. If the modelling tool does not support this subdivision of objects, it may be better to identify the objects on detailed level from the beginning.

## 5.4 Horizontal Layers

Normally, a building model is created from bottom to top; in a regular building: storey by storey, from basement and upwards. Consequently, it is suitable to attach the building objects to horizontal layers so that they can be accessed as a unit. For instance, when the building model is visualised, it is convenient to be able to include or exclude the horizontal layers one by one.

A horizontal layer is defined as a collection of objects, which are placed on approximately the same altitude in the building<sup>8</sup>. The number of layers is often depending on the complexity of the building. For simple buildings, the division on layers may be the same as for storeys but for more complicated buildings, additional layers may be defined. For instance, floor objects and the wall objects may be separated in different horizontal layers. The rooms of a storey could also be collected in a separate layer. Furthermore, multiple horizontal layers may be created with objects, which are placed side by side. For instance, the set of inner walls on one storey could be defined as a different horizontal layer than the external walls.

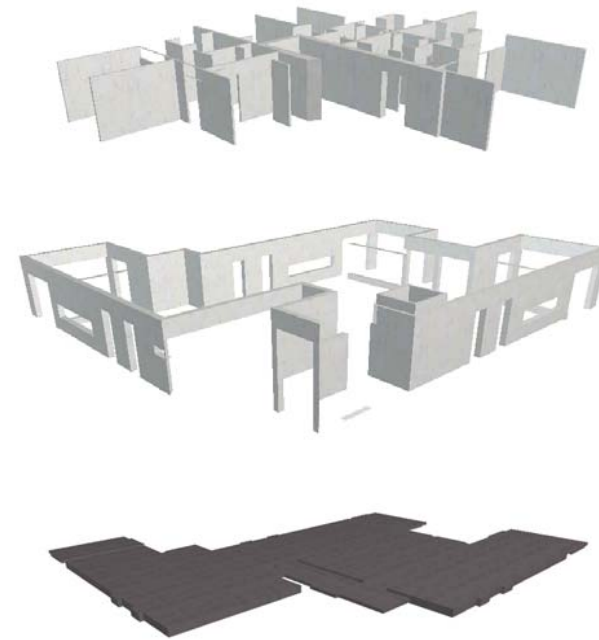


Figure 33 – Three horizontal layer collections: floor slaps, external walls, and internal walls

## 5.5 Horizontal Layers and Storeys

As already indicated, a horizontal layer is different from a storey. A horizontal layer does not necessarily include all objects of a storey. For instance, the roof belongs to the top storey but all objects of the roof could be collected in a separate horizontal layer. Also, if some floors in a building are dislocated half a storey between each other (see Figure 34), additional horizontal layers may be inserted.

<sup>8</sup> Horizontal layers must not be confused with horizontal planes, which are often used in connection with visualisation, i.e. cutting planes. Besides, horizontal layers are different from the layer concept, which is traditionally found in 2D drawing software.



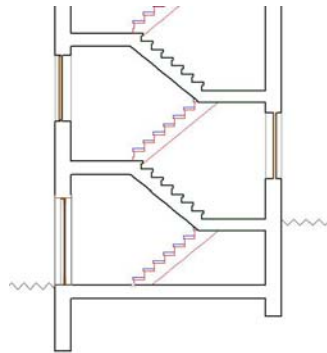


Figure 34 – Storeys with dislocated floors

It should be possible to define the objects belonging to a storey as a set of horizontal layers, i.e. a collection of collections. Therefore, the delimitation between particular layers should follow the need for delimitation between storeys.

An often used overall principle about floors is that floors are attached to the same storey as the walls and columns they are supporting. The roof construction is attached to the storey, which it is covering but, as stated above, it may be placed on a separate horizontal layer. If rooms have special ceilings, they should be placed in separate horizontal layers so that these layers can be handled separately, for instance when visualised.

Depending on how detailed the building model is supposed to be, it may be suitable to set a first delimitation in the rough building model and change this setting, when more details are added to the model. The precise setting of the delimitation between storeys is not straightforward for external walls. If the first model is developed with simple solid objects representing walls, floors, foundations, etc., then it is most convenient to set a storey limit at the top of the foundations (see Figure 35) and then similarly for the storeys above (see Figure 36). Often this level is the level of the upper side of the floor objects.



Figure 35 – Delimitation between storeys at foundations

Later, when external wall objects are detailed and material layer objects are created, the floor objects are normally extended into the outside wall and the inner wall material layers is placed on top of the floor objects. The outside wall material layer is often displaced relatively to the inner wall objects layer. Consequently, it may be most suitable that the storey limit may vary depending on the details of the construction (see Figure 35 and Figure 36).

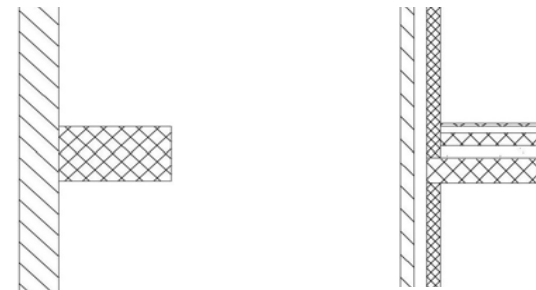


Figure 36 – Delimitation between storeys at external walls

In order to support this, modelling tools should be able to assist in making the necessary changes. For instance if all walls in a horizontal layer have to be shortened, it should be possible to perform this in one operation. Also, if all objects in one or

multiple horizontal layers should be raised to another altitude, this should similarly be possible in one operation.

Objects that reach multiple storeys, e.g. column objects and wall objects in hallways, can either be attached to their lower storeys or they can be cut artificially between storeys. Similar arrangements must be made for pipes, wires and other kinds of installations.

### ***5.6 Other Detailing Guidelines***

There are a number of other typical examples of how the building model should be detailed, for instance special arrangements regarding wall joins, wall corners, wall footings and wall-ends. Special issues arise, when walls and floors are subdivided into wall layers and floor layers so that the individual layers must be handled and joined. Again, it is important that the modelling tool provide support for these operations.

As already mentioned, all modelling tools have a number of object types available for selection, when individual objects are created in a model. First of all, the primary objects of the building model are available but other types of objects can be selected this way, e.g. windows, doors, stairs, kitchen units and sanitary appliances. However, many of these objects may be regarded as temporary because they are supposed to be replaced by specific building products. In such cases, the suppliers of these products should ideally offer models, which can be inserted in the building model. When producers develop product models, they are able to include a large number of attributes, which can be valuable in later life phases. Some models may be configurable so that the producer does not need to be involved in the selection of the final end-product.

### ***5.7 Identification and Further Specification of Objects***

All objects of the building model are uniquely identified by the modelling tool, when they are created. These unique identifiers<sup>9</sup> are used internally for making references to the objects. However, another attribute for unique identification should also be provided to the designer. For instance, all objects can have a name assigned so that they can be identified in a more human readable form. Giving names to the objects may not be necessary for the many ordinary objects of the model but can with advantage be used for objects, which require special attention and around which, some important communication is raised.

As stated, all objects can have a large number of other attributes included. These attributes can be used for many purposes, especially for analysis and simulation. For instance, attributes, which can describe the objects ability to provide thermal insulation, can be used in an analysis of the estimated heat consumption of the building. Similar attributes can be included for e.g. cost analysis and acoustics analysis. Ideally, such attributes should be applied automatically by the modelling tool, when the model is detailed and especially when materials are selected. However, the analysis and simulation tools are often separated from the modelling tool and the attributes are then assigned during the process.

### ***5.8 Application of the General Guidelines***

In the introduction of this section, it is stated that the general guidelines for building modelling are developed independent of the functionalities of the modelling tools, which are currently available. Therefore, the guidelines must be considered

---

<sup>9</sup> In IFC, this identifier is termed Global Unique Identifier (GUID).

accordingly. Some proposals are not possible to implement with the available tools; they are ideas, which have to await better functionalities in future versions of the tools. Other proposals are possible to implement by use of some existing facilities but an unintended way. In the next chapter, the facilities of current modelling tools are summarised and more specific guidelines are presented.

The general guidelines can also be applied differently, depending on how the overall phases of building projects are laid out. Especially, this concerns the degree to which the model is detailed, at what stages it is detailed and by which parties of the project. If for instance, the model has to be approved by certain authorities, specific requirements will be set to the content of the model at certain phases of the project. This may somewhat conflict with the modelling framework and the guidelines. Therefore, project dependent implementation of the guidelines may be considered.

## 6. Characteristics of Current Building Modelling Tools

In this chapter, the primary characteristics of the currently available building modelling tools are presented. Many tools exist, but in the following, only two modelling tools are included as representatives, ArchiCAD (AC) version 9 from Graphisoft and Architectural Desktop (ADT) version 2006 from Autodesk<sup>10</sup>.

The characterisation is based on the previous chapter about modelling and general guidelines for building modelling. There, it is underlined that modelling should be performed at different abstraction levels and that building modelling should include modelling of spaces as well as modelling of the construction components. A building modelling framework is proposed.

Regarding modelling on higher abstraction levels, i.e. requirements modelling and function modelling, both tools are not comprehensively developed. Furthermore, the tools are not suited for separate modelling of spaces on different composition levels. Both tools are primarily focused on construction modelling.

The two modelling tools focus first of all on form and geometry and they are primarily oriented towards architects. Many functions in the two tools look similar on the surface but, studied more closely, they are structured and programmed rather differently. Besides, the terminologies are different and different modelling approaches are recommended.

---

<sup>10</sup> AC and ADT are the modelling tools, which have been used in the research project. During the project, two versions of both tools have been used, AC version 8, AC version 9, ADT version 2005 and ADT version 2006. Graphisoft has now released version 10 and Autodesk has released version 2007.

### 6.1 Building Model Organisation

There is a big difference on how the two building modelling tools have organised the data about the building model. This includes primarily the file structure and content of the computer files.

The internal organisation of the files is proprietary and not revealed by the vendors but, it is known that there is a rather big difference between the two tools at this point. While AC is truly object-oriented, ADT is still to rather high extend a drawing tool.

The building model in AC is stored in one model file and can thereby be characterised as a database implementation. In contrast, ADT can be characterised as a file system implementation because it is based on an external reference system, where multiple separate files, sub-models, are linked to each other.

In addition to the model files, both tools include a library of object types, which can be used for fast creation of objects in the building model. With ADT, these objects are created by making copies in the model files while AC only creates simple objects with the attribute values and makes references to the library objects???. By this organisation, the AC model file is not increased by adding unnecessary data.

The advantage of the database implementation in AC is that cross going references can easily be established and utilised efficiently. For instance, it is possible to make selections of objects of all types and places in the model. This is very convenient, if the same change must be made in multiple objects at the same time. However, when the building model is

getting more detailed, the AC model file can grow very much and cause the tool to become slow.

With ADT, it is only necessary to work with one file at a time and, thereby, the tool can work fast, even on large models. The disadvantage, however, is that the number of files will increase, when the building model is detailed. Furthermore, it is rather difficult to create cross going references in ADT models. An advantage of the external file references is that it gives an opportunity to put the separate sub-models together in various combinations.

Because of the database implementation, AC is much closer to realisation of the concept of building model server with the possibility to perform concurrent operations on separate parts of the model. In accordance with this, AC offer the teamwork functionality.

When working with a building model, it is often suitable to be able to try out alternative solutions. None of the tools have direct support for versioning by which it should be possible to identify and work with multiple versions of the building model at the same time. In AC as well as ADT, this is only possible by saving versions of the model files.

Consequently, the assumption stated regarding the general guidelines that the building model is kept united as one model throughout the entire modelling process can not be supported by any of the tools. This approach can only be followed during architectural design and to a reasonable extend only by use of AC and the teamwork functionality. However, if the use of ADT is organised so that individual users work exclusively on different files, e.g. by use of a file server in a local network, it is also possible to perform concurrent operations with this tool.

## ***6.2 Templates and Library Object Types***

As stated, AC as well as ADT includes libraries of object types, which can be used when objects in the building model are created. The libraries include the primary construction object types like walls, slabs, columns, beams, windows, doors, furniture etc. Object types for spaces are also included. The object types of the two tools are mainly suited for architectural design.

When an object is inserted in the building model, drag and drop operations can be used and the objects can be manipulated with the graphical interface. In addition, each object can be opened and various attributes can be viewed. For most object types, the attribute values can also be changed numerically to suit the user's wishes. Some library object types, like those for sanitary objects, are finished objects and must be used as they are. The two tools work with library object types in nearly the same way.

When it comes to the content of library object types, there is a great difference between AC and ADT. The AC library object types are developed with the Geometric Description Language (GDL) developed by Graphisoft. This language is integrated in AC and open for use by everyone.

With GDL, it is possible to develop advanced and high quality object types with a large set of adjustable attributes. Object types can be developed from scratch or skeleton programs can be generated automatically from existing objects. The variables of a GDL program can represent not only geometric attributes but all kinds of attributes, e.g. material properties, cost values, surface specifications and colours. Object types of GDL can include specifications on an internal structure of sub-objects. Characteristically for GDL object types, the values of the adjustable attributes can be constrained by specification of relations between the program variables.

The ADT library objects are much more simplified and with fewer possibilities for user interaction. Many objects are only modelled with simple geometry attributes and with fixed values.

An international community has been established around GDL and several members of this community have published a large number of add-on library types, which can be downloaded from the internet. Most of them are free or very cheap. Some of the object types are additional generic types but many object types have been developed as models of manufactured commercial products. It is possible to get a GDL plug-in for other applications and for internet browsers.

### ***6.3 Support for Add-On Modules***

AC offers a programming interface (API) by which separate add-on modules can be developed and extra functionalities of the tool can be provided. Such modules can also be developed by third party vendors. With current versions of AC, a number of add-on modules are already installed, e.g. RoofMaker, TrussMaker, StairMaker and MaxonForm. Other modules are import/export modules for separate applications, for instance SketchUp and Artlantis. AC is designed for easy installation of add-on modules. New modules can be loaded by a special add-on manager.

ADT is not to the same extent designed for add-on modules as AC and third party modules are typically made available as separate software tools that operate on the ADT model files. This is the case for e.g. Viz Render and SketchUp by use of import/export modules integrated in ADT in advance. The same goes for the IFC interface, which must be installed as a separate program.

Graphisoft develops their own IFC add-on modules and has made them highly integrated with AC.

### ***6.4 Support for Sketching***

In the beginning of the design modelling phase, sketching is important. Later more precise modelling becomes the most important. To meet these demands the two modelling tools offer some specific modules.

ADT has a module called Massing and a module called Space Planning. These modules are relatively easy and flexible to use. Objects created by Massing are parametric and can be formed and shaped very freely. A set of basic forms are available but new forms can be created. New forms can also be created by combining existing forms with use of addition and subtraction operations. At any time, objects created by Massing can be identified as either space objects or construction objects. Space objects can be further specified and when space objects are suitable placed, construction objects like walls and floors can be generated automatically.

AC does not include a similar module. However, the add-on module Maxon Form is offered. This module can be used to create new forms in a similar way as ADT Massing. Furthermore, Maxon Form is designed to create free forms, like organic forms. Objects created by Maxon Form can be identified as construction objects (not space objects??). Space objects can be created before construction objects but AC does not offer the possibility of automatic generation of construction objects around space objects.

### ***6.5 Creation and Detailing of Model Objects***

As already indicated, all primary building model objects are created by insertion and adjustment of objects from library object types. During subsequent modelling, the objects may need further adjustments and sometimes existing objects must be replaced by other objects. Both modelling tools support these operations equally.

Before the objects can be inserted in a model, both tools require that at least one storey is created and selected as the current storey. When subsequent objects are inserted, they are automatically attached to the current storey.

In addition, both tools can work with layers so that each object can be attached to a layer. Some features are included in the tools to carry out operations on all objects belonging to a layer. This is mainly aimed at visualisation.

Some modelling operations can be performed graphically by move or drag operations. During such operations, both tools perform snap operations so that adjacent objects can snap to each other, e.g. wall to wall, slab to wall, wall to foundation and column to slab.

As stated above, ADT can automatically generate walls and floors around spaces but if these objects at a later stage are going to be subdivided, e.g. walls into individual wall segments at each room, it is necessary to create new objects and delete the existing object.

Instead of automatic generation of objects, AC gives the nice opportunity to create connected walls simply by making a series of markings at corners of the wall on a floor plan. By this, each wall goes from corner to corner and, if such walls must be subdivided at rooms, new wall objects must be created similar to what is necessary in ADT.

In the early stages of design modelling, it is acceptable that the floor slabs just snap to the inner side of walls. Later on, when the walls are defined as composite walls, floor slabs should be extended so that they are supported on wall parts.

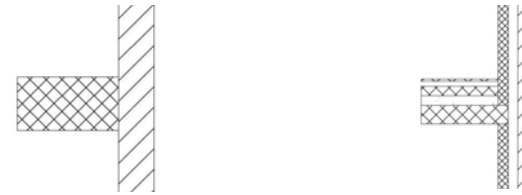


Figure 37 – Detailing of floor slab and supporting wall

This is not possible to handle in a satisfactory way in any of the two tools. In AC, composite walls and slabs does not consist of sub-parts. The specification of composites of walls is only used graphically and it is not possible to model the individual composites.

In ADT this is different. Once the wall is divided into its composites, they act as individual objects. In a wall therefore it is possible to specify a displacement of the back part of the wall leaving space for a slab to be placed so that it is supported by the back part of the wall. Although walls can be modelled to the required detail, ADT does not provide means for slabs to be subdivided into composites and, therefore, these details can not be satisfactory modelled.

Both tools have support for inserting openings with windows and doors. However, there are differences about how detailing of walls are performed and how interrelations between objects are handled by AC and ADT between walls and windows or doors. Primarily, this is about how a window or an opening relates to the wall cavity (see Figure 38).



Figure 38 – Detailing of wall at windows and doors

In AC, the solutions are specified by setting what is termed Cavity Closure. Cavity Closure is included in the window, door objects and can easily be set. AC offers only a limited number of different design alternatives, which can be selected very easily.

In ADT, these specifications are termed Wall Endcap and they are not part of the window, door or opening objects but need to be performed as a special wall design. In this way it is possible to create any desired design but it requires some work to be performed. As Endcap is solely a part of the wall, it does not interact with opening or window/doors.

Similarly, AC and ADT offer support for detailing of walls with special ends (see Figure 39). In AC it is termed Wall Ends and the tool offers a limited number of different design alternatives to select. Detailing in ADT is also performed by Wall Endcap as special design.

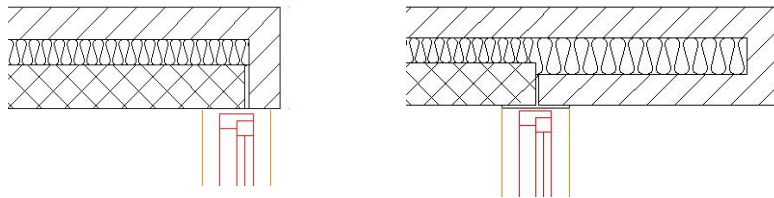


Figure 39 – Detailing of walls at wall ends

As described above, AC offers the add-on modules RoofMaker, TrussMaker and StairMaker. These modules are primarily considered as detailing tools and, characteristically, they create objects of rather high quality.

Both tools recommend that all details, which it is too difficult to model as correct 3D objects, should be developed as 2D drawings. Both ADT and AC have detail modules to support this kind of work.

## 6.6 Graphic Presentation of Building Models

AC as well as ADT has advanced means for graphic presentation of building models from the structures of data objects, which are represented in computer memory.

