Scale Level and Design Phase Transitions in a Digital Building Model

Stefan Boeykens¹, Herman Neuckermans¹
¹Catholic University of Leuven, Department of Architecture, Belgium
http://www.asro.kuleuven.be/IDEA

Abstract. Research and development on Computer Aided Architectural Design often focuses on simulating a building as a digital model. Our research on the early design stages explores concepts we feel are lacking in current design tools and research projects. Building models are usually static models, serving as a snapshot of the design. We aim to support design phase and scale level transitions, to better support the workflow of the designer. This paper formulates our approach at supporting transitions in an integrated design environment for architecture.

Keywords. CAAD, Architectural Modeling, Design Process.

Introduction

Building Information Modeling (BIM) makes a distinction between simulating a building in a digital form and geometric modeling or drawing. The BIM concept is also referred to as the Generic Building Model (Eastman, 1995) or the Virtual Building™ (www.graphisoft.com/products/virtual_building). The “BIM Debate” on the LaiserinLetter Website (www.laiserin.com/features/bim/index.php) gives a detailed summary of articles and discussions, mainly by commercial CAD vendors. Where traditional CAD primarily focuses on drafting and modeling, BIM specifically uses building semantics including both physical and conceptual elements, such as walls and floors, spaces and activities.

A number of academic research projects have developed BIM concepts in different formats. The COMBINE project (Augenbroe, 1995) and the RATAS project (Björk, 1994) are two examples. A more in depth overview can be found in (Galle, 1995). But BIM is not only an academic research topic. The concepts are commercialized in several architectural CAD-applications, which differ widely in their implementation. Leading examples include ArchiCAD and Revit. They are increasingly used by architects worldwide, in construction projects ranging from single-family housing to high-rise buildings. Examples of the latter include the WTC Freedom Tower by Skidmore, Owings & Merrill LPP (www.som.com) and the Eureka Project by Fender Katsalidis Architects (www.aecbytes.com/feature/EurekaTower.htm).

Design should remain an iterative process based on exploration and feedback. In the early stages of the design, information is limited, yet the impact of design decisions is still large. Tools that support evaluation in the early design stages can improve this feedback, but the additional input at the design stage needs to be recycled in later design phases. By enabling the building model to
transfer between different scale levels and design phases, the design can be transferred from the sketch phase to the construction phase.

The design environment

In our own research and development, we explore concepts that we feel are lacking or limited in current design tools and research projects. The transition of the design model between different phases and scale levels is one of those problem areas. Experience in using current design software for architects lead us to consider an Integrated Design Environment for Architecture (www.asro.kuleuven.be/idea), which is a software framework to support an architect in the early design stages. This framework consists of a custom data structure to describe a building model and a prototype application to evaluate the data structure and interact with the model.

The Conceptual Framework

The foundation of the integrated design environment is the Conceptual Framework (Neukermans, 1992). The main scheme presents scale levels and design phases with integrated evaluation tests adapted to the scale level.

The Core Object Model

The framework was translated into a conceptual object model, using the MERODE methodology (Hendricx, 2000).

The IDEA+ Prototype

The Core Object Model forms the basis for the development of the prototype of the integrated design environment, IDEA+.

Its main purpose is capturing design decisions. During the development of a project, the designer decides on positioning of building elements, on the categorization of these elements and on possible relationships between these elements. To support the workflow of the architect, IDEA+ needs to sup-
Port design phase and scale level transitions, so the model can be explored from different angles, leaving its integrity intact. Evaluation tests can inform the designer directly from within the design environment, to provide immediate feedback, without the necessity of exporting the project into an additional simulation tool. Through a choice of different design representations, the architect looks at specific information in a different format. These include graphic views but also hierarchical, relational, and textual representations.

IDEA+ is a C++ data structure with a graphic prototype application. The application is a hybrid between a CAD and a 3D application. The intention is to illustrate the concepts of the framework through a usable design tool.

Transitions

We distinguish between scale level and design phase transitions.