Visualisation of building models is preferably made as 3D projections to 2D – perspective depiction. In addition, the traditional 2D floor plans and elevations are always required. Both AC and ADT are equipped with modules or tools, which can produce all such graphic presentations both on screen and paper and in different scales. As already stated, some details of the planned building may even be represented solely as traditional drawings.

Especially, AC has good features for fast visualisation in 3D. Individually selected objects, e.g. on a floor plan, can easily be presented as a 3D projection and, if wanted, the geometry of the objects can even be manipulated in the 3D graph. In addition, AC offers special features for selection of objects and for setting cutting planes horizontally and vertically. AC has also advanced features for generation of different looks of the 3D presentation – rendering.

In order to produce high quality visualisations, it is necessary to include data about the surfaces of the construction objects, for instance by specification of materials. This is necessary for calculations about how light is reflected from the surfaces.

In AC, the integrated LightWorks module makes use of such data for high quality rendering. With ADT, similar visualisation can be carried out by the separate tool Viz Render, which uses a link between the tools. Both software products can work with daylight, sunlight as well as artificial light from lamps.

Both modelling tools have the possibility to swing back and forth between presentations in different scales.



## ***6.7 Data Extraction for Other Purposes***

AC as well as ADT can produce some tables and lists over the model content. Thereby, different simple overviews can be acquired. Furthermore, the tools have modules by which it is possible to make calculations about estimated quantities and costs of the modelled building. Although these modules can be configured to suit various requirements, they are rather inadequate for professional use. More advanced tools should be used.

Very important data extractions are performed by the publishing modules, which can first of all produce drawings. The modules can work with templates or drawing masters for specification of sheet size, orientation, colour palettes, etc. The publishing modules are similar to each other in AC and ADT.

In general, there is a need for extraction of data from building models to other separate analysis and simulation tools within areas like static analysis, energy analysis, acoustic analysis, and quantity take off and cost estimation.

Both tools can save building models in their own proprietary file formats as well as a number of other file formats. Especially the DWG file format of ADT has been used very much. Although it is proprietary, many third party vendors have made software, which can read.

The most interesting file formats are those developed by IAI to enable representation of the IFC data model, the IFC file format and an XML based file format. These formats have increasingly become important since they are independent of commercial vendors.

AC offers a rather comprehensive implementation of IFC and AC is gradually being structured to conform even better to IFC. ADT, on the other hand, is not structured well for IFC and, as mentioned previously, the IFC interface for ADT is developed by

an independent third party vendor. Due to the internal representation in ADT, this interface is not nearly as complete as the one for AC. Especially, the import operation is insufficient.

Consequently, the use of model servers to support the entire modelling process with multiple partners is not yet recommendable because the quality of IFC interfaces is inadequate. Currently, Graphisoft is trying to improve the possibilities and AC can to some extent work directly with the EPM IFC Model server. ADT has no interface, which can connect directly to model servers.

## 7. Building Modelling – Specific Guidelines

In this chapter, relative specific guidelines for building modelling are presented. The guidelines are developed on the basis of the previous chapters and, in particular, the general guidelines for modelling are adjusted to what is advisable with the current features of the two building modelling tools, ArchiCAD (AC) version 9 from Graphisoft and Architectural Desktop (ADT) version 2006 from Autodesk.

### ***7.1 Limitations Regarding Modelling Approach***

Because of the current status of these features, the use of the two tools is rather limited compared to the general guidelines for building modelling. As stated the tools are primarily aimed at architects and can only to limited extent support the entire modelling process performed by multiple parties.

As also stated, both tools do not adequately support modelling at higher abstraction levels, e.g. function modelling and sketching. They are best suited for modelling which can be initiated by use of the object types, which are included in their libraries. If it is suitable to perform sketching by working with these objects, it can sometimes be an efficient approach.

The most often preferred modelling approach for architects is to emphasise on the outer shape of the building. Both tools can support this kind of modelling but their special modules are the most suitable, i.e. Massing in ADT and Maxon Form for AC. As previously mentioned, another approach is to concentrate on the bearing structure of the building. Although this is most often not performed by architects, both tools are well suited for this kind of modelling. None of the tools, however, have modules, which can perform structural analysis.

As stated, space modelling can be performed with both tools but, because AC can not make use of a space model to generate construction objects, it has less value in this tool.

### ***7.2 Creation of the First Model***

When the first version of a building model is initiated and detailing of the model is planned, it is recommended that a number of different possible approaches are considered and the most suitable approach is selected. These considerations are often related to how and when detailing of the model is supposed to be performed. If different alternatives are required as foundation for preliminary decisions about the model, rough models may be sufficient. On contrary, if decisions about the overall shape and structure are made already, it would be advisable to create a more precise model from the beginning, i.e. the measurements of objects should be precise and consistent.

As described in the general guidelines and stated about the two modelling tools, detailing of construction objects can be performed in many ways but AC and ADT have limited possibilities for subdivision of existing objects.

In both tools, it is necessary to create new wall objects and delete the existing wall objects when they have to be subdivided into smaller wall segments. Therefore, it must be considered what is most convenient; either to create the walls the easiest way from the beginning and re-create the walls at a later stage or to create the walls in segments at once.

Similarly, when floors are created, it is suitable to consider what the end result should be. If the elevation, height or the material layers vary from room to room, it may be most convenient to

create the different floors at the beginning. However, it also depends on how walls are subdivided into material layers. If this requires change of wall width, all floors must be changed anyway.

For creation of more complex construction objects in AC like trusses, roofs and stairs, it is advantageous to use the special add-on modules. However, for the first model version, it can be advantageous to create more simple objects and replace them at a later stage.

As previously stated, both tools require that all objects are attached to a storey. So, before each object is created, the right storey must be selected as the current storey. Furthermore, it is recommended to follow as much as possible the guidelines, which is previously described about delimitation between storeys.

### ***7.3 Detailing of Objects***

Beside the issues about subdivision of walls and floors into smaller sections, there are a number of other issues about detailing that must be handled differently in the two tools.

Modelling of material layers is performed differently in the two tools and, especially, the possibility to define dislocated layers is different. In ADT, the support for modelling of dislocated walls can be used, but there is no support for this in AC. About detailing of walls at openings in AC, the Cavity Closure feature can be used, if the available options suit the requirements. Similarly, the Wall Ends feature in AC may be used. In ADT, the similar feature is Endcap.

In AC, the advanced add-on modules Truss Maker, Roof Maker and Stair Maker can be used to create detailed objects. Further in AC, skilled users can with GDL carry out modelling down to any detail all objects, which can not be created from library

object types. This is of course very suitable, if such objects can be reused in other building models and also if GDL is used to create component models, where some attributes are made adjustable so that they can be configured for multiple uses.

With both tools, it is also possible to use the free form modules, i.e. Massing in ADT and Maxon Form in AC. Besides, it may be easiest to make special use of the library object types, e.g. by combinations of slab, wall, beam and column elements. However, this must be regarded as misuse of the library object types unless each resulting object can be retyped. If for instance, some column, slab and wall elements are put together to form a chair object, this object should be typed as a furniture object.

In cases, where correct detailing of objects is not required, it is necessary in both tools to make further specifications by detailed 2D drawings...

Regarding detailing of a building model, it should always be considered carefully, how detailed it should be made with these modelling tools. This is also dependent on how modelling tasks are divided among partners. For some of the detailing operations, other more appropriate tools may be better to use.

Also, detailing must be considered over time and in connection with what is required at different stages of the modelling process. In order to make decisions about this, it is recommended to study the proposed building modelling framework and levels of detail, which is presented in chapter 4.

### ***7.4 Visualisation and Model Publishing***

During the modelling process, it is very valuable to make use of the visualisation features of both tools. Especially, the possibilities for 3D presentation are important for the designer. Very often, traditional errors of wrong spatial placement of

objects can be avoided by a 3D presentation, where the presented objects are seen from different viewpoints and from different distances. Especially, heavy use of the features of AC for selection and presentation of objects is recommended.

Traditionally thinking indicates that electronic drawings should look similar to hand made drawings. However this does not have to be the case. The electronic media will set its own standards and as long as the electronic drawings contain the necessary information for the building process the way drawings look like will change.

### ***7.5 Data Extraction***

Data extraction for the closely related separate modules and third party applications are easily performed. So, if some valuable analysis or additional modelling can be performed in this way, it is recommended to make use of these possibilities.

The use of IFC based applications get increasingly interesting. Many of these applications are currently low cost applications but also their quality varies. The benefit of these tools also depends very much of the quality of the IFC interfaces on both sides.

One of the most recommendable IFC based application is the IFC Model Checker from Solibri. With this application, it is possible to check the model for a long list of errors, inaccuracies, inconsistencies, etc. which obviously can secure the quality of models at early stages of the modelling process.

## Conclusion

Building modelling is a special kind of system modelling. Therefore, many principles, terms and approaches from systems theory can be applied to building modelling. However, there are also special conditions for building modelling to take into consideration so it is necessary to develop particular modelling approaches aimed at building modelling.

Based on the fact that buildings can be regarded as a system, a number of system concepts are directly applicable to building modelling and building models. However, the building domain has also developed its own terminology, which should not be completely neglected. But because computer based building modelling is a relatively new area, some traditional used terms may be inappropriate for future use. Consequently, new terms must also be introduced and precisely defined.

Based on the general understanding of systems, system models and system modelling, building modelling is regarded as synthetic modelling on multiple abstraction levels. On the highest levels of abstraction, a rough structure of the building is modelled and few attributes are identified and defined. On the lowest levels of abstraction, a detailed structure of objects with many attributes is handled.

In accordance with this, three kinds of modelling are identified by what is the main substance of modelling: function modelling, design modelling and detail modelling and, further, three corresponding primary phases of all modelling projects are identifiable.

Because a building is generally regarded as a product, building modelling is often compared with product modelling. However, buildings must also be considered differently. Working with spaces is what makes buildings a special kind of product. Above all, the creation of spaces is normally the primary purpose of

building construction so modelling of spaces apart from modelling of the construction is unique for building modelling.

When resources are allocated to computer based modelling tasks, it is important to consume these resources in a carefully prepared way. Because computer based models offers good opportunities for visualisation and other kinds of analysis, it is extraordinary important to make thorough planning for the initial stages of modelling.

Based on the fact that it is important to make the right decisions in the early phases of modelling, a framework for modelling is introduced, where the initial activity is termed integral modelling. This activity is considered highly iterative and integrative and it produces the first version of the building model. The intentions are that this should be carried out with a minimum use of resources but in a way in which it is proved with reasonable probability that the functional requirements can be fulfilled and that all design proposals can be seen and tested as a whole.

Two subsequent modelling activities are identified as modelling of user spaces and modelling of construction components. Both activities follow integral modelling and can to a great extent be performed as concurrent activities. They are only interrelated to a minor degree. These activities contain a mixture of design modelling and detail modelling. Subsequent detailing activities are carried out to the required level.

Based on a detailed description of this framework and the included activities, some general guidelines for building modelling are presented and to some degree related to future and present modelling tools. The guidelines cover considerations about the initial modelling approach, how model objects are identified and how they are subdivided.

In order to apply the guidelines to currently available building modelling tools, the two selected CAD tools are evaluated and specific guidelines are developed.

By the statements and the reasoning in this report it can be concluded that building modelling has its own characteristics but nevertheless can be seen as special kind of system modelling. Further, it is useful to apply some general system theory concepts and approaches to building modelling.

## Glossary

### 2D

Two dimensional, i.e. a plane. Computer screens and paper sheets are 2D media and what is shown on these media is shown in 2D, e.g. a projection from 3D.

### 3D

Three dimensional, i.e. the space. 3D objects are represented with 3D geometry. Presentation of such objects on 2D media is performed as 2D projections.

### 3D modelling

Modelling with objects represented in 3D. This kind of modelling is only one approach of *building modelling*.

### abstraction mechanism

Mechanisms for making abstractions. Two fundamental abstraction mechanisms exist: *composition* and *classification*.

### abstraction

Intentionally simplification of something. A *model* is an abstraction of something that often exists in the real world. The degree of abstraction or abstraction level characterises the extent to which the simplification is made.

### analysis

Analysis (of something existing) is 1) to investigate *properties* of it and 2) to divide it into components and structure.

### analytic model

Model of something existing, often physical.

### attribute

Part of a *model object*. Attributes are separated into *appearance attributes* and *behavioural attributes*. A synonym for appearance attribute is data attribute. A synonym for behavioural attribute is method, i.e. procedure or function. Appearance attribute has a name and a value. In *synthetic models*, such attributes also has a data type.

### building component (subtype of *component*)

General concept for all parts of a building. Building components are either spaces or construction components.

### building model

Model of a building. A building model can be of different nature, can be expressed in different ways and can contain different selections of data about the building. Report view: *computer-based building models*.

### building modelling

Development of building models by modelling.

### building modelling tool

Software with which *computer-based building models* can be created.

### class

See *object type*

## classification

*Abstraction mechanism*, which forms hierarchies of *object types* determined by their *attributes*. Such a hierarchy is termed *taxonomy*.

## component

A part of something, i.e. a component can be considered a part of a larger component (super-component) and each component can be divided into smaller components (sub-components).

## composition

*Abstraction mechanism*, which forms hierarchies of objects determined by whole-part relationships.

## computer-based model

*Model* represented in computer memory. Model objects consist of *data attributes* and *data structures*.

## construction component

Component of the building construction.

## data attribute

See attribute.

## data object

Object in *computer-based models* or in software.

## data object reference

A directed relationship to a *data object*.

## data structure

*Structure of data objects*.

## design modelling

A kind of *modelling*, where a *model* is created and modelled to a suitable level of detail.

## detail modelling

A kind of *modelling*, where a created *model* is detailed to a require level.

## element

A *component*, which in a certain context is considered not to be further subdivided. Building components in IFC are termed building elements.

## external wall

Wall, which is facing the exterior.

## function modelling

A kind of *modelling*, where purpose and primary functional characteristics are identified and specified.

## IAI

Acronym for *International Alliance of Interoperability*.

## IFC

Acronym for *Industry Foundation Classes*.

## IFC model server

A *model server*, which implements the complete *IFC data model*, i.e. the database can accept and store all *IFC objects*.

## Industry Foundation Classes

*Data model* developed by *IAI* for representation of buildings. *IFC* is ISO standard ISO PAS xxx.



## information

data + *semantics*.

## integral modelling

Integrated and iterative design process in the beginning of the *design modelling* phase, where a minimal amount of resources are used to create a first consistent *model* version. All functional requirements are considered and a balance is sought between contradicting requirements. If possible, the *model* is tested and analysed, e.g. by analysis applications.

## International Alliance of Interoperability

International organisation independent of companies in the building industry. A primary responsibility is to develop *IFC*.

## ISO 10303

ISO standard for exchange of product models. Previously named *STEP*.

## material layer

Subdivision of construction components, where the subcomponents normally consist of different materials.

## model

Intentionally simplified description of something, e.g. *system*, building, *object*, etc. Models are *analytic* created by *analytic modelling* or *synthetic* created by *synthetic modelling*.

## modelling matrix

Diagram showing *modelling* by two dimensions, modelling of *properties/attributes* and modelling of *structure*.

## modelling

To create a *model*, *analytic* or *synthetic*. When the model is created, a certain purpose is fulfilled.

## model object

Basic *component* of a *model*. A model object consists of *attributes* and a *structure* of model objects.

## model server

Database system, which implements a complete *data model*, e.g. for representation of physical buildings or *building models*.

## object

Real world: Something physical or mental, which is treated by an action or thought about by a human being.

Models: the basic *component*, which consists of *attributes* and a *structure* of objects.

Software: common for variable, method, record, data structure.

## object type

Description from which individual *objects* can be generated. Each object type has a description of *attributes* and *relationships*. Relationships are described by *relations*, which are specifications of how data structures can be created, i.e. rules or constraints. Object type is identical to *class*, except that class indicates an *analytic* view while type indicates a *synthetic* view.

### object-oriented

Paradigm for *models* and software with which *model components* and software *components* resembles real world living organisms by including *behavioural attributes*.

### parametric model object

*Model object* in which the values of the included *attributes* can be changed.

### Part 21

Part of *ISO 10303 (STEP)* standard specifying a file format for representation of product data. *IFC* files are normally referred to as Part 21 files.

### presentation form

Method used for presentation of an extraction from a computer-based model. A presentation form is often highly related to the selected media.

### property

Characterisation of something, e.g. a *system*. It has a name and a value, for example height: 200 cm.

### semantics

The meaning of something.

### space

Spatial delineation. Spaces are divided into two complementary kinds of spaces, the *user spaces*, which are utilised by the users, and the *construction spaces*, occupied by the construction *components*.

### STEP

Standard for exchange of product models – *ISO 10303*.

### structure

Set of relationships.

### sub-model

Model extraction, which is determined by a certain purpose, e.g. for getting a simplified view of the model.

### synthesis

Synthesis (of something new) is 1) to create it by relating existing *components* to each other by a structure and 2) to add *properties*.

### synthetic model

*Model* built from ideas, thoughts and imaginations and obtained in some kind of representation. Synthetic models are typically created as a foundation for some kind of new construction, which eventually will become physical, an artefact.

### system

Delineated part of the universe. Internally, a system has a set of properties and a structure of sub-systems. The system form is constituted by a set of visible *properties*, while the remaining properties and the internal *structure* constitute the system content. Properties can also be divided into *appearance properties* and *behavioural properties*.

### system model

Intentionally simplified description of a system, fulfilling a certain purpose.

taxonomy

Hierarchy of *object types* showing general-special *relationships*.

## Literature

[Choo 2004]

Choo, S. Y.: *Study on Computer-Aided Design Support of Traditional Architectural Theories*. Technische Universität, München, 2004  
[tumb1.biblio.tu-muenchen.de/publ/diss/ar/2004/choo.pdf](http://tumb1.biblio.tu-muenchen.de/publ/diss/ar/2004/choo.pdf)

[Eastman 1999]

Eastman, C. M.: *Building Product Models: Computer Environments Supporting Design and Construction*. CRC Press, 1999, ISBN 0-8493-0259-5.

[Ekholm 2002]

Ekholm, Anders and Lehtonen, Riikka: *Creative construction Briefing with Object-Oriented Technology and IFC*. eSM@rt 2002 Conference Proceedings Part A, pages 229-234. University of Salford 2002. ISBN 0902896415

[Fällman 2003]

Fällman, Daniel. *Design-oriented Human-Computer Interaction*. Proceedings of CHI2003, Conference on Human Factors in Computing Systems, Fort Lauderdale, Florida, published in CHI Letters, Vol. 5, Issue No. 1, pages 225-232. ACM Press, 2003.

[Froese 2002]

Froese, Thomas: *Current Status and Future Trends of Model Based Interoperability*. eSM@rt 2002 Conference Proceedings Part A, pages 199-208. University of Salford 2002. ISBN 0902896415

[Kagioglou 1998]

Kagioglou, Michael; Cooper, Rachel; Aouad, Ghassan; Hinks, John; Sexton, Martin; and Sheath, Darryl: *A Generic Guide to the Design and Construction Process Protocol*. University of Salford 1998. ISBN 0-902896-17-2

[Kiviniemi 2005]

Kiviniemi, A.: *Requirements Management Interface to Building Product Models*. STANFORD UNIVERSITY, 2005  
[cife.stanford.edu/online.publications/TR161.pdf](http://cife.stanford.edu/online.publications/TR161.pdf)

[See 2007]

See, Richard: Building Information Models and Views. In: Journal of Building Information Modeling, National Bim Standard and National Institute of Building Sciences, p.20 ff., Fall 2007.

## Web Sites

IAI International: <http://www.iai-international.org/>

IFC Wiki: <http://www.ifcwiki.org>

BLISS Project: <http://www.blis-project.org/>

EuroStep: <http://www.eurostep.com/>

EPM Technology: <http://www.epmtech.jotne.com/>

SECOM: <http://www.secom.co.jp/english/index.html>

## Appendix A – System, System Models and System Modelling

Methodologies for system development should be based on general systems theory ([Boulding 1956] and [Bertalanffy 1967]). In accordance with this theory, a *system* is a delineated part of the universe. Internally, a system has a set of *properties* and a *structure of sub-systems*. Properties can be divided into some, which are *visible* from surrounding systems, and the remaining, which are *invisible* or *hidden*. The visible properties constitute the *system form* while the remaining properties and the internal structure constitute the *system content*.

In a system, the properties can also be divided into *appearance properties* and *behavioural properties*. The appearance properties are properties, which can be determined through some kind of measurement, while behavioural properties are the system's ability to perform something. Figure 1 illustrates a system.

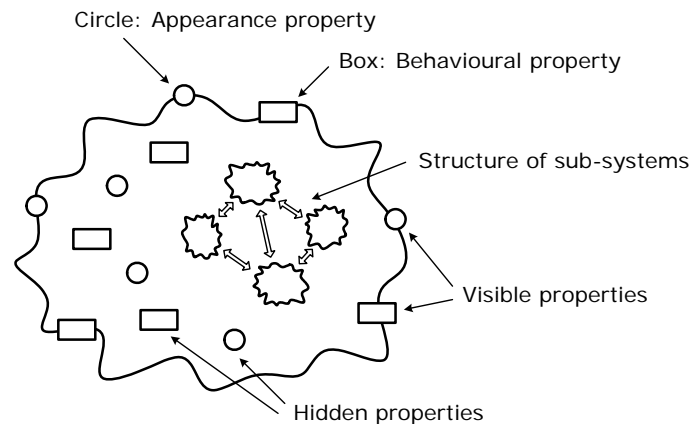


Figure 1 – Illustration of a system

A *system model* is an intentionally simplified description of a system. When the model is created, a certain *purpose* is fulfilled, i.e. some choices are made in order to select the most

important properties, sub-systems and relationships. Thus, a system model can e.g. be suitable for communication between different actors because, with the model, it will be possible to concentrate on the most important aspects of the system. In order to underline the difference between system and system model, the term *attribute* will be used in models instead of the term property.

### Analysis and Synthesis

System modelling includes two fundamental concepts *analysis* and *synthesis*.

*Analysis* (of an existing system) is 1) to investigate properties of the system and 2) to divide the system into system components and system structure.

*Synthesis* (of a new system) is 1) to create the system by relating existing systems to each other by a structure and 2) to add properties to the system.

Based on these concepts, *analytic modelling* is a modelling approach, which creates an *analytic model* through abstraction and *synthetic modelling* is a modelling approach, which creates a *synthetic model* as a basis for realisation of an artefact. Both these modelling approaches are illustrated by Figure 2. Analytic models are models of something existing, often physical. Such models serve as a description, where a deliberate simplification is made, i.e. a selected set of properties, components and structures.

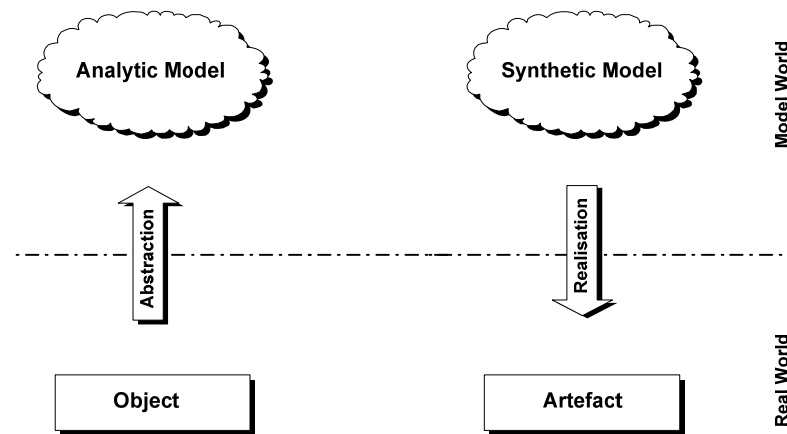


Figure 2 – Analytic and synthetic modelling

In contrast to analytic models, synthetic models are not built from anything existing but instead, a synthetic model is created as a foundation for some kind of new construction, which eventually will become physical. Hence, synthetic models are typically built from ideas, thoughts and imaginations and obtained in some kind of representation. Consequently, by modelling is a development approach, where a synthetic model is created as an intermediate result and the final result is an implementation of the model in the real world.

It is important to realise that synthetic modelling does not purely involve synthesis. It is normally a mixture of synthesis and analysis but synthesis is the primary substance. For instance, when a proposal is created, it is often appropriate to analyse a set of alternative possibilities.

Characteristically for synthetic modelling, it is also important to divide between two separate approaches: modelling of *requirements* and modelling of *solutions*. Modelling of requirements is to specify limitations for solutions as e.g. minimum/maximum values of selected attributes. Modelling of

solutions is to generate possible alternative results that fulfil the requirements.

An important reason for synthetic modelling is to enable manipulation and test of the model before the actual physical system is built. Modelling should make it possible to ensure that proposals are correct and by various presentations of the model at different stages, it should be possible to see the consequences of decisions and to obtain a good impression of the final result.

When synthetic modelling is performed, it is often important to view many different aspects of the model and to represent the model on multiple abstraction levels. This is especially necessary at the beginning of the modelling process before decisions are made about various details. One way of abstraction is to focus on multiple systems instead of individual systems. Analysis, synthesis and modelling can be performed on individual systems as well as a set of systems as a whole.

### **Fundamental Abstraction Mechanisms**

A fundamental basis for system modelling is the *abstraction mechanisms composition and classification* ([Smith 1977a], [Smith 1977b], [Rosch 1978] and [Sowa 1984]). In essence, composition is the development of a hierarchy of systems determined by *whole-part relationships*. These relationships can be identified by the complementary operations *aggregation* versus *separation*. The composition abstraction mechanism can be applied to both individual systems and to multiple systems. Classification can only be applied to a set of systems – *classes* – and involves development of a hierarchy of systems determined by the properties of the systems. The classes are related to each other by *general-special* relationships. Such a hierarchy is termed *taxonomy* and can be identified by two complementary operations *generalisation* versus *specialisation*.

The abstraction mechanisms are related differently to analytic modelling and synthetic modelling. Composition is used in analytic modelling to identify primary system structures either of individual systems or of classes of systems whereas, in synthetic modelling, composition regards identification and definition of *constraints* about how structures can be created. Classification is often performed in connection with analytic modelling and the result is used as a foundation for synthetic modelling. But, when a synthetic model has to be created and available taxonomies are insufficient, further development has to be performed and if no appropriate taxonomies are available, some classifications need to be carried out first. Hence, classification supports the identification of model components and structure at the type level. Based on this, individual components and structures can be selected and created.

In order to distinguish clearly between analytic modelling and synthetic modelling, the term *class* is only used in analytic modelling while the term *type* is used instead in synthetic modelling. Hence, classes are identified from existing individuals while types are used as the basis for creation of new individuals.

### Generic Model Component

In accordance with general systems theory, a generic system model component is introduced and illustrated in Figure 3. Corresponding to the illustration of systems in Figure 1, *appearance attributes* are shown as circles and *behavioural attributes* are shown as boxes. Appearance attributes are also termed *factual attributes*.

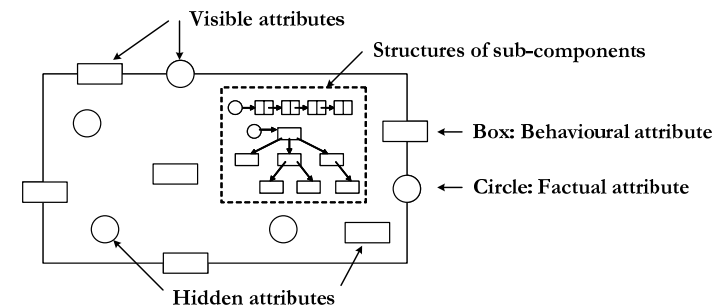


Figure 3 – Generic model component

Such model components can be used to represent every kind of system. Visible as well as invisible parts of the system can be represented and, besides the physical parts and relationships between systems, some mental model components and relationships can be created. For instance, a component can be created in order to handle a group of model component as a whole.

When a synthetic system model is considered, a foundation for the model must be established by creating *types* of components. Therefore, a generic model of a system type is also introduced and used in the following. The generic component type is illustrated in Figure 4 as a box with rounded corners.

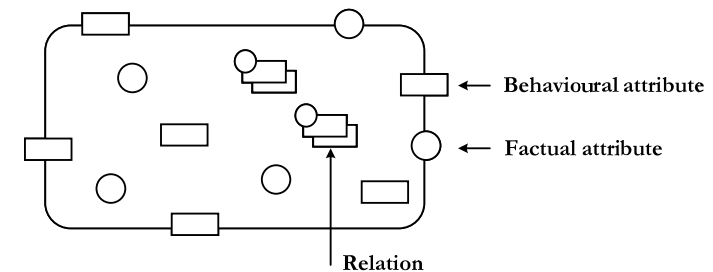


Figure 4 – Generic component type

Each component type includes a specification of a set of attributes with *name* and *data type*. Identification and specification of structures can also be included in the component types by creating *relations*, which formulate the *constraints* regarding attributes and combinations of sub-components (see figure 11). Simple relations specify *references* and *collections*. The main purpose of references is to create structures, which avoid *redundancy*. When further specifications are added to relations, special component types may be added to the model.

A component type is a kind of template and, from each type, an indefinite number of components, instances, can be generated (see Figure 5).

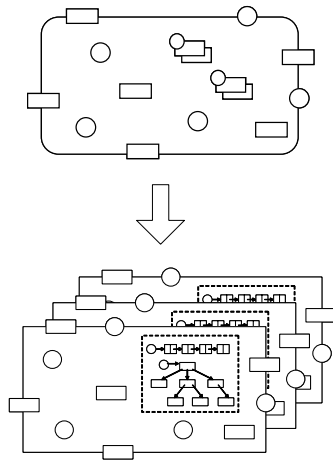


Figure 5 – The component type is the basis for generating components (instances)

### ***Identification of Multiple Abstraction Levels***

To illustrate how multiple abstraction levels can be identified, a complete system is considered and the generic model component in Figure 3 can illustrate such a system.

Simplification in an analytic modelling approach can be performed along two dimensions: *reduction of structure* and *reduction of properties*. Reduction of structure means identification of upper level systems, which aggregate lower level systems. Ultimately, the internal structure of the system is disregarded completely. Reduction of properties means disregarding sub-sets of the properties. First internal properties can be ignored so that only the form of the system is considered. Subsequently, a further degree of abstraction would be to focus on the behavioural properties and ignore the factual properties. Such a description is solely aimed at the operations, which can provide transformation of input to output. Hence, at the top level of abstraction, the emphasis should be put on what primarily represents the abilities to perform functions (see Figure 6).

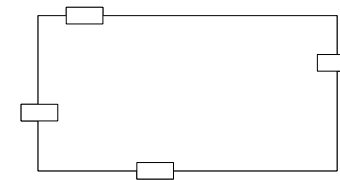


Figure 6 – Analytic description on top level of abstraction – behavioural attributes

Applied to synthetic modelling of systems, an overall approach of identification of multiple abstraction levels would be to following the analytic approach in reverse order. In general terms, this will contain considerations about two corresponding dimensions: *modelling of properties* and *modelling of structure*. Modelling of properties means gradually identification, definition and specification of properties of the new system. Typically,



properties of the form are modelled before those of the content. Modelling of structure means identification of sub-systems and their mutual relationships. For each sub-system, both modelling of properties and modelling of structure are applied recursively. The combination of the two modelling dimensions forms a modelling matrix, in which all modelling projects start in the upper left corner and are supposed to end in the lower right corner. Which route to be followed more specifically through the matrix depends on circumstances and will typically differ from project to project. In some projects, it is important to allocate more work on a high level of abstraction before the details are defined. But in other cases, it is possible to go faster to identification of details.

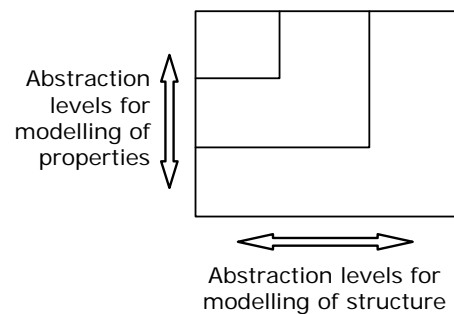


Figure 7 – Modelling matrix: modelling of properties and modelling of structure

Modelling of properties on top level comprises the system's ability to perform functions. This means that the resulting system must be equipped with behavioural properties (see Figure 6). Similarly the system model can have corresponding behavioural attributes if, for instance, simulation of the system's behaviour is required.

In synthetic modelling, however, the functional behaviour of the system must be transformed to factual attributes. Hence, each function is characterised and represented by a set of factual attributes, which can be used in connection with specification

and test of requirements. In the synthetic model, a component must be created in order to represent the system and the necessary attributes (see Figure 8).

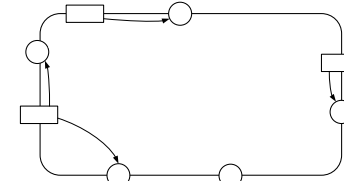


Figure 8 – Synthetic model on top level of abstraction – behavioural attributes transformed to factual attributes

Creation of model components is performed by selection among the component types, which must be available in taxonomies. Which type to select depends on what attributes are required.

Finally, internal/hidden attributes and relations of different kinds must be modelled.

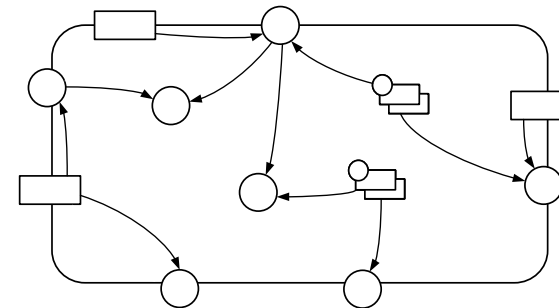


Figure 9 – Synthetic description of product family model including internal attributes and relations

## References

[Bertalanffy 1967]

Ludwig von Bertalanffy: *General Theory of Systems - Application to Psychology*. Social Science Information, Vol. 6, 1967, pp.125-136.

[Boulding 1956]

K. Boulding: *General Systems Theory - The Skeleton of Science*. General Systems Vol.1, 1956, p.11.

[Hammer 1978]

Michael Hammer and Dennis McLeod: *The Semantic Data Model: A Modelling Mechanism for Data Base Applications*. Proceedings of ACM/SIGMOD International Conference on Management of Data. Austin Texas, pp.144-156, 1978.

[Rosch 1978]

Eleanor Rosch: *Principles of Categorisation*. In: Cognition and Categorization. Laurence Erlbaum, Hillsdale, Jew Jersey, 1978.

[Smith 1977a]

J. M. Smith, D. C. P. Smith: *Database Abstractions: Aggregation*. Communications of the ACM vol.20, no.6. pp.405-413 New York 1977.

[Smith 1977b]

J. M. Smith, D. C. P. Smith: *Database Abstractions: Aggregation and Generalization*. ACM transactions on Data Base Systems, vol.2, no.2. pp.105-133 New York 1977.

[Sowa 1984]

John F. Sowa: *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, 1984.

## Appendix B – Modelling with ArchiCAD (AC)

ArchiCAD is a 3D building modelling tool produced by Graphisoft Hungary.

The building model in AC (called the Virtual Building by Graphisoft) is stored in one model file and as such is characterised as a database implementation. Every item is described only once. Document specifications and other data are included. Changes are made to each item in only one place.

Thus, every user of the repository can be certain that what they are seeing is exactly the same information that every other project participant sees. This one change to the way project documentation is handled can greatly reduce the number of communication problems that slow down projects and increase costs.

Based upon objects, quantity surveys, bill of materials, window and door schedules can be prepared automatically.

### ***Line of action using AC***

The very first step in a modelling process is to decide about the structure of the virtual model in relation to its stories. This implies defining the number of stories as well as the different floor heights of the building. In AC each object is attached to a storey.

The next step in modelling process is to decide about the layer structure of the building model. It is important to place the many objects of the model like walls, slabs roofs etc at their relevant layers. One purpose for this is to be able to isolate parts of the building during modelling by switching on and off separate layers.

The third important issue in modelling is to think about the virtual model as about a real building. It is logical to start making the virtual model bottom up starting with its foundation, then the ground floor, first floor etc. The modelling then becomes similar to the way the building is made in real.

### ***Space Modelling***

AC uses the term zone instead of spaces, and interior space modelling takes place with the Zone tool. It is possible to start with space modelling before construction modelling, but space modelling with the Zone tool is limited in terms of sketching, modelling and editing. Spaces with the Zone tool has to be made in 2D plan. Later walls around the spaces can be automatically created by using the Magic Wand tool.

Because of the lack of a comprehensive space modelling tool Construction modelling normally takes place before space modelling, placing walls, slabs and roofs. Then the Zone tool is used to automatically make the spaces. Space modelling is about surfaces, lightning, fixtures furniture etc. The walls as the space perimeter are the basis of automatic space generation. Spaces can also be drawn manually with the Zone tool.

However once the spaces have been made they can be shown in 3D in a satisfactory way. Information about the sizes of each single space as well as total space information can be shown on lists to be printed out.

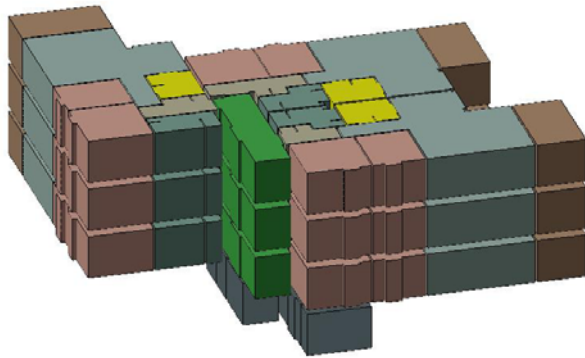


Figure 1: 3D interior spaces.

## ***Construction Modelling***

AC has a number of construction modelling tools. There are several general object tools like the Wall tool, the Slab tool, the Beam tool, the Roof tool etc. They are designed for their specific purpose like modelling a Wall, a Slab etc. These tools are parametric and contain material properties and are by default placed on specific layers. Then there are Window, Door and Opening tools which can only be used as part of a Wall. They are also parametric and contain material properties. Then there are specific objects to choose between such as example furniture, lamps etc.

Objects like windows, doors, openings, furniture etc are all GDL objects. GDL means Geometric Description Language. GDL is an open programming language in AC based on visual basic. As such also new specific GDL objects not found amongst the mentioned construction tools, can be developed either through programming or created as GDL objects automatically. Virtually any type of shape and form can be created in AC by the use of GDL.

Through the construction modelling process the virtual building model will undergo several model stages from the very first conceptual stage of a very low detailed model further to the final and very detailed stage of the model. In early stages very simple Wall Window Door, Slab and Roof objects will be used. Later these objects will be more detailed by making use of for example correct shaped and detailed iron beams like HEB profiles.

During this process several reports may be drawn from the model. Such reports will include reports for visualization, quantity calculations, 2D drawings etc.

During the construction modelling process it is important to be able to move up and down between stories easily. Because the AC model is a comprehensive all in one model it is possible by a single click to move between stories up and down. This feature is very handy and useful though the modelling process.

## ***Particular possibilities with AC***

AC has a number of tools to make use of in the modelling, editing and visualization process.