In a scale level transition the scope of the building changes from a large scale Masterplan level, where the building is described on site, over a mid-scale Block Level, where rooms and spaces are described, to a small scale Space Level, describing the actual building elements. We usually follow a top-down approach from the global level in the Masterplan to the specific details in the Space Level. It is, however, perfectly valid to adopt a bottom-up approach, starting from a construction detail and scaling up to a global building system. We specifically allow the design to be started at any scale level and to freely explore the other scale levels during the design process.

In a design phase transition, on the other hand, we follow the chronological order of a design, from conceptual planning, over sketch design to preliminary design and eventually into a construction design phase. We can even extend the design phases to the post-built phase into Facility Management.

Although the order of the design phases is chronological, we want to support the reverse transition for design exploration and feedback. When exploring a design, switching to higher scale levels might also trigger a transition to an earlier design phase. It is not necessary to look at the construction design phase, from the Masterplan Scale Level.

Transitions in BIM software

Some applications support aspects of these transitions, but there are certain limitations that might disrupt the workflow.

Architectural Desktop

Autodesk Architectural Desktop (www.autodesk.com/adt) is an AEC application, using AutoCAD as its core CAD-engine. The software utilizes a display manager to allow customized representations, so the display of different scale levels and design phases is supported. Apart from the regular physical building elements, we can model spaces as a design entity. A space boundary can even be

Figure 5. Example of Scale Level transition in ADT ArchiCAD
converted to walls, but the walls carry no connection with the original space.

The software also has means to convert a massing model into a set of building elements, through the use of Mass Elements and Mass Groups. A building volume can be created from a combination of shapes, profile extrusions and Boolean operations. This model can be sliced, to generate floor slabs. Unfortunately, this is also a one-way approach. Once the model is converted, there is no link between the Mass Elements and the generated building elements. Additional modifications of the design will not be reflected to the other elements.

Graphisoft ArchiCAD (www.graphisoft.com/archicad) is an architectural design application, built around the BIM concept. It is a standalone application, not a module for a CAD application.

The representation of design elements can be influenced by the display settings and also by the display scale. On the one hand, display settings can trigger visual indications, such as reference lines for walls and handles for text or images. Display settings can also trigger output properties, such as patterns, fills and line weights.

On the other hand, through the nature of the GDL-scripted elements, building objects can be programmed with scale sensitivity. The display scale, which is disconnected from the zoom level, allows texts and dimensions to always be drawn at their regular point size, but, more importantly, allows library objects to take a different representation when this display scale changes.

These two options give a fair support to represent different design phases, but scale levels are much harder to support. There is a zone tool, to model spaces and they can be connected to enclosing building elements, but exploration of dif-
In their current state, these applications don’t allow a free design exploration. It is not possible to switch back and forth between scale levels and modify the design, while maintaining a coherent building model. The architect expects adjustments to the mass model to be translated into an adjustment of the building model and vice versa, if at all possible. Modifying the amount of floor levels on a building block is a good example.

The support for design phases is better, since this can be largely solved through representation settings. As an example, the design information for a wall is (partly) contained in its endpoints and its composition, which aren’t modified by choosing a different representation.

Implementing transitions in IDEA+

The conceptual meaning of these transitions has been described in (Boeykens et al., 2002; Boeykens and Neuckermans, 2003a and 2003b). The elaboration in the IDEA+ prototype is supported by several key features, which we will describe more in detail.

Creating a grid of reference points

To allow a design to be elaborated in different scale levels, while maintaining the integrity of the model, we introduce reference points. These are positions in the model which can be shared between CAAD Entities. They are the control points for elements, allowing elements to be connected to each other and maintain their connections during modifications.

These reference points can be used to build up a reference grid. Adjusting the grid can move...
reference points and the elements from which they are referenced. A regular grid can be manipulated into an irregular grid, by transforming grid lines. Deleting grid lines will disconnect the reference points without the need to delete the building elements. Moving the control points of elements to grid points can establish a connection.

Additionally, reference points can be split up in a 2D part and in a height reference. The height reference can act as a building floor reference, to allow elements to be positioned relative to a building story. Adjusting story heights can actively modify the model and allow elements to stay connected. When we refer to the 2D position of a reference point, this position can be maintained for elements on different floor levels. It is not common for grids to differ between floor levels. Through the support of grids as a design entity, the user can connect elements to any grid that seems relevant.