### *Tools to cut and isolate objects.*

Through the modelling process it is very important to be able to look at smaller parts of the model at any time for analysis. Very soon in the modelling process any model consist of a vast number of objects such as walls, doors, windows, slabs etc. In Wireframe it is often impossible to find a specific object for editing. Therefore tools for cutaway and isolation of objects are extremely useful tools.

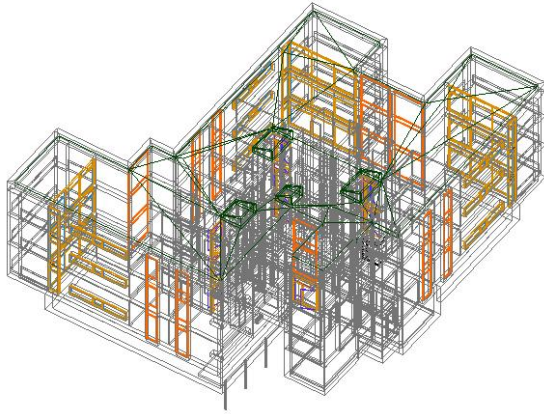


Figure 2. Wireframe Presentation

AC has several tools for that purpose. One simple tool is the Layer system. That is a common tool in most CAD software. One simple method is to place objects on individual Layers and then switch off and on layers. Further in AC the use of Layer Combinations is an extra and important part of using layers. By combining selected layers in various combinations it is possible to show parts of the virtual building for editing and presentation purposes.

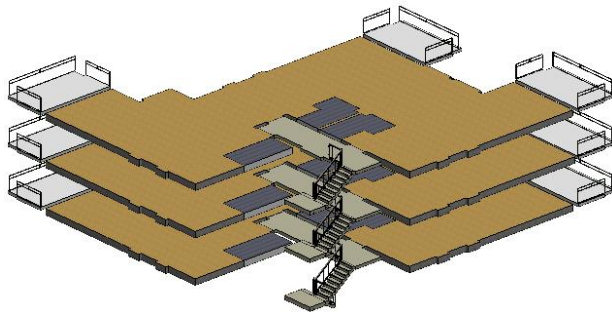


Figure 3. Layer Combination example

Another very important tool is the marquee. This is the far most exceptional and useful tool specifically found in ArchiCAD. Using the marquee makes it possible to cut any part out 3 dimensionally for further examinations. The cut out can then be turned around in 3D for examination viewing it from various sides, zoom in and out. The result is a fine grasp of how the building parts in 3D interacts with one another. The marquee tool can be set to cut through all stories or just one storey.

Isolate parts of the building by using "Elements to show in 3D" is another important tool in AC. By using this it is possible to control which types of objects to be shown in a view. It is also possible to decide if this should take place for one, two or all stories. Object types to switch off and on are Wall, Beam, Column, Window, Door, Object, Lamp, Slab, Mesh and Roof. This can be used for editing as well as for visualization purposes.

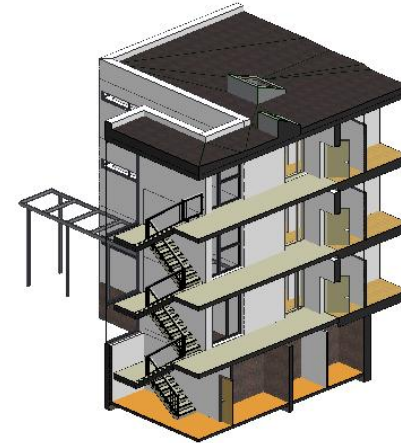


Figure 4. The marquee tool – cut through all stories.



Figure 5. 3D elements shown in 3D – Walls not shown – two stories.

The tool 3D Cutaway is yet another use full tool. Drawing a cut line makes it possible to view objects either under or above the cut line.



Figure 6: 3D Cutaway

These described tools, Layer and Layer Combinations, the Marquee, Elements to show in 3D and 3D Cutaway can also work in combination with each other. This of course implies extremely unique and strong possibilities to completely decide and control which objects and what parts of the building model is to be edited, viewed and visualized.

## Creating Reports from the Model

The virtual 3D building model is the base of the project, the place where any changes to a project are to be made. A number of different reports can be extracted from that model for example 2D drawings, bills of quantities, schedules etc

## 2D drawings

For some time still to come production of ordinary 2D drawings will still be necessary in the building production process. 2D drawings are extracted and unlinked from the 3D model and the quality of these drawings varies due to the model stage and requirements of the drawings.

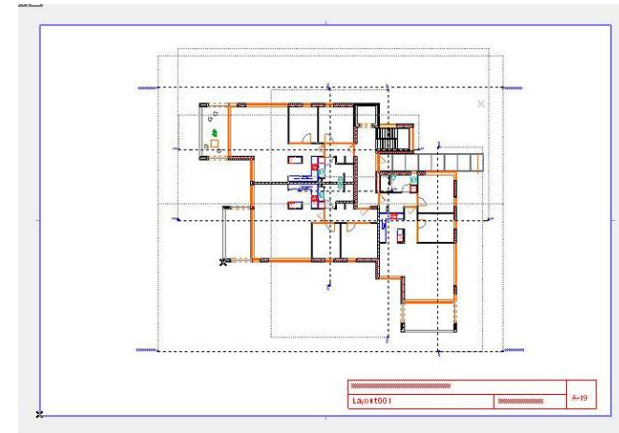


Figure 7: Drawing extracted from the 3D Model

Most often extracted drawings have to be detailed further by using 2D tools. Often a floor plan view will be acceptable as a 2D working drawing without any further detailing. In particular in scale 1:100 and 1:50 where as for example section drawings most often have to be further detailed.

Sections and elevations are extracted out of the model through the use of the section / elevation tool. The quality and detailing is controlled by the detail of the model as well as through the Display system. The Display system will control the individual lines and hatching in the drawing. Elevations and Sections can further be displayed with shadow casting solid or hatched shadows.

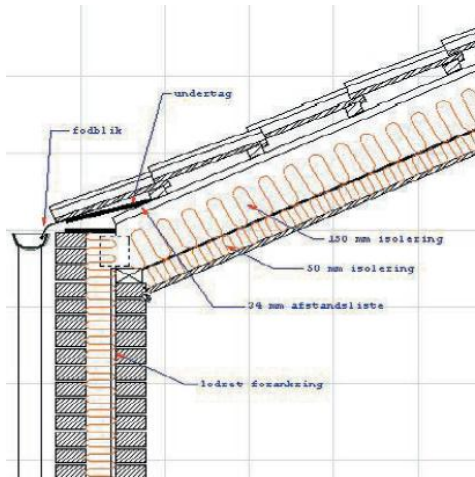


Figure 8: Detail tool used

Details and special effects can be added to a section or an elevation. This often applies to selected parts shown in scales like 1:5, 1:2 by making use of the Detail tool before they reach the level of information necessary in the building production process.

Production of 2D Drawings is laid out in a separate application PlotMaker which is linked to AC.

## Particular issues

A few topics in AC causes problems in particular Cavity Closure and Wall Ends. Also composites, slabs and flat roofs causes some problems.

### Cavity Closure

When windows, doors and openings are inserted into a wall the wall parts to accommodate it. When walls are displayed in detail showing its components (composite) the wall segments around window, door and opening is made by using Cavity Closure.

Cavity Closure is defined as part a window, door and opening object. As such the design of cavity closure is pre-designed and with no possibility for individual design solutions.

Cavity closure therefore is a kind of label rather than a real designed detail around wall and window. The choice of design is often a choice of three such as None, Turn in, Turn out or Continuous. In other situations the choice falls between None, traditional or prefabricated.



Figure 9. 3 Choices of Cavity Closure in a particular window.

Yet in other situations the possibility of choices is slightly more complex. Cavity Closure therefore has to be accepted as an abstract label depending on specific chosen window, door and opening type.



Figure 10 – Illustration of Standard Cavity Closure



Figure 11. 3 More choices of Cavity Closure in this particular window

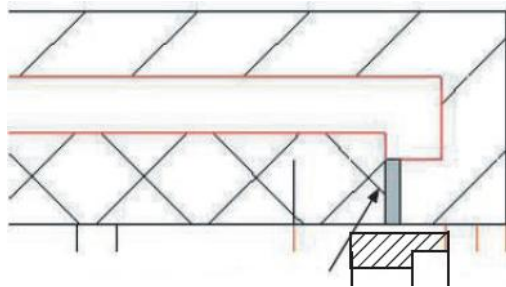


Figure 12: Cavity Closure

These problems about Cavity Closure would lead to the conclusion that perhaps Cavity Closure should rather relate to the wall and not to the window, door or opening objects. This would open for more individual design solutions related to how the Wall is composed.

### Special Designed Windows and Doors

The AC package contains a number of standard pre-designed generic window and door objects. In the early stages of the modelling process these window and door objects full fill the requirements. However, later in the modelling process, there is sometimes a need for making use of special designed window and door objects.

One solution to this is when possible to download the specific window and door objects from a manufacturer's homepage. The alternative solution is to program window and door objects through the use of the in built GDL programming tool. This cannot be done easily and requires some programming skills.

### Wall Ends.

Another similar problem to cavity closure is how to design wall ends. The specific Wall End tool is slightly more advanced than is the case of cavity closure. Often wall ends are simpler to make when making use of ordinary wall solutions. The wall end tool dialog gives some choices.

With more wall building the wall end tool is simple and the need for individual design possibilities arises.

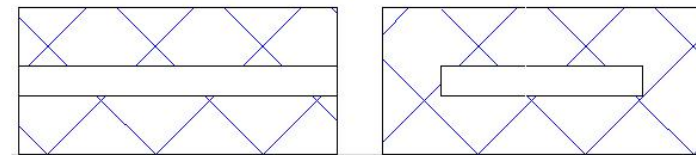


Figure 13: Simple Wall end solution done directly by the tool



### *Design regarding Cavity Closure and Wall Ends*

Special design of Cavity Closure and Wall Ends are done by making use of either the Patch Tool or the Detail Tool.

A patch is placed on top of the area where a window, door or opening meets the wall and on wall ends. Using the Patch tool makes it possible for individual design of the connections between window and wall and at wall ends.

### *Wall intersections*

The way walls are intersected is defined by their intersection / justification lines. A wall of the same type is precisely connected when these lines meet precisely or intersect. This works perfect with simple connections of walls placed at the same Z level and with the same wall heights. When more walls are placed in different levels with different widths and different heights wall intersections causes problems.

The same problem is also seen when walls with justification lines not placed in the same wall side meets.

### *Wall Composites*

A Wall construction consists of a number of separate material layers. An example of a wall consisting of 3 parts is an exterior walls of 328 mm. On the outside is a brick part of width of 108 mm, on the inside a concrete element part of a width of 120 mm and in between an insulation part of 100 mm.

This construction is to be displayed on plan as well as on section drawings. In AC composites are used for this purpose and as such composites are fine.

However composites do not divide the wall into three separate wall component objects. Composites are graphical displays.

This causes problems in situations where it is desired to dislocate / offset the concrete element part of the wall. This will for example be the case where the exterior wall meets the floor slab for correct connections.

With graphical display it is only possible to adjust heights on the total wall.

### *Slab composites*

A similar situation is seen with slab composites. As an example a normal floor constructions consists of 220 mm thick concrete slabs, 50 mm insulation, 100 mm concrete and 20 mm wooden floor a construction consisting of four parts.

Again the composite tool in AC works fine and shows the composite construction graphically well. Since the floor construction is not divided into four separate construction objects there is no possibility for dislocation of parts of the floor.

### *Roof composites – flat roof*

The same applies to roof constructions using roof composites. In the case of a flat roof the slab tool is to be used with the problems just described above.

Often a flat roof has to accommodate for leading rainwater to pipes the rooftop is not flat. The roof construction therefore has to be made by another tool than the slab tool. The Mesh tool is a choice that can be used.

The Mesh tool cannot make use of composites. Therefore the way of solving is to use two individual geometric elements. First

a slab consisting of a number of parts displayed with composites and, on top of that, a separate Mesh object with slopes.

### *Free standing Windows and Doors*

Most often it is fine that windows and doors can only be inserted into walls. But some times it would be nice if it was possible to place windows and doors free standing without having to insert them into a wall.



Figure 14: Window in Wall – Window replaces wall

A free standing window or door can only be made if the Wall and the window placed in that wall have the same dimensions and therefore completely replaces the wall. (se figure 14)

### *Development of individual Doors and Windows*

As long as making use of the windows and doors that come with AC is acceptable it is fine. But if it is not possible to download these objects GDL programming is the only alternative.

GDL objects can simply be made automatically (not parametric). This way is not very complex. More complex window and door objects have to be made in GDL programming. This however will require some programming skills.

### *Layout and Print of 2D drawings*

Drawings in AC are produced automatically as views, further detailed and then grouped in View Sets. Layout and print does not take place in AC but in a the separate application PlotMaker. The idea is to make editing and changes in AC and then only use PlotMaker for layout and print. This is slightly complicated since often some editing has to be done in PlotMaker in terms of line widths, colours etc. The way it works appears to be complex and cumbersome.

### ***IFC export / import in AC***

IFC is fully integrated in the AC structure. IFC is an add on module which can be downloaded and installed in AC. As such IFC is very consistent and easy to work with.

### ***Conclusion***

AC is an easy Building modelling tool to work with all stored in one model file safeguarding that changes made in the model will be reflected in all extracted reports.

Many splendid tools are available for construction modelling for analyzing, editing and visualizing purposes through the modelling process. The marquee tool in particular and specific to AC is very significant. As a whole the possibilities of individual combinations of tools like layer, layer combinations,

cutaways and isolation of object and parts of the building model is fine.

Space Modelling cannot be done in a desired way. As a substitute the Zone tool is usable if it is used after construction modelling has been done. Sometimes there is a need to model spaces first. This can be done with the Zone tool however very stiff. In particular it is fine that spaces can be shown in 3D.

The way cavity closure is made is problematic without any individual design possibilities. Composites are fine for graphical display but are not more than a label. There is a need for real wall, slab and roof components for individual changes and design.

Extracting 2D drawings from the AC model works fine. Further detailing of 2D drawings is mostly needed to be done on section drawings whereas less detailing has to be done on floor plan drawings.

The layout and printing facilities through an external application PlotMaker linked to AC is a bit cumbersome.

Due to direct integration of IFC in AC it is very easy to use IFC. Saving an AC file to IFC is easy and works perfect. The same applies to importing an IFC file into AC. Only minor problems arise through these operations. The problems of IFC exchange is often due to the fact that other software not being fully version compatible or fully developed for IFC.

## Appendix C – Modelling with Architectural Desktop (ADT)

ADT is a 3D modelling tool produced by Autodesk USA. ADT is built upon AutoCAD, which means that all AutoCAD tools are found in ADT. It is possible to work with 2D AutoCAD tools in ADT at the same time as working with ADT 3D modelling tools.

ADT is not a compact model but is characterised as a file system implementation based on an external reference system where multiple separate files are linked to each other. The project is termed Building information model (BIM) by Autodesk and consists of two parts, the 3D model itself and the reports extracted directly from the 3d model.

The 3D model consists of a number of Constructs and Elements. The reports consist of a number of Views and Sheets. Within a project Constructs and Elements are X ref`ed into Views that are again X ref`ed into Sheets.

A Project is described in an XML file for each dwg file. The XML file contain relevant information about a project such as level and division information.

All objects in the drawing file are parametric and their data information can be calculated and placed in tables.

As such the 3D ADT model is not a unified model but a model consisting if several separate drawing files, which are overlaid or attached to each other.

### ***Line of action using ADT***

Prior to modelling all requirements about functions of the building has been taken (function modelling) and are available as specification of threshold values of the buildings performance. Considerations about space modelling and

construction modelling components may take place in relation to the ADT software file structure and performance.

After creation of the new project the first thing to do is to decide about how the building model should be structured. First decisions are about the number of levels (stories) and heights of levels have to be taken. Then decision about divisions also has to take place when working on large projects. Dividing projects into smaller parts will speed up processing time.

Next step is to start modelling constructs and elements as a number of files automatically organised in different folders.

Later reports are to be drawn from views and sheets. A view is consisting of a selected number of construct and element files. A view typically will be collected as one or two construct files or of construct drawing files. As such a view can be a collection of construction files representing part of the building model or the building model as a whole.

Finally sheets are made for placements of different selected views for printing and publishing.

The views are either used for output in IFC format or placement on sheets. Sheets are either used for paper prints or internet publishing purposes.

The project navigator is the central place to control the many different parts divided into four tabs.

The first is called the Project tab. Here all information about the project is gathered. The next tab in the navigator is called the Constructs tab containing all separate files of the project.

The third tab is called the View tab where a chosen collection of X refs (drawing files) is placed to produce desired Views.

The last tab is the Sheets tab in the Navigator where all Views are placed on sheets.

## Space Modelling

It is possible to start with space modelling at each different level (story) and later designing the construction parts like walls, slabs and roofs around the spaces but placed on separate drawings. It is also possible to do the construction modelling first and then generate space afterwards bases op the constructions.

Interior space modelling works fine in ADT. Space modelling is one side of integral modelling and can take place in the very early stage of the modelling process. Space modelling is about surfaces, lightning, fixtures furniture etc.

Space modelling takes place in 2D. The space modelling tools is flexible and fine in terms of sketching, editing and modelling of spaces. Visualization of spaces also shows best in 2D. Space modelling in 3D is very simple only showing the floor and the ceiling of a space. As such it is not fine for 3D visualization purposes. In Figure 2 it is shown how the result of modelling one single space looks like in 3D.

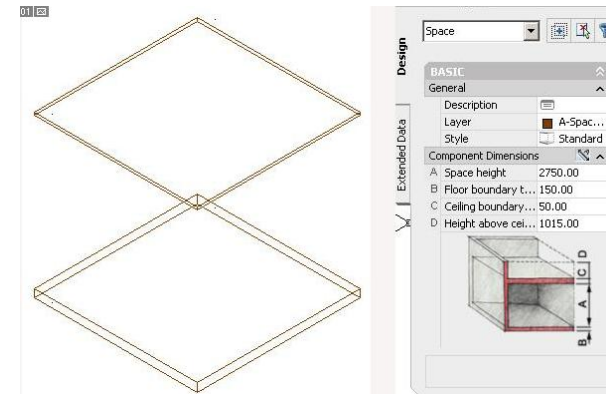


Figure 1: Single space and space dialog

From the very early start of the project interior space modelling will take place on drawings placed in the construction tab as described earlier. Spaces can be put together, cut and assembled in a sketch like way until the desired spaces are developed. Then the construction elements like walls, slabs and roofs can be placed around the space structures.

Space modelling gives instant information about the sizes of the interior spaces individually and as a total. This information together with surface information can be presented in printed lists. Spaces are defined by their types with individual type

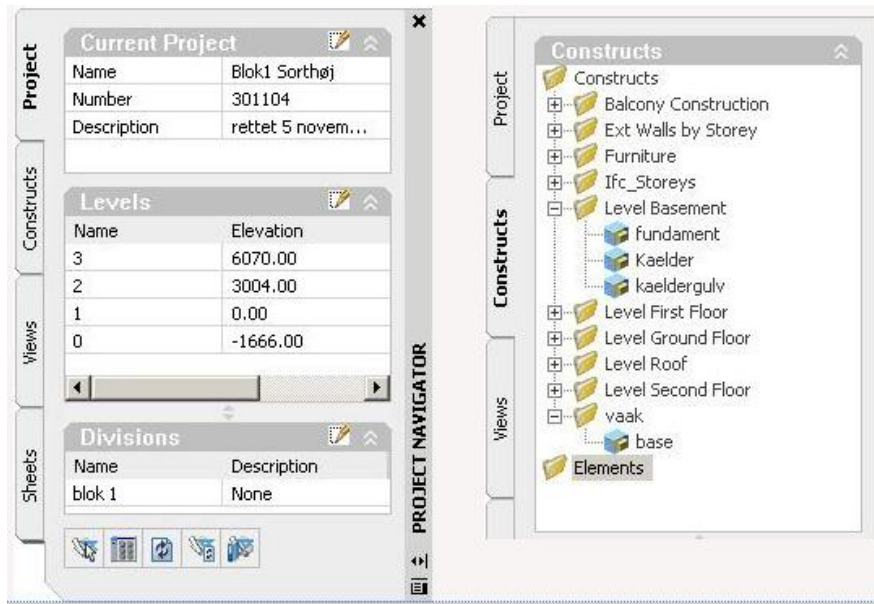


Figure 2: Project, Level, Division info and structure of constructs

## Construction modelling

There are several general object tools like the Wall tool, the Slab tool, the Beam tool, the Roof tool etc to be used for construction modelling. They are designed for their specific purposes like modelling a wall, a slab etc. These tools are parametric, contain material properties and are placed on layers. The Window, Door and Opening tools are designed to be used as a part of a wall or placed independently. Windows and doors are parametric objects, containing material properties and are placed on layers.

Then construction modelling takes place on each of these separate files. Often a construction file consists of a few construction objects as considered most appropriate, like just exterior walls, windows and doors.

Each construction file has a number of layers representing each construction object with the possibility to switch on and off each layer. Each layer controls Colour, Lineweight, Linetype Plot Style. It is only when a selected number of drawing files are connected into a view the building model as a whole will appear.



Figure 3: Spaces shown in plan view.

## Particular possibilities with ADT

Through the modelling process the virtual building model is going through several model stages from the very first conceptual stage of a very low detailed model further to a detailed stage of the model. In order to do so it is important that the modelling tool used is easy to use, to analyse and view the model, to edit the model, to change parts of the model with other more detailed elements and to draw information directly from the model such as area and volume information and reports for visualization, quantity calculations, 2D drawings etc through the various modelling stages.

In ADT parts of the model can only be isolated in a very few ways. Since each space, construct and element is placed on separate drawings it is only possible to edit in one separate drawing. This is because of the X`ref system.

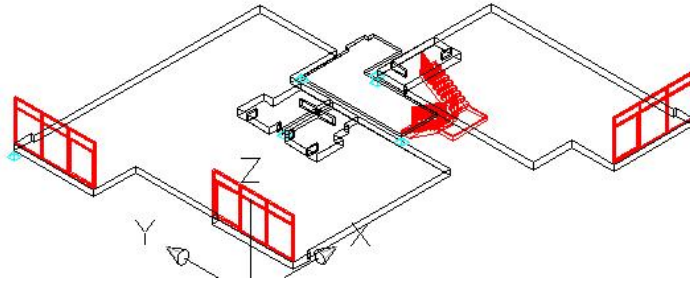


Figure 4: One construct with windows and stairs

If exterior walls are placed on one construct and interior walls on another then editing can only take place on one these two separate drawings at a time. In ADT modelling and working takes place on only one drawing at a time. For visualization purposes it is possible to drag and drop any other drawing as an X`ref into that drawing. However since this is an X`ref system it is only possible to edit in the drawing where the objects are created.

The X`ref system is very fine for visualization purposes. It is possible to decide which and how many drawings should be X`refed into that visualization. Using this method will reduce the amount of data speeding up rendering.

Layers can also be isolated by switch off and on or isolated through a filter group system.

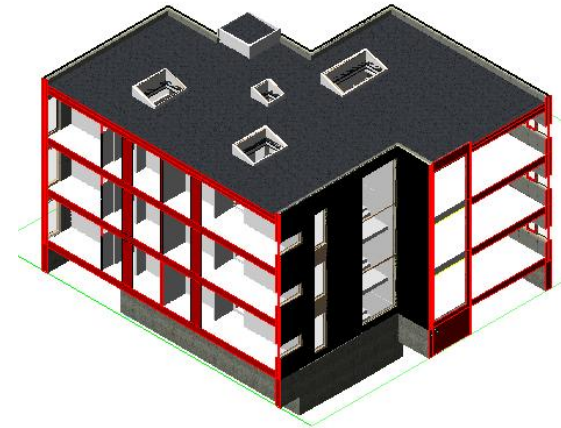
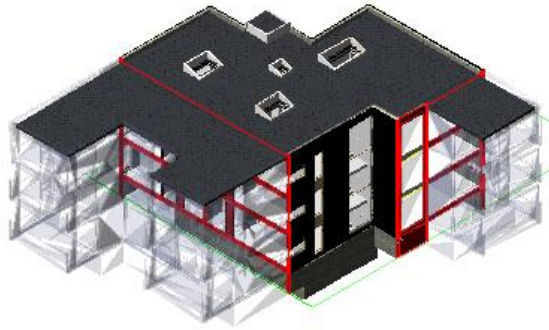


Figure 4: Live Section

Another way of isolating objects in a drawing in the modelling process is to make use of the Model Viewer. By selecting one or more objects these will be shown separately in the Object Viewer. The viewer is not useable for editing but only for visualization on the screen.

By using either the horizontal and vertical section tool or a combination of the two it is possible to make cuts in the Model in separate smaller pieces in order to look at cut parts of a model. This is done with Live Section. A cut in a building model however is a cut through a number of X`refs which makes editing impossible. For visualization purposes it is fine.



*Figure 6: Transparency in cuts – Live section*

The conclusion is that editing possibilities in ADT in the modelling process is limited to editing in each single X`ref drawing is poor whereas combining selected drawings, isolating objects using the object viewer and Live section cuts are fine for visualization purposes.

### *The wall components*

In modelling detailing of the modelling components are of importance. An exterior wall is one such component. As an example an exterior wall may consist of three parts. an outside brick part 108 mm wide, a concrete element 120 mm wide and in between an isolation part 100 mm wide.

The wall is built up by its three components. In ADT these components are individual geometric parts which can be dislocated individually. Each part can have individual heights. As such it is possible in ADT to make a correct connection detail where the outer wall meets the slabs. In plan and section views the wall detailing will be shown fine.



*Figure 7: Wall Offset of concrete part and insulation*

### *Free standing Windows and Doors*

Now and then it is important that windows and doors can be inserted directly in walls as well as individually without having a wall to place them in. ADT includes the possibility to choose either of them. As an example a window assembly can be placed directly on a slab without having to be part of a wall or relate to a wall.



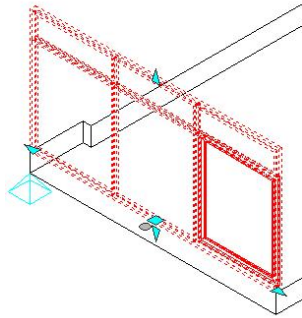


Figure 8. Window separate to wall

### Endcaps

Another important function of a modelling tool is its ability to individual design of end caps on wall ends and around windows, doors and openings. There are two types of endcaps in ADT, wall end caps and opening end caps. Wall end caps are used at wall ends and opening end caps are used where windows, doors and openings meet the wall. In ADT End caps can easily be designed individually and changed anytime through edit in place. End caps are part walls not of the window, door or opening object which makes them very easy to design individually.

### Creating reports from the model

The 3D building model is the base of the project, the place where any changes to a project are to be made. A number of different reports can be extracted from that model for example 2D drawings, 3D Live sections, calculations of areas and volumes, schedules etc.

### 2D Drawings

2D section and elevation drawings are made within the Navigators Views Tab. A chosen collection of drawings is X`ref`d into a 3D model which again is used by the section / elevation tool to produce 2D drawings. The quality of these drawings is fairly low depending on the detailing of the model. Therefore the 2D drawing usually have to be further refined by using ordinary 2D AutoCAD tools for adding Text, Lines and change of Lineweights and hatching etc.

Detailed drawings for example in scale 1:5 or 1:2 has to be further detailed by using tools from the Detail Component Manager for automatic detailing or by using traditional manual AutoCAD 2D tools.

Since there is no automatic way to do this the problems arises when changes are made in the model and new sections and elevations are drawn from the model.

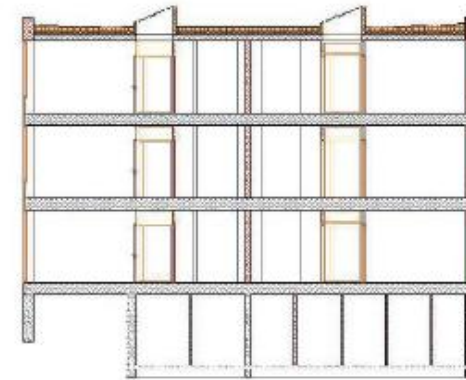


Figure 9: Example of section drawing generated directly from the model

### ***Particular issues***

ADT appears to be very complex with its structure of constructs and elements, views and sheets all as a number of separate files being put together into the complex X ref system. As such it is not an easy modelling system.

### ***The slab components***

A wall construction consists of a number of separate parts. An example of a wall consisting of 3 parts is an exterior walls of 328 mm. On the outside is a brick part of width of 108 mm, on the inside a concrete element part of a width of 120 mm and in between an insulation part of 100 mm.

It is desirable if the slab tool had a component tool with an offset possibility like is the case of walls. Then it would be possible to model a floor construction geometrically precisely how it is in order to make the slab and wall meet correctly. Figure 9 shows in an ordinary 2D drawing how this should look like.

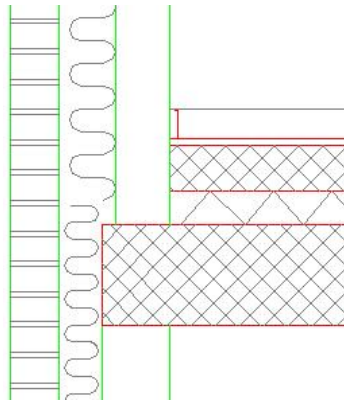


Figure 10. Slab detail in ordinary 2D drawing

### ***The Roof Components – Flat Roof***

Whether it is a sloped or a flat roof, there is no possibility of a component tool. An example of a flat roof with two components is a concrete slab element and an insulation element above. The Roof construction tool offers no component tool and as such it is not possible to model the roof geometrically correct or show the roof details graphically. Modelling a flat roof is then done by modelling each of the two components individually. The top insulation part has to be modelled in some strange software in order to make correct slopes.

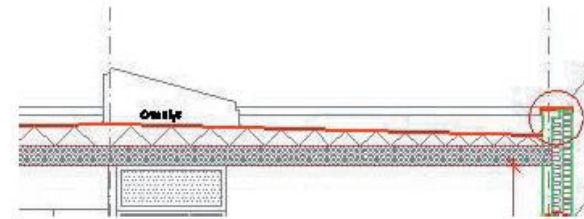


Figure 11: Roof components – hand drawn presentation

### ***Corner windows***

There is no specific designed corner window tool to be used in ADT. Therefore making corner windows is complicated in ADT. To make corner windows the first thing is to make a hole in the corner wall and then place two separate windows in that hole.

There are two ways to making holes in a wall. The first way is to cut the hole with the opening tool and use it on both of the two walls forming the corner. The second way is to cut holes by using mass elements and subtract these from the walls. This second solution may cause problems later with the IFC file.

## *Stairs*

The stair tool can be difficult to work with. The user interface is not good enough and an easier tool to work with including more grips and possibilities to create new stairs would be fine.

## *Layout and Print of Drawings*

Finally the 2D drawings have to be layed out and printed. This is done from the 2D drawings produced in the views tab by placing these views on the layout sheets shown in paper space. The layouts / Drawings may contain several drawings. PlotStyles (Colour dependent or Named) for good presentations of the prints.

Editing of lines and line colour is not easy within this complex X`Ref system. It is not possible to edit line weight and colour at Sheet tab level. One has to go back to Construct tab level to make the changes which seems to be a bit complicated.

## ***IFC export and import in ADT***

Since IFC is not integrated in ADT, IFC exists as a separate add-on module produced by a third party vendor Inopso in Germany. The present IFC version is 2x for ADT 2005. The fact that IFC is produced outside ADT gives rise to several problems.

The first one is that the IFC version to be used in ADT always developed later among others because it is developed outside Autodesk and because of the complexity of the ADT structure and X`ref system.

IFC 2x2 is not fully integrated i ADT which makes it more difficult to work with. It has to be installed and the software has to be purchased at extra costs.

There is not well documented manual to follow on how to use it properly. Therefore after installation time was spent finding out how to export an ADT model to IFC with the correct IFC containment structure consisting of Project, Site, Building and Storey.

It also seems as if the developers of the IFC add-on module has worked mostly on the IFC export and less on the IFC import. Exporting an ADT file to IFC is a rather complicated process. The conclusion to that is that an optimal IFC solution does not exist for ADT.

## ***Conclusion***

ADT is a complex building modelling tool to work with due to its many single drawing files put together through its complex X`ref system. As such the model is split up in a large number of files and therefore not compact.

In the modelling process selection of smaller parts for visualisation is limited only using the X`ref system. For editing this can only be made on the file where the objects were created. The advantage with the X`ref system is that by X`ref`ing only the necessary files the computer processing will work faster.

The endcap system with individual design possibilities is very usefull.

The offset for wall components works fine and gives good control of the assembling of wall and slab. There appears to be a number of problems in modelling components detailed with slabs and roofs.

Corner windows cannot be made easy since there is not such tool for this.

2D drawings are easily extracted from a model consisting of chosen number of X`refs but needs quite some further detailing using AutoCAD tools.

Space modelling takes place only in 2D and works as such fine and is designed for the sketching editing and modelling process. Space modelling can take place before or after construction modelling. 3D display of spaces can take place but is very poor.

Since IFC is not integrated into ADT but is a separate add-on module, it is complicated to work with. It is mostly developed for export functions.

## Appendix D – Comparison between ADT and AC

In order to make comparisons between the two used program software AC and ADT, the most important issues in modelling with the two has been pointed out.

Both types of software is basically for building modelling in 3D. On the basis of the 3D model they are used for making reports in the form of 2D drawings for building production, for visualization purposes and for IFC use in relation with simulation software etc. From that structure several issues has been listed to be used as headings for the comparisons.

### ***Modelling***

#### Software structure

There are great differences between AC and ADT in terms of software structure. The building model in AC is stored in one model file and as such characterised as a database implementation. Everything is placed in one compact file.

ADT can be characterised as a file system implementation based on an external reference system where multiple separate files are linked to each other.

#### Database – intelligence

Any object in each of the two software systems has some intelligence. Each object “knows” that it is a door, a window, a wall etc and as such each is able to carry data about themselves to be used in calculations of quantities.

### ***Parametric objects***

Both software packages are based on the use of parametric objects.

### ***Stories***

Both software packages are based on 3D models working with stories. In AC the name is Storey whereas the name in ADT is Level. Although the concept is almost the same moving between stories / levels takes place in very different ways.

Moving up and down between stories in AC is easy don through Pop-Up icon or directly in the Navigator. This is interactive and can be termed as a live moving between stories.

Moving between Levels in AC due to the X`ref system can only take place by click on separate drawing files placed in the navigators construction tabs. The move can be termed as static.

### ***Analysis and editing during modelling***

An important function in building modelling is the ability to isolate objects for modelling, editing and visualization purposes through various tools.

AC has a number of advanced tools such as the use of Layers and Layer Combinations, the Marquee, Elements to show in 3D and 3D Cutaway as well as various combinations of them.

In ADT because of its X`ref system the isolation of objects can take place by showing one or more drawing files, using the

Model Viewer, switching on an off layers and using the vertical and horizontal section tool.

### *Modelling Support*

When modelling is performed, the programs offer various support.

#### Making New objects

Often the software packages do not contain any kind of special objects to be used in a model. This can for example be window, door, stair objects etc. It is therefore often necessary in building modelling to be able to make new objects in an easy way.

AC has specific possibilities for making new objects through programming in GDL. GDL (Geometric Description Language) is a programming tool included in AC. It is open for programming use and virtually any shape and form can be scripted in GDL.

ADT is closed for programming of new objects. It is possible to make GDL objects in AC and through a GDL add on module named Object Adaptor installed in ADT also to make use of GDL objects in ADT. However the GDL Object Adopter is not a well developed add-on. Problems with colour translation and transparency in glass have not been solved properly.

#### The use of layers

The use of Layers in building modelling plays an important part and the use of layers is a suitable way to structure objects and with the possibility of switching on and off layers.

Also layers are important carrying information about objects colour, lineweights etc is important. In AC, layers are related to object types. Walls will by default be placed on the Exterior Layer controlled by a specific template. Each layer can be

locked, switched on or off and all objects on that layer can be shown in wireframe or shaded. There is no control of linetypes or colour related to layers in AC. Most often a layer contains only one object type. In AC layers can be combined in layer combinations.

In ADT layers can also be locked, switched on and off. I can also be frozen. The big difference here is that colour, Linetype, Lineweights and PlotStyle is also part of the layer, Often one layer contain several object types. In AC layers can be organised in layer and filter groups.

#### Snap functions

In order to be able to model accurate and precisely the snap functions of a software becomes of great importance. In AC snap work in relation to Grid, to other objects and to hotspots. This is very efficient.

In ADT snap works also in relation to grid, to individual objects and points within an advanced OSnap system. Snap works very precise and perfect in ADT.

#### Navigation

Building modelling is in AC and ADT controlled by a Navigator.

The Navigator in AC is built as a control system in relation to its database implementation system all in one file. As such the navigator contains controls for the whole model and model levels and related to model parts in the building model.

The Navigator in ADT is based on the system implementation as an X ref system. The Navigator is structured in relation to separate and linked files placed in different tabs such as Constructs, Views and Sheets.

## Display

It is of great importance how a model and working drawings from the model looks like. It has to do with how easy the model and the drawings is to read. A good display system controls how the model and drawings are displayed.

In AC the Display system is very simple and modest. It is very easy to understand and to operate. The Display system has to be adjusted in one place and as such very easy to learn. In ADT the Display system is extremely complex and not easily understandable. The Display system has to be changed and adjusted many places and as such very difficult to learn.

## *Modelling tools*

Each software package has a vast number of modelling tools. To mention some of the most used ones:

Wall, Slab, Roof, Window – Door, Column, Stairs

### The Wall tool

This tool has always been considered the most important. As an object a wall is a parametric object and as such it can be changes and edited in many ways.

ADT has very strong features to manipulate walls. They can be stretched by the mouse or by keyboard in all directions. Walls can be given different shapes and different shapes and forms can be added or subtracted to other walls. Walls can also be divided into their components shown and manipulated geometrically. When placing windows, doors and openings in a wall, the wall parts to accommodate it. Since it is possible to make changes to the components geometrically the solution where exterior wall and the slab meets can be made perfectly.

In AC these possibilities is a little more constricted. A wall can be stretched by the mouse or by the keyboard, it can be changed in width and height through the wall dialog. Walls can be divided into its components through the composite tool. The divisions are made only graphical. This gives some problems around the detail where a slab meets the exterior wall and the solution can not be made perfectly.

Important to walls is also how well they are to cleanup wall intersections. Wall intersections are built upon a walls justification line. In ADT wall cleanups are controlled by their type of wall cleanup group definition and the priority of the different component the wall comprises of. In AC wall intersections are controlled through the composite tool deciding on which component is the most important core component.

Basically the two software packages works very much the same in relation to wall clean up.

### Slabs

IN AC it is possible to divide a slab into its parts – composites. Composites are not geometrical components but just a way to show the way the slab is divided graphically.