**Add Classification Information**

Most CAD software relies on layers to structure a drawing. This has lead to the ISO standard on Layer naming conventions (Björk, 1997). The use of BIM can diminish the need for layers, which act as a one dimensional vector into a list of element properties. Layer conventions usually lead to a fairly large amount of layers and each additional parameter that is introduced will simply multiply the possible amount of layers. A better solution is making these parameters into attributes of building elements. Different layer schemes can still be derived from element parameters, but now we have a multi-dimensional structure.

We apply the BB/SfB classification in the IDEA+ data structure (De Troyer et al, 1990). This is based on CI/SfB and is a generic approach to structure element attributes. Combining the classification attributes with the proposed attributes from the ISO 13567 standard defines our common attributes for all CAAD Entities.

The simplicity of only having to assign a layer to an element does not outweigh the additional information that is kept when using several parameters, reducing the amount of choices per attribute. If required for export purpose, it is always possible to extract layer information out of the common element attributes.

**Connect Properties**

Each building element can connect one or more of it’s properties to other building elements. This concept is implemented in our prototype and is similar to parameter wiring in the animation applications 3ds max (www.discreet.com/3dsmax) and Maya (www.alias.com/maya). Ideally, such a connection can be embedded with some intelligence through the use of a parameter expression, rather then a simple equality connection, but that is not elaborated in the current state of the prototype.

In theory any property can become connected to any other property. The designer can embed design intentions directly into the elements, rather
then relying on interpretation of markup information.

The resulting effect on transitions

Scale Level transitions will be facilitated through the use of a grid of reference points. Masterplan elements, spaces and regular building elements can share reference points, staying connected at all times, even when modifications are performed on a different scale level. This doesn’t imply that there will be no conflicts during modifications, but that should stay the responsibility of the designer, rather then of the software. By laying out our design elements on a common grid, the implicit relations between these elements are maintained. The grid can be seen as a hierarchical grid, to which the different scale levels connect. We propose the grid as a design entity, generating additional reference points to which building elements can connect.

The connection of properties is a way to store relations between elements. In contrast with the sharing of reference points, connection properties also allow the storage of non-graphical relations. The current approach in the prototype is directly editing these relations, but in a more elaborate version of the prototype, this can be partly hidden from the GUI.

The Classification Information is important for a generic access to element information, regardless of their actual implementation. The classification of an element allows transitions to be automated or to be proposed to a designer in a wizard-like format, where only the relevant choices are presented. It also sustains element filtering, which is necessary in the preparation steps for a transition.

A design phase transition is mostly performed through the additional of element information, e.g. a simple planar element can become a cavity wall, with a certain composition. To perform this transition might involve modifying elements. This is obtained by decoupling elements from their topology, allowing elements in the IDEA data structure to apparently switch type at runtime. A reverse design phase transition, however, can not involve another type switch, since this would lead to loss of element information. This reverse transition is therefore handled through the representation settings. These concepts are explained in (Hendricx, 2000), where the Core Object Model is described.

A scale level transition will have to create additional building elements, if required. When a Masterplan block is set up, it doesn’t contain walls or floors. They can be added initially, in the transition to Space and Block level, the former creating spaces and rooms and the latter creating actual building elements. The reverse transition depends on the potential existence of a Masterplan block. This could be generated as the outline of the building, containing reference points from all external walls. If, however, these building elements had already been generated from a prior Masterplan Block in the first place, we need to update the existing one.

The grid of reference points forms a connection between different scale levels. Most modifications can be reduced to manipulations on these reference points, e.g. transformations. Other modifications, such as choosing a different composition for a planar element, should not influence the other scale levels.

Conclusions

Although support for design phase and scale level transitions is partly enabled in current architectural CAD applications, support of the designer’s workflow is limited. This paper elaborated features to better support these transitions, by describing the development of an integrated design environment for architecture (IDEA+).

References