In ADT it is not possible to divide a slab into its consistent components.

### Roofs - sloped

IN AC it is possible to divide a the Roof into its parts – composites. Composites are not geometrical components but just a way to show the way the slab is divided graphically.

In ADT it is not possible to divide a Roof into its consistent components.

### Roofs - flat

Roofs are made to be shown in degrees between 1 and 89 in AC as well as in ADT. Flat Roofs therefore have to be made with the Slab tool. In relation to division of a Roof slabs composing part it works as described above.

Therefore flat roofs in AC as well as ADT has to be made consisting of two or more slabs. One slab is a flat concrete one below. On top of that is insulation part.

Mostly the top part has to be given a fall towards the water pipes. In AC the Mesh tool is a possible tool to use for that purpose. The Mesh tool is very easy to work with for such solution.

In ADT there is no such tool. Instead it is possible to make use of Massing elements. Massing objects are slightly more complicated to use for such specific purposes.

### Roof Skylights

In AC there is a number of pre-designed skylights to be found in the Object tool. Any of these may not be suitable for a particular purpose. In AC there are no such objects in the Design Center.

In AC this may be solved in two ways either during a GDL programming of a skylight. Skylights can also be designed with the Wall and Mesh tool and be generated automatically into a none parametric GDL object.

In ADT it is more complicated by using the mass elements.

### Windows and Doors

In general the generic windows and doors included in the software packages are suitable. However sometimes it is

necessary to design special window and door types In both types of software it is possible to design special window types in different ways.

Producing special windows and doors in AC has to be done in GDL programming. In ADT there is a "window and door engine" in the window assembly tool within special designed windows and doors can be made.

### Stairs

AC as well as in ADT has a stair tool to make generic stairs. In both types of software the tool is small module containing various stair parameters. The way stairs are designed in the two modules is very similar. The stairs designed are parametric and can easily be edited later.

In ADT it is possible to edit a stair in place whereas in AC stairs have to be edited in the stair module.

### Special building components

An important aspect in modelling in the final model stage is to make use of Product manufacturers building components. These components should be found on the manufacturers home pages for download. Very often it is not been possible to download such components because they are simply not found yet.

As an example a stair exactly as the one produced by the product manufacturer Dalton in Aarhus as well as an internal door produced by the product manufacturer Swedoor was developed in GDL. This was done in order to simulate a download from the internet.

These two building components was easily placed in AC as the native software for that. In ADT the special software called a GDL adaptor for stairs was used with success.



## **Reports**

Different reports can be drawn from the 3D building model such as 2D drawings, partial models for visualization, calculation etc.

### 2D drawings.

The quality of 2D drawings is of great interest. The intent often is to be able to produce 2D drawings for building production which looks similar to traditional hand drawn drawings. However digital 2D drawings made directly from computer models does not look exactly look the same and does not necessarily have to look the same.

However 2D drawings extracted directly from a 3D model will always require some extra 2D editing of the drawings. In AC this will be done using 2D tools and the detail tool. In ADT this will similarly be done by using AutoCAD tools and readymade objects from the Detail Component Manager.

### Elevations

They are easily drawn from the 3D model in both types of software.

In AC it is possible to place shadow and control shadow casting graphically in terms of choosing hatch type and the placement of the sun. These drawings will also require some editing yet not as much as with ADT.

In ADT elevations drawn directly from the model are very simple drawings with lines and some hatching. Further editing is required.

### 2D Sections and 3D live sections

In AC 2D sections are made by the use of the Section & Elevation Tool applied to the 3D building model. The 2D section

can be made with or without a link to the 3D model. When unlinked the drawing is separate a 2D drawing ready for further detailing using AC's 2D tools. For detailing in scales 1:100 and 1:50 only a little drawing and editing is being done. In scales 1:20, 1:5 and 1:2 the Detail Tool will be used for further detailing of particular parts of the section.

In ADT the horizontal and vertical section tool is used directly on the 3D model as presented in the Views Tab. The section will be linked to the model but can also be unlinked. In scales 1:100 and 1:50 little editing takes place. Because of the X`ref system an unlinked drawing may have to be exploded before further detailing. Detailing in scales 1:20, 1:5 and 1:2 will be done with automated details form the Detail Component Manager.

Editing linetypes, lineweights and hatching in ADT also takes place within the Display Properties of the section.

### Plan drawings

Plan drawings are made directly as they are seen on the screen in plan view. These drawings will normally require a minimum of editing before printing as long as they are produced in scale 1:100, 1:50. In AC as well as in ADT editing in plan is mostly about text and dimensioning

Plan drawings in the scale 1 : 20 or more will require more editing. In AC editing will be made by the use of the Detail tool or the Patch tool for smaller areas. In ADT this will be similar operations cutting out smaller parts by using the AutoCAD tools.

### Detail Drawings.

Both software packages has a special tool called the Detail tool to be used in drawings in scale 1 : 20 and up to 1 : 1. In ADT a detail component manager offers some automated detailed hatching systems to automate detail drawing. In ArchiCAD a special Detail tool works for "cutting out" the detail area and at

the same time change in that area into a 2D drawing. The detailing work is done with the ordinary AC 2D drawing tool and the fill tool.

### Printing

Printing in AC can take place within separate module linked to the AC file called PlotMaker. PlotMaker is a layout application where a number of different drawings can be laid out. In PlotMaker correct measurable scales can be set automatically. Although some editing of scale, lines etc can be done in PlotMaker it is recommended that all editing, pens and colours, hatching etc. are done in AC. Any changes made in AC are later updated in PlotMaker.

In ADT all material for print is set in the Sheets area. Layout and Printing takes place within ADT. In Sheets lines and hatching can be controlled directly inside ADT switching between model and paper view.

### Visualizations

In AC visualization is made directly in AC. There are more than one engine for visualizations. The engine making visualizations with the highest quality is LightWorks. The highest quality is similar to the one made in for example 3D Studio. Animations and Virtual Reality (VR) can be made in AC.

In ADT visualizations can be made via a separate visualization module called VIZ Render. Through a link any changes made in ADT is to be updated Visualizations in VIZ Render are made at the highest quality with Radiosity. Animations can also be made in VIZ Render. Visualization can also be made directly in ADT but with low quality.

### Calculations

Both software packages are designed for quantity calculations. In ADT quantity calculations are very simple and easy to make. The result can be places in schedules which can then be exported to Excel format.

In AC calculations seem to be more complicated to do. In Sorthoj we did not try out the possibilities for using the software for calculation.

### *IFC*

With IFC interfaces, it is possible to export a model file in IFC format and import a model file in IFC. The programming structure of AC is designed for IFC and IFC is built in AC. This means that saving the AC model for IFC export and also importing an IFC file into AC is very easy and simple.

With ADT it is different because ADT is a file system implementation based on an external reference system. IFC is not included and it is necessary to make use of an IFC add-on module. Export for IFC is a complicated manual task and when done it works ok. Import is also complicated and does not work well. The IFC software developer seems to have focused mostly op the export function.

IFC is being used for many purposes. One such is simple file exchanges for example between ADT and AC. IFC files are also used for as input into various software for Simulations, calculations, construction analysis (Robot), thermal analysis, Solibri model check etc..

Many software developers on software for these simulations are now being based on IFC.

## ***Conclusion***

AC and ADT are both building modelling tools. As such they are in many ways very similar, yet there are great differences.

They are both based on parametric and "intelligent" objects. They are both building information models based on databases that control that any changes made one place is coordinated anywhere else in the model. Any building information is stored in the data base. As such they both claim to meet the same request from users of today's building industry.

As pointed out earlier the way they are structured are very different. The fact that AC is a database implementation and ADT is a system implementation. This difference has implications for all functions and operations done with the tools.

Modelling takes place in both applications in a similar way. The way the construction objects works rather similar. A Wall for example has similar parameters in both applications. Some parameters vary slightly but in at modelling situation this causes no problems.

The display system is very simple in AC controlled by Display options. It is very easy to understand and control. In ADT it is more complicated controlled by the Display Manager.

Navigation In both programs is controlled by a navigator. Because of the way AC is structured the whole model is controlled live in the Navigator. It is felt as navigating in a 3D model In ADT navigation is more static since the navigator is used to call specific separate drawings. In ADT it is felt as navigation in a number of drawings.

During the modelling process AC has a number of tools to isolate and cut out parts of the model for editing and visualization. In ADT these possibilities are rather few.

In AC it is possible to make new special objects by using the in built GDL programming tool. GDL opens up for the creation of objects of virtually any desired form. In ADT the programming options are more closed and not easy to use. There is some smaller modules built in for example the window and door assembly tool within which new objects can be created.

It is easy through the navigator to move up and down between stories in AC. The way it works is also felt like in real. In ADT navigation between levels are done clicking on separate drawings and is not felt like moving in real.

Layout and print is in AC done in a separate application linked to AC. This way makes it all more complicated. In ADT layout and print takes place within the application and is as such more simple.

In AC visualization takes place within the software in a high quality module called LightWorks. Lightworks is a bit complex but with a simple set up it works fine. In ADT visualisation takes place outside the application in a separate module called VIZ Render. Everything works fine and visualization is of high quality.

## Appendix E – Modelling Sorthoj

The Sorthoj 3D building modelling took place as test modelling at the same time as a traditional building process based on 2D drawings took place.

The project Sorthojparken consists of five 3 storey buildings each of which has 9 flats. These five blocks are rather similar in design yet there are smaller differences in terms of placements of stair and elevator due to the placement of the buildings on the site.

It was early decided that 3D building modelling should concentrate on modelling of two buildings by the use of two different software applications, ArchiCAD and Architectural Desktop. By doing so this would be a good opportunity for making comparisons of the quality of the two.

Building number one was modelled in ADT and building number two modelled in AC

### ***Focus was based on the following issues.***

The first issue was to gain experience with the quality of the chosen modelling application in relation to the modelling process and the ability of the application to meet the requirements of detailed modelling and detailing as a whole.

The second issue was to look into the quality of the 3D modelling application in relation to its ability to produce suitable documents for the contractor to build the building in terms of visualization of the project and production of 2D drawings to be used by the contractor. Further the ability of the application to make use of various external simulation applications for e.g. construction analysis, quantity calculation and thermal analysis.

The third issue was to look into file formats and the ability to make exchange through IFC between the 3D modelling application and external IFC based simulation software.

The fourth issue was to make an evaluation of the two applications in relation to the mentioned issues.

### ***The modelling process***

The two 3D models one in AC and one in ADT was modelled simultaneously.

Because the traditional working process had begun some months earlier than the 3D modelling process, the 3D modelling was based on the 2D drawings already produced by the architectural firm. Therefore no considerations were made about the function modelling phase.

3D Modelling is basically a geometric discipline. Starting from the first conceptual model, ideas are developed through several stages of the modelling process to the final resulting As Built Model. As such, the model evolves through stages that represent the increasing levels of detail requested.

Typically the 3D building model consists of such parts as spaces, surfaces, equipments, constructions, structures and properties. In the early model stages the model is simple in geometry and detail and very few decisions about surfaces, constructions etc has been made. Over time the geometry of the model becomes more and more detailed as long as decisions are made about all aspects of the building. This process also means that the more detailed a model become the more the model resembles the real world building to be built.

How close it is possible to come depends on several factors such as the abilities of the actual software, the processing power of the computers as well as how close to reality the modeller aims to go etc.

The building model has many purposes. One purpose of the virtual building model is to use it for producing documents for contractors to build the building such as creating 2D working drawings incl. building details, quantity calculations and schedules etc.

Another purpose of the model is to use it for analytical simulations. Examples of simulations are visualization, static construction analysis, thermal analysis, quantity and cost calculations etc.

Any 3D model will be an abstraction and any report drawn from a model will need further editing and detailing. But for all purposes, the closer it is possible to come to the real world building, the less work will have to be spent on editing.

### ***Modelling phases***

Modelling is an iterative process. This means going back and forward in the process all the time. However certain stages in the process can be identified depending upon administrative or practical requirements.

An example of this is the Finish modelling phases<sup>11</sup>:

Space model and data

Basic building component model

Building component model

Product data model

As built model

As maintained model

The Space and model data includes the use and quality of space. The basic building component model is the sketchy model in variations. The Building component model is a 3D generic model of the building where the product manufacturer is unknown and the Product data model is a 3D model of the building where the product manufacturer is known.

The Sorthoj modelling phases were only two called.

The Building component model incl. space model

The Product data model

### ***The Building Component model***

In this phase the model was made rather simple and less detailed.

Space modelling is a simple generation of spaces based upon the construction modelling.

Construction modelling of walls, slabs and roofs was made with their total thickness and not divided into their real components. Windows, doors and stairs was made by generic objects found in the applications. As such no product manufacturers was known.

The use of Building Component models is basically for presentation and visualization purposes in order to present ideas of the building as the base for further decisions.

<sup>11</sup> Product modelling guidelines, Seppo Niemioja, Finland 2004

### ***The Product Data model***

In this phase the model is detailed and more complex. In construction modelling the walls, slabs and roofs is detailed into their real components. Also components as stairs doors and windows are now real components and very detailed and has a known manufacturer

These objects will be possible to download from the internet in the future directly from the manufacturer's homepage. However this was not possible yet and therefore building objects were made in GDL.

This Product Data modelling represents the greatest editing process in modelling. A survey done in England shows that 60% of the work takes place in this stage of modelling.

The Product Data model was then used for making reports for simulations and tests. Types of reports are: 1) Visualization and animations (Turntool, VIZRender, Lightworks and VR4 max media lab), 2) 2D working drawings, 3) Construction simulations, 4) Space utilization plans, 5) IFC model check - (Solibri), 6) Quantity calculations for cost estimates (Spile DK, Granlund, Finland), 7) Building automation and thermal analysis.

These reports and tests will be further described in appendix F.

### ***Learnings from 3D modelling***

As stated in the previous paragraph the difference between the two phases, the Building component model and the Product data model is found in the degree of detailing of the building model.

### ***Modelling the building component model***

Through this modelling stage modelling was based on using the ordinary tools in the applications like walls, slabs, roofs etc with no detailing.

Through the modelling process some mistakes modelling was experienced. These mistakes appeared in discussions with external organisations.

In general it is important to make use of the tools in such a way that editing and making changes can be done in the easiest way.

Because the models of this stage will be the base for models in the next modelling stage it was important to think ahead and prepare for future the modelling phases. An important decision to take for ADT was to decide about the project structure because ADT is a file system implementation and for both applications to decide about levels / stories and their heights. If these considerations and decisions are not made properly the problems may arise later in a changing and editing process.

#### *Separation between stories*

It is important to decide where the division line are placed between stories and therefore which parts of an object belongs to which story. An example is material layer object where the ceiling belongs to the first story whereas the concrete part belongs to the story above – the second story.

#### *Modelling the foundations*

Another important decision to make was about which tool should be used for modelling foundations either the Slab tool or the Wall tool. For a start the slab tool was used but later it was learned that the wall tool was the best tool to use. One reason is the slab tool used for foundations makes it impossible for

flexible editing, whereas using the wall tool (later classified as concrete) is easy and quick to handle for later changes.

### *Modelling the exterior walls*

From the start the exterior walls was modelled through three stories. This was based on the idea of the curtain wall. This method of modelling is described as an ideal way of modelling in the Autodesk tutorial.

It was learned that it does not work well for several reasons. One is that of showing only one story at a time the wall as a curtain wall will only be shown at one story. Another reason is when using calculation software it is based on the idea of calculating quantities for each story. Therefore some time in the modelling process this was changed so that the exterior walls was modelled to belong to each story.

Another side of modelling of walls is to decide about where to place justification lines. These lines may be placed arbitrary in the wall or on the left, centre or right side of the wall. Most often the best way is to place the justification lines on the inside of a building. Then later if the wall in the product data modelling stage has to be increased or narrowed these changes will not cause any troubles for constructions inside the building.

### *Roof in relation to storey / level*

From the beginning the roof was considered to belong to an independent story and was modelled as such. After talks with the firm Granlund in Finland it was learned that the roof should not be considered as a story in relation to IFC simulations of the calculation process of quantities. Therefore this was changed in the product data model phase.

### *Roof construction*

The roof at Sorthoj is a flat roof. Normally this will be modelled with the Slab tool. The roof is a flat roof with slopes for water drainage. In this modelling stage the roof was modelled flat with the slab tool.

### *Corner windows*

Placing corner windows in AC was working fine because of the built in corner window tool. In ADT there is no such corner window tool. Here there was two possible ways to pursue, either making a hole in the wall by a Boolean operation and then placing an independent window in the hole or trying to place two windows from the two walls where they meet in the corner (the "corner method"). Using the Boolean method means that no later editing is possible. Using the corner method means complicated and incorrect connections.

### *Generic objects*

A number of generic objects were used in the model in this stage such as windows, doors, stairs and furniture. These objects are parametric and the purpose is visual.

In AC these objects are programmed in GDL and they are basically of very high and detailed quality. Generic objects in ADT in particular furniture is very simple and of bad visual and detailed quality.

### *Modelling the Product data model*

The characteristic of this modelling phase is the increase of detail and that objects such as for example windows is no longer generic objects but product specific objects produced by a specific manufacturer.

### *Detailing of walls, slabs and roofs*

When it came to detailing of the walls, slabs and roofs into their material layers / real components there are differences between ADT and AC. In AC detailing of walls, slabs and roofs is called composites and the detailing here is a graphical representation. In ADT only walls can be divided into its components. Here components are real geometric parts. Slabs and Roofs cannot be divided into components.

### *Product specific objects*

In this modelling stage it is important to make use of objects produced by real manufacturers so they are product specific. They be downloaded from the homepage of a manufacturer. However to day it is mostly not possible. Some manufacturers have issued a CD with such objects like furniture and lights.

In the Sorthoj case as an example some product specific objects was modelled in GDL simulating a download.

One is the stairs produced by Dalton. The stair was modelled in GDL as an exact Dalton stair model and used in the Sorthoj model. The components made by GDL were easily placed in the AC model basically because it is their own native environment. In AC the placement of the GDL object caused no problems whereas in ADT placement caused problems. Placement of GDL objects in ADT is not directly possible but requires a small application called GDL Object Adapter. By using this it is possible to place a GDL object in ADT. However this method is not optimal. The GDL Object Adaptor is not fully developed. Glass for example cannot be shown with transparency.

The same applies to another product specific object used the internal doors produced by Westwood Swedoor Jutlandia also made in GDL.

Another detailing in the project data model was made around the balconies. The balconies was designed with a wide overhang meaning the support of massive steel constructions. These steel constructions was modelled as an example only in ADT in great detail for later computer simulation analysis.

### *Detailed Roof construction*

The roof is a special construction due to the fact that it has slopes on its top. No real well working tool was found. In AC the Mesh tool was used but it is extremely complicated to correct for slopes falling in many different directions. In ADT it was even more complicated. A way of doing it is by using massing elements. They are more complicated to use and control in terms of correct sloping. In such a case a choice has to be made whether a complicated not completely correct sloped roof is to be used or the more abstract solution with a complete flat roof is the best solution.

### **Conclusion**

The modelling experience is derived from two modelling phases the Building component model and the Product data model.

For both it is important to decide at start of modelling about the structure of the model, the levels and heights.

It is important to decide about how the model can be separated into parts such as which objects belong to which story. The example here is the exterior wall through several stories contra the wall following only a particular story.

It is also important to decide about which tools to be used for which purposes. The wall tool should be used for foundations instead of the slab tool.



Building parts such as a roof should be placed as an object not as a story. This means that it is important to decide about where the stories are divided.

Generic objects are fine in the building component stage and are often fine to use. They are of varies quality for visualization, particularly fine in AC.

There is a need for possible download of product specific objects fra manufacturers to be used in the product data model.

Detailing of walls, slabs and roofs can be made in AC as graphic presentation. In ADT detail can only be made in the Wall and in a geometric way.

GDL objects are fine to use. Especially in the case, where particular objects are not found. This is the case of the flat roof with sloping areas.

### ***Literature***

- 1) The Building Information Model: A Look at Graphisoft's Virtual Building Concept, A Cyon Research White Paper, January 2003
- 2) New Features Guide, Autodesk Architectural Desktop 2005, Marts 2004.

## Appendix F – Sorthøjparken - Presentation and Evaluation

### ***Reports of the Product Data Model***

Several extractions of the product data model in AC and ADT has been made as a base for simulation. Sometimes these simulations have taken place directly in the application either in AC or ADT, 3D Studio format or they have been taken on the basis of ifc.

Visualization and animations.

Quantity calculations

Construction analysis in ADT2005

Thermal analysis

DDS Viewer

IFC model check - (Solibri)

### ***Visualization and animations***

Software applications for visualizations has in ADT been Autodesk VIZ 2005, in AC Lightworks and VR4Max based on a 3D studio file.

In AC visualization in very high quality can be made within the software itself by using Lightworks. In ADT the visualization application is external through a link to VIZ Render. VIZRender makes absolute high quality visualizations. Any AC 3D view can be saved in 3D Studio format and in ADT the VIZ Render file is saved as 3D Studio file.

The models were also tested in relation to animation and VR. In AC animations like walk through and VR can be made directly within the application. In ADT walk through and VR is made in the external application VIZRender.

For both AC and ADT other solutions for animation are available through saving the model in 3D Studio format. These are Turntool and VR4Max. Turntool is a plug-in to 3D Studio. A fully made and of very good quality Turntool solution was made.

Finally VR4Max was tested not for the purpose of making a final high quality animation only for the purpose of testing the ability of the software formats for exchange. These tests proved to be fine.

### ***Visualization in AC***

AC has different rendering engines to perform the work of visualization such as Internal engine and Lightworks engine..

### ***Internal rendering engine in AC***

This rendering engine gives good quality rendering as shown in figure 1 mostly for sketchlike presentations. Shadow casting is fine for project presentation.

When it comes to final high quality presentations Lightworks rendering engine will be required for the highest quality rendering.

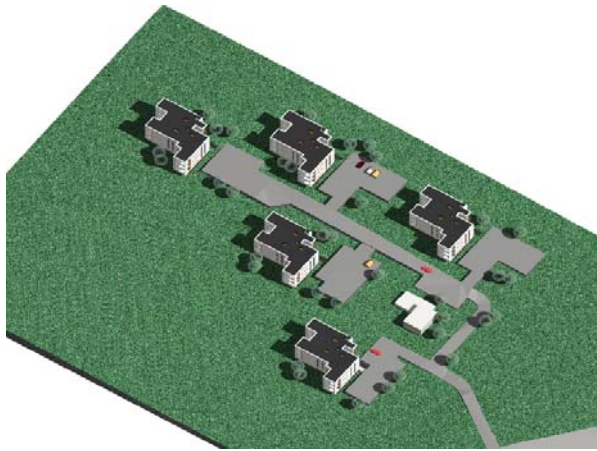


Figure 1 - AC model – internal rendering engine

### *Lightworks rendering engine*

This rendering engine is a high quality engine which gives the very best quality rendering in terms of shadow casting, reflections in glass eg from skies et. The quality resembles reality very close to real world.

Lightworks uses Ray Tracing by tracing the path taken by a ray of light through the scene and calculating reflections, refraction or absorption of the way whenever it intersects an object in the world giving much more realism to rendered images.

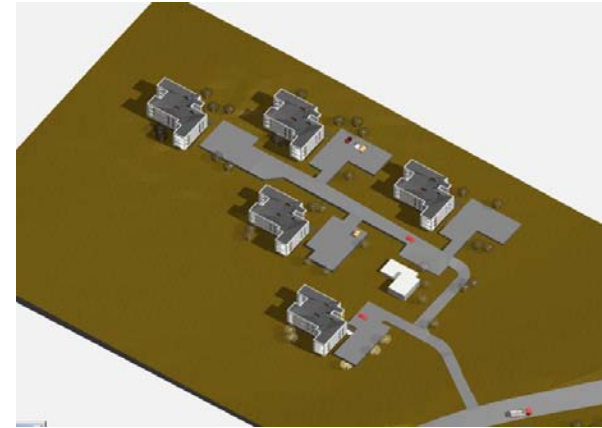


Figure 2- Example 1 in AC Lightworks rendering engine



Figure 3 - Example 2 in AC Lightworks rendering engine

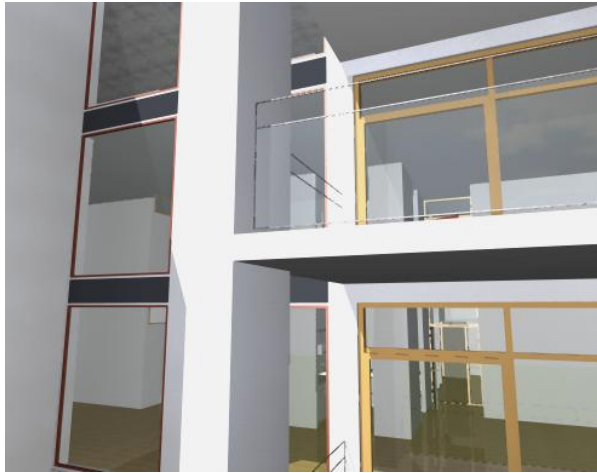


Figure 4 - Example 3 in AC Lightworks rendering engine

### *Rendering in 3DS format*

In some cases 3DS format is used for rendering in AC. First the model is saved in 3DS format and opened in Autodesk VIZ.

The quality of the renderings this way proved to have the same quality as described for ADT models rendered in Autodesk VIZ..

### *Visualization in ADT*

In ADT visualizations are normally carried out within VIZ Render or Autodesk VIZ. The two ways of rendering is more or less the same both based upon 3DS format. The term VIZ covers rendering in VIZ Render2005 and Autodesk VIZ2005.

### *Rendering in VIZ.*

In ADT the model is linked to VIZ Render for high quality render. The quality of rendering in VIZ Render resembles the quality in Autodesk VIZ.

Rendering in VIZ does not always work completely well. As an example the figure below shows that the background shows itself through the windows, despite fact that inner walls in reality closes for this. It seems as the rendering system does not work completely well..



Figure 5 - VIZ Render billede ikke helt korrekt udført

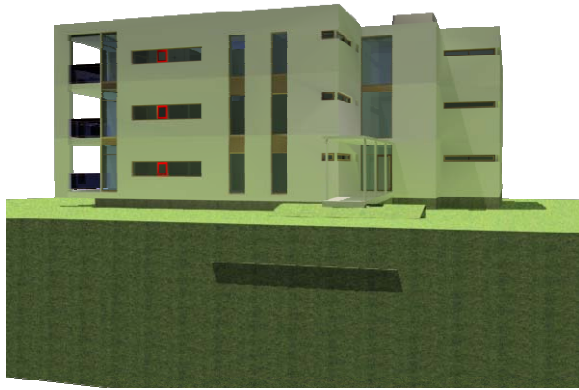


Figure 6 -

### Virtual Reality

Virtual reality becomes more and more important as visual representations of projects. It is important that clients are able interactively to walk around in the buildings controlled by the user itself on the internet.

Two types of VR which meets these demands, has been tested in Sorthoj. The one is with the application TurnTool and the other through VR4Max. Other VR types can be made directly in AC. They are called VR Scene and VR Object. These types of VR was not tested.



Figure 7 - 3D from the air showing rooms

### TurnTool VR

TurnTool is a small plug in installed in Autodesk VIZ. and later used on a model in 3DS format.



Figure 8 - TurnTool interface showing view 1

A fully made and of very good quality TurnTool presentation was made and placed on the internet. The presentation has five user control possibilities shown as buttons below the window. They are shown in figure 7 and 8.

The first button gives access to a full user controlled Walk around outside the building. The second button gives access to an automatic show around pre-made without any user control.

The third button gives access to a view from the drawing room with user interactive move around possibilities. The fourth

button is similar from the dining place towards the kitchen. The fifth button is a 3D view from the air of the total building with no roof. The visible second floor is equipped with furniture and the user can control the moving around.

### VR4Max VR

First the idea was to make use of the IFC model for the purpose. However the IFC model was not good enough because all surfaces was equal in terms of materials and colours. Further the problem was how the IFC model was interpreted by the system.

Therefore the 3D model was taken into the VR system via the traditional 3DStudio format. The Model file received by VR Media Lab was generated in ArchiCAD. This model also contained site information, which the IFC model did not contain. The Model was then imported into 3D Studio Max, this time was with various surface information.

The model was placed into the Panorama viewer. A sky dome was placed around the actual site area with the horizon as realistically placed as possible. The pictures used were not originally photographed for the purpose. Therefore, overlaps of photos around the panorama were visible. The model was finally supplied with trees casually placed.

Finally the model was ready for export. A VR file is generated via a plug-in in 3DS Max (VR4Max). VR4Max is also used for the panoramic presentation.

The Panoramic presentation is special compared with presentation on a traditional computer screen. One side is the presentation itself the other is the technical solution. A panoramic presentation will require 3 PC set up in a cluster synchronised in order to present the same picture. Further is used dedicated VR software which can operate in connection

with the cluster and present a geometrically correct picture on the curved screen.



Figure 9 – The Panoramic setup

The experience of this is very realistic with the peripheral view as well as the scale of 1 to 1. The viewer feels his own body dimension as a part of the show resulting in realistic experience of volume and shapes of the virtual building.



Figure 10 - The model in VR4 Max

### **Quantity calculations**

The ADT model was used to test different types of quantity calculations. In example 1 calculation test was made directly within ADT with the tools of ADT and the data was the exported into Excel.

Example 2, calculation tests, was made through IFC exports into calculation systems handled by professional quantity surveyors.

#### *Example 1. Calculations made in ADT / Excel.*

Quantity calculations were made for specific parts of objects in the building. This was done for Exterior walls, interior walls, windows and doors, roof and foundations. The example here is from the exterior wall schedule.

ydevægge_tegl								
ID	overskrift	Beskrivelse	enhed	mængde	vægtype	bredde	højde	længde
6	Wall		m2	8.74	350 mm hulmur tegl tegl	350mm	9825mm	890mm
4	Wall		m2	27.56	350 mm hulmur beton tegl	350mm	9825mm	2806mm
1	Wall		m2	8.74	350 mm hulmur tegl tegl	350mm	9825mm	890mm
20	Wall		m2	29.6	350 mm hulmur beton tegl	350mm	9825mm	3356mm
9	Wall		m2	27.73	350 mm hulmur beton tegl	350mm	9825mm	4265mm
	Wall		10m2	21.25	350 mm hulmur beton tegl	350mm	9825mm	1812mm
8	Wall		m2	50.91	350 mm hulmur beton tegl	350mm	9825mm	5590mm
7	Wall		m2	102.25	350 mm hulmur beton tegl	350mm	9825mm	15368mm
11	Wall		m2	124.16	350 mm hulmur beton tegl	350mm	9825mm	18499mm
19	Wall		m2	2.29	350 mm hulmur beton tegl (2)	350mm	9825mm	2230mm
14	Wall		m2	16.87	350 mm hulmur beton tegl	350mm	9825mm	4810mm
14	Wall		m2	29.23	350 mm hulmur beton tegl	350mm	9825mm	4450mm
12	Wall		m2	8.74	350 mm hulmur tegl tegl	350mm	9825mm	890mm
3	Wall		m2	23.59	350 mm hulmur beton tegl	350mm	9825mm	2401mm
5	Wall		m2	8.74	350 mm hulmur tegl tegl	350mm	9825mm	890mm
15	Wall		m2	14.31	350 mm hulmur tegl tegl	350mm	9825mm	2940mm
20	Wall		m2	8.74	350 mm hulmur tegl tegl	350mm	9825mm	890mm
2	Wall		m2	92.19	350 mm hulmur beton tegl	350mm	9825mm	15802mm
				605.66				

Figure 11 - The schedule for walls directly from ADT

First quantity calculations for all exterior walls was made in ADT. The result is shown in figure 10.

ydevægge_tegl										
ID	overskrift	Beskrivelse	enhed	mængde	enhedspris	pris	vægtype	bredde	højde	længde
6	Wall		m2	8,74	kr 0,00	kr 0,00	350 mm hulmur tegl tegl	350,mm	9825,mm	890,mm
4	Wall		m2	27,56	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	2806,mm
1	Wall		m2	8,74	kr 0,00	kr 0,00	350 mm hulmur tegl tegl	350,mm	9825,mm	890,mm
20	Wall		m2	29,6	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	3356,mm
9	Wall		m2	27,73	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	4265,mm
17	Wall		m2	21,25	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	1812,mm
8	Wall		m2	50,91	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	5590,mm
7	Wall		m2	102,25	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	15366,mm
11	Wall		m2	124,16	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	18499,mm
19	Wall		m2	2,29	kr 0,00	kr 0,00	350 mm hulmur beton tegl (2)	350,mm	9825,mm	2230,mm
14	Wall		m2	16,87	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	4810,mm
14	Wall		m2	29,23	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	4450,mm
12	Wall		m2	8,74	kr 0,00	kr 0,00	350 mm hulmur tegl tegl	350,mm	9825,mm	890,mm
3	Wall		m2	23,59	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	2401,mm
5	Wall		m2	8,74	kr 0,00	kr 0,00	350 mm hulmur tegl tegl	350,mm	9825,mm	890,mm
15	Wall		m2	14,31	kr 0,00	kr 0,00	350 mm hulmur tegl tegl	350,mm	9825,mm	2940,mm
20	Wall		m2	8,74	kr 0,00	kr 0,00	350 mm hulmur tegl tegl	350,mm	9825,mm	890,mm
2	Wall		m2	92,19	kr 0,00	kr 0,00	350 mm hulmur beton tegl	350,mm	9825,mm	15802,mm
				605,66	kr 0,00	kr 0,00				

Figure 12 – tilbudsskema I Excel

Next step was to export the schedule into Excel where the design of the tilbudsark was made, figure 11. All columns in the Excel schedule are locked apart from the green areas. The left green area is for typing the unit price. This will result in price

calculation results automatically made and shown in the right green column. Below in that green column the total cost of the brickwork on the exterior walls will be listed.

### Example 2. Calculations made in AC and exported via IFC

Another calculation test was carried out by Byggeriets IT in Copenhagen Denmark to be used for quantity take-off and pricing. Analysis was made on an ifc file from AC using the application Visual calculation 1.13. The general evaluation points out that object types like IfcWall, IfcWallStandardCase are well defined. The model was not shown properly and it seems as if there are no references to several openings in exterior walls. Several proxy elements were found. Connections between space and walls were found. The number of doors, windows and spaces seems to be ok.

### Construction analysis in ADT2005

An isolated part of the building was used for structure simulation and analysis. This part, the balconies is the most interesting part for analysis because of their large overhang. The structure is made in steel and hidden in the concrete and brick structure.

First steel structure was modelled in detail. Then it was exported as an IFC 2x file, and finally structure calculations was carried out in Robot Millennium/CBS v.17.5.

The purpose of the study was to examine first if IFC is a suitable exchange format and second to use Robot for calculations to find out how suitable the IFC file in Robot is for calculations.



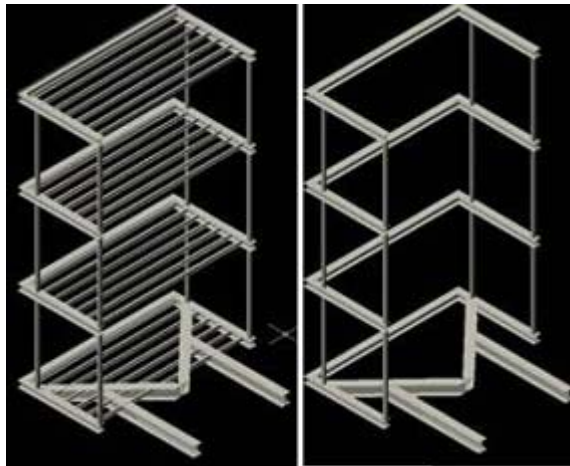
### *Static calculations of the balconies*

An alternative to use IFC as exchange format is through using ACIS format or DWG/DXF format.

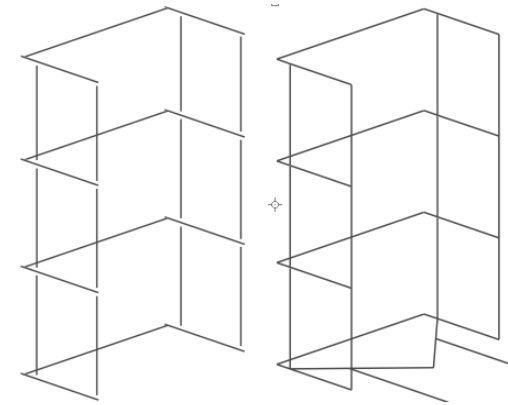
It was decided only to make use of IFC exchange format. The calculations made are based on those loads shown in static calculation documents made by the engineer Erik Holbech. Here the structure is calculated only for gravity loads and live loads.

Most calculation software is based on Finite Element Method. This also applies to Robot. In this theory a beam element is considered a one dimensional element in the length direction with no dimensions for height and width. This means that calculations are based upon the centerlines of the steel profiles. Each element has to be connected through their respective centerlines as well as being connected in their endpoints with nodes.

In a real live structure centerlines will not always be placed at the same level and the endpoints will most often not be connected.



*Figure 13 - Left shows the steel structure as a hole. To the right all simple beams has been removed for easier overview.*



*Figure 14 – Centerlines connected to endpoint nodes*

### *Adjustments*

When the model is imported from ADT into CBS (Robot) the real form looks like this seen from the centerlines point of view.

As shown to the left beams in the bottom of the structure is not imported within the IFC file for some reasons. Perhaps this is due to the fact that the structure is not adjoining.

The diagonal beams in the bottom of the structure are correctly imported with the IFC file if it is imported from another CAD program for example AC.

It is therefore necessary to modify the structure in relation to these assumptions around connection of centerlines. Every structure element has one node in both ends, and the elements has to be connected through these nodes (see figure 13).

## Loads

The structure has been calculated with gravity loads and live loads.

## Construction

The structure can now be placed in Robot via CBS. In this process a weakness of this version of Robot CBS is presented. If the structure was made up by concrete or wooden columns the IFC file would transfer all data from ADT into Robot/CBS.

In this case where all elements are steel profiles all elements has to be redefined. This means that the information about a HEA 240 beam has to be given again in Robot.

When this is done the structure is ready for calculations in Robot. Now Robot knows the profiles by their names and with the inbuilt cross section properties e.g. HEA 240 profiles.

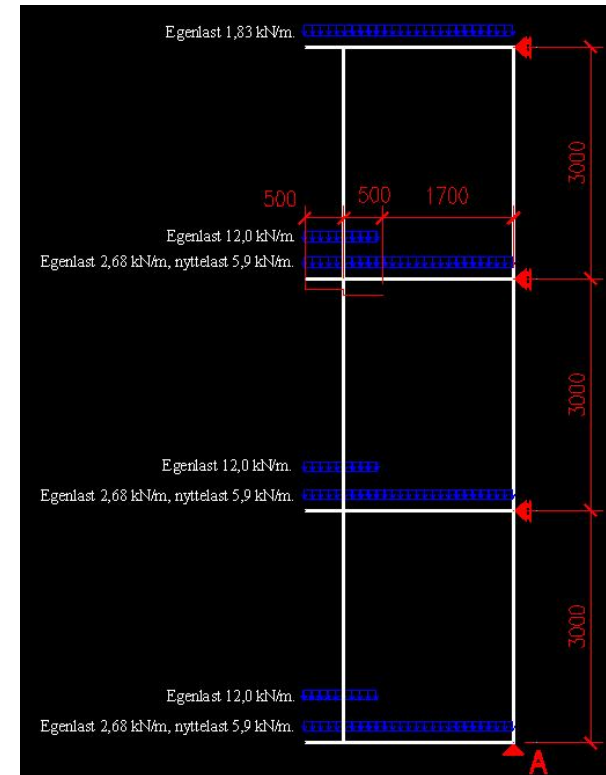


Figure 15 -

## Numbering of elements

In the calculation programme any element will be given a specific number.

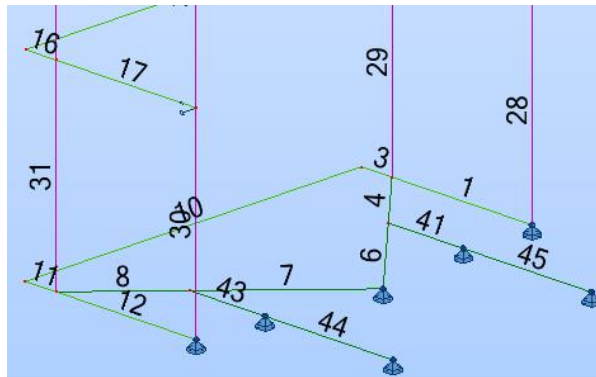


Figure 16 -The result of the calculations are shown as such

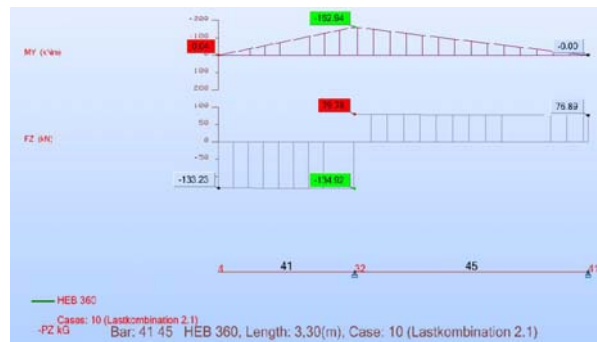


Figure 17 - Conclusion of the structure calculation test.

The ACD model in ADT shows a great overview of the structure and how it works.

### Beams and frames

The problems with centrelines were difficult to deal with. This is because the calculation program requires centrelines to meet through nodes. When a structure in real life consists of profiles with different heights and with the top of these profiles is placed

at the same height centrelines will not meet. Therefore the model always has to be adjusted to meet these requirements.

The IFC transfer from ADT to Robot CBS would most likely have been with no problems if the beams had had rectangular sections like the case of concrete beams. It is expected that future Robot software will be made for calculations of beams with different sections than the rectangular.

The calculations itself in Robot proved to be very flexible and smooth and is designed with facilities for a great number of different analysis

The results of these analyses have proven suitable for making necessary adjustments and changes in the ADT model.

It seems that in version v19.0 of Robot millennium / CBS the problems with centrelines enjoyment are solved.

### Thermal analysis

A thermal analysis test based on an ifc file was carried out through the firm Granlund in Finland using the application Riuska.

As test examples two tests were made. The first test was done as a test of total energy consumption of one story of the building. This included analysis of consumption of lightening, equipment electricity, HVAC Cooling electricity, HVAC other electricity and Heating (see figure 19).

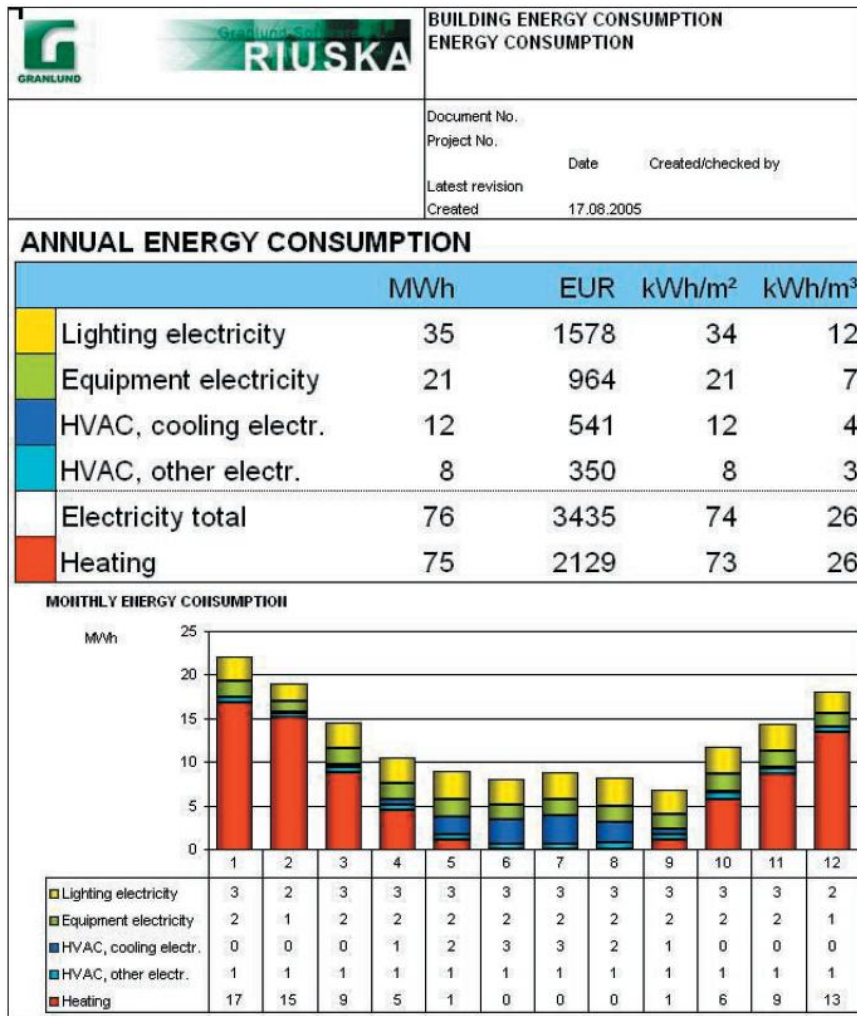


Figure 18 – Energy consumption simulation

The second test was done on comfort simulation of one room in the building. The size of the room is 10,9 square m and 30 cubic m (see figure 20).

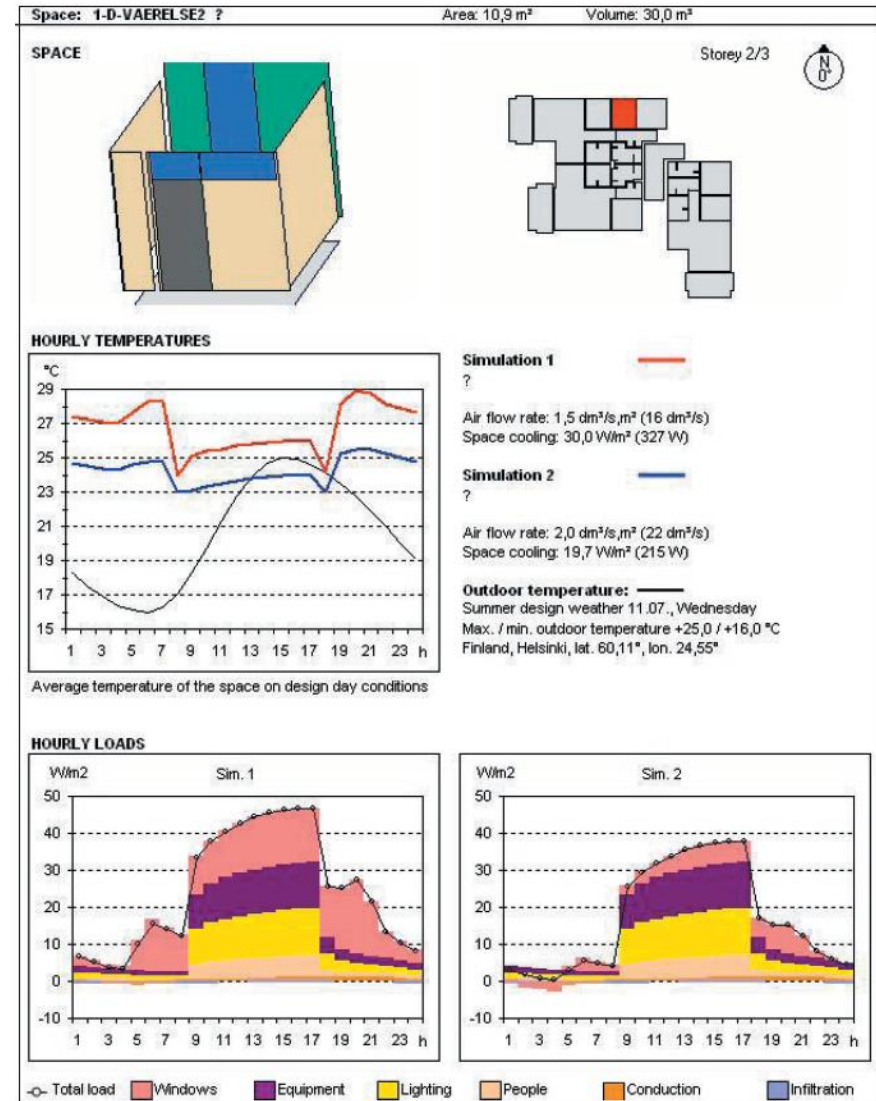
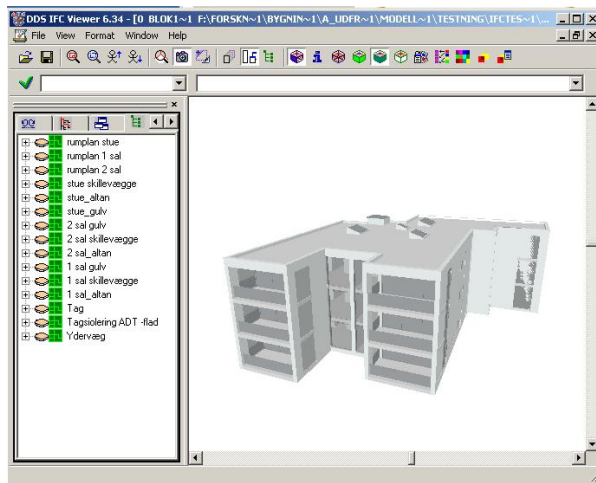


Figure 19 – Comfort consumption simulation

### ***The DDS Viewer for IFC model analysis***

First the AC IFC model has undergone analysis through the DDS IFC Viewer. By using the Viewer it is possible to see which objects in the model has undergone full IFC translations and which has not. With ArchiCAD we have had good results so concerned. Another analysis has been carried out by using Solibri Model Checker.

Finally several other test of IFC has been done in order to get experience about file exchange between ArchiCAD and different kinds of software to be used for simulations. The quality of the IFC ArchiCAD file is already known from the tests in the DDS viewer.



*Figure 20 – Model shown in DDS Viewer*

### ***Using the DDS Viewer for 3DS export***

The IFC format was not found as a possible exchange format for visualizations yet. However it is possible to save the ADT and AC models in IFC format and open them in the DDS Viewer. From there it is possible save the model in 3DS format.

3DS format is the key format for exchange to many visualisation programs in particular Autodesk VIZ (see figure 21).

### ***Solibri Model Checker***

Solibri Model Checker (SMC) is an application used for analysis of building models for integrity quality and physical safety. The system offers user visualization with an intuitive walk-in functionality. Solibri X-rays the building model and reveals potential flaws and weakness in the design, highlights the clashing components and checks that the model complies with the building codes and organizations best practices.

SMC has a number of built in Rule Sets for checking the 3D building model. One Rule Set is BIM Validation Architecture. As an example this Rule Set checks that architects doors and windows don't fit the openings in the structural model. It also checks that all components are contained in a building storey. Further Space checks can be made for example that every space has a unique identifier (in the whole model or in the same space group). It also checks that identifiers are numbers, and correct numbers (when this is required).

Other Built in Rule Sets are Structural versus Architectural, Intersections between domains, Model Comparison – Architecture, Model Comparison – structural and building services, Security etc. Further it is possible to define ones own rule Sets.

The results of an SMC analysis is visually shown by the generation of for example coded arrows that indicate different problem categories, colour indication of problem objects together with a written report outlining the problems.

In Sorthoj SMC, IFC models of the AC and ADT models was checked as a BIM Validation Architecture with great success. The SMC check this way was used to examine and reveal the problems of modelling.

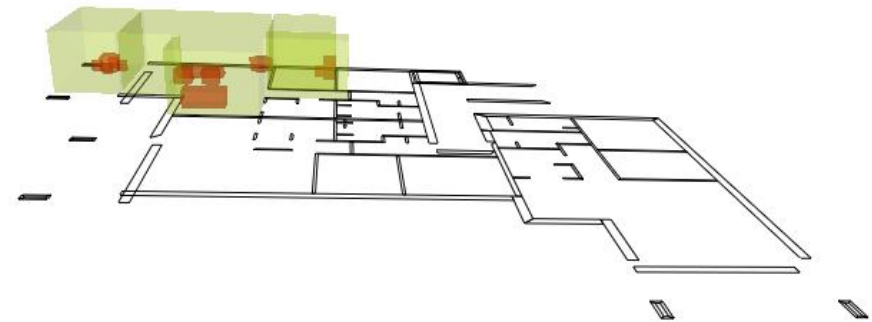


Figure 22 – Example, Space analysis in Solibri

As an example Solibri Model Checker finds the places where building services components collide with load bearing structural elements. Another example is analysis of whether load bearing components have supporting structure underneath. Also space analysis has been carried out checking alignment with surrounding walls and slabs overlap and space height.

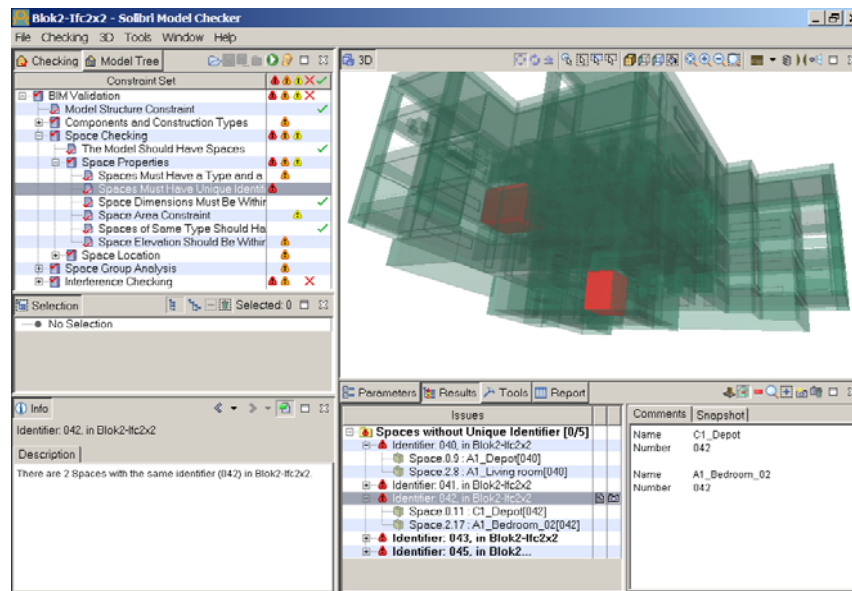


Figure 21. Solibri IFC Model Checker

### Reports – 2D working drawings.

2D working drawings are easily drawn from the 3D model. However they are simple and not detailed enough compared with traditionally made drawings. The question arises should drawings produced directly from 3D models look exactly like traditional made drawings as long as they are informative enough to be used by the building industry. Drawings drawn directly from models cannot be more detailed than the model is made.

The quality of sections drawn from AC and ADT are almost the same. In AC they are slightly better because composite detailing on all constructions can be made automatically. In ADT

only composite detailing can be made on walls not on slabs and roofs.

The quality of elevations drawn from AC and ADT can be drawn automatically. However elevations drawn from AC can be made automatically with shadows.

Both will require some detailed editing.

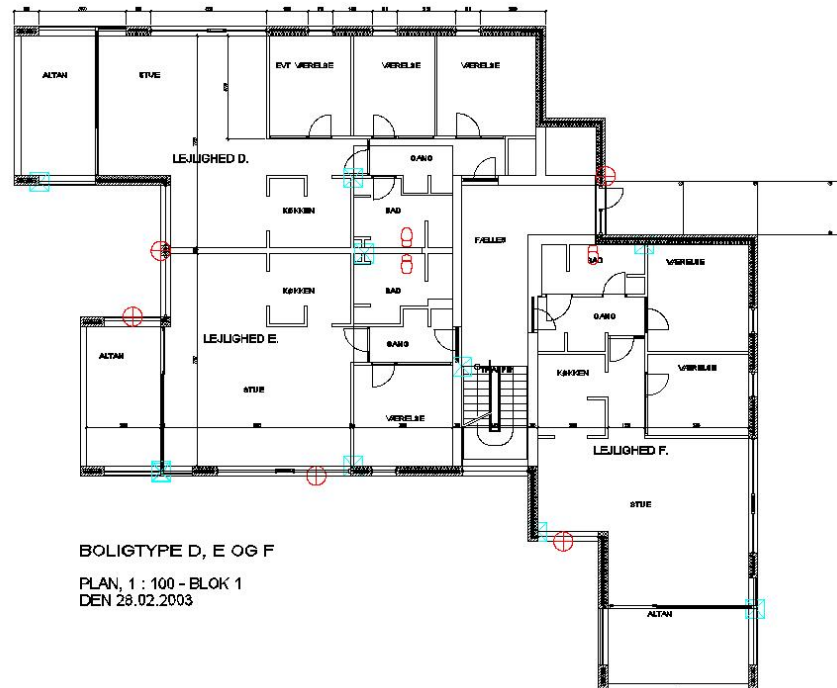


Figure 23 – Extract from model – 2D plan drawing

## Conclusion

This appendix only describes a few simulations out of many which can be carried out on the basis of 3D building models. Some of these has been done directly within the building modelling application itself, others done on ifc models extracted from these models.

It is becoming more and more common that simulations are carried out on ifc models. This is because software developers increasingly develop their simulation software in ifc format in order to be compatible with any type of application used for building modelling.

In the case of Sorthøj some simulation end evaluation was done within the software itself such as visualization. Some more advanced visualization was done by external parties like VR Media Lab, AAU.

Other types of evaluation and simulation were done in external IFC based software often done by external parties. This was the case of Thermal analysis, Structural analysis and Cost calculations. Also internally analysis was done such as calculations, Model Checking etc.

Great interest by private parties in testing the models of this project was shown. They received the models in ifc format. Despite the great interest shown by these firms it was not always easy to get their feet back. In some cases no feed back has yet been received.

However as a whole it seems possible to make good use of various kinds of simulation software. It seems as if the quality of simulation applications increases these tools will be very important in the future in order to gain great productivity from building models.

In terms of visualization high quality visualization can be made within the software. However specialist applications and professional firms are able to produce products of particular interesting types of visualization and quality.

The same applies to calculation. From the building modelling applications data for volume, amounts etc are easily extracted from the models. However, professional firms have developed specific software, which can transform these data into for example lists for bidding to be used directly by the builders. Also Thermal analysis is a specialized task, which requires specialized applications.

In this research work it has only been possible to carry out limited analysis and simulation tests. Despite, these tests indicate great potentials for model simulations.