StructuralComponents 4: Conceptual building models with structural design justification

Arie BOVENBERG¹, Anke ROLVINK¹, Jeroen COENDERS¹

¹Delft University of Technology, White Lioness technologies
Stevinweg 1, 2628 CN Delft, the Netherlands
a.c.bovenberg@student.tudelft.nl, ankerolvink@white-lioness.com, jeroencoenders@white-lioness.com

Abstract
The adoption of computation in the building industry has led to the development of a range of tools intended for design. Although commonly used for detailed analysis (FEA) and documentation (CAD/BIM), few tools address the needs of the early design stages for flexibility and quick, insightful feedback and documenting the justification of design choices. This absence of suitable tools leads to missed opportunities for computation in early-stage analysis and documentation of design progression and rationale.

StructuralComponents has been developed as a tool for the early conceptual stages of structural design, until recently mainly focused on high-rise structures (Rolvink [13]). This paper describes the next steps of the development of a more generally applicable building model which supports composition of design justification (Coenders [5]): the collection of analysis models, reasoning, mental models, scenarios, etc. which form the backing of the choices which form design. This paper discusses the proposed building model and its underlying concepts, based on research into design theory, modelling approaches, and structural design practice as well as its application in practical examples.

As a proof-of-concept, a prototype has been developed which demonstrates this modelling approach is flexible and versatile enough to document various types of designs, processes and reasoning. Opportunities for future developments include extending functionality for geometry and visualisations, as well as investigation into standardization.

Keywords: conceptual structural design, computational design tools, design rationale, building information modelling

1. Introduction
The process of designing a structure is a complex process, with no predetermined “optimal” result (Chapman [4]). The design process runs from the early design stages, where there is a great deal of flexibility but a lack of information, to the later stages, where this situation is reversed (more information, less flexibility). The effect of this trend is that engineers often have to make influential
decisions when information needed to make these decisions is lacking. Making changes later in the design process is often costly.

Most design tools, however, focus on detailed analysis and documentation and are therefore primarily suited for the later design stages. In the early design stages, designers often apply simplified models which are more flexible and insight-providing. StructuralComponents is a software tool for the early design stages which addresses these needs. As a design tool, its purpose is not to perform design itself (as is the case with expert-systems or optimisation tools), but to provide a set of tools to complement existing design techniques. This concept is based on the Structural Design Tools approach (Coenders and Wagemans [6]). In this way, StructuralComponents aims to provide a new generation of digital sketching paper: a design environment where the engineer can develop his or her design justification or concept, select tools to perform analysis and where computation serves to perform repetitive tasks, leaving the designer with more time to focus on the creative design process.

Until recently, StructuralComponents was only focussed on the design of high-rise structures. In addition to extending the functionality to more common typologies, this research focuses on the development of a conceptual building model in which to represent the structural design justification: not as a single model or analysis method, but as a conceptual “design story” of various models, simplifications, schematisations, and scenarios (Coenders [5]).

2. Methodology

The approach for the development of the conceptual building model is partly research- and design-driven. Research into (structural) design and conceptual building modelling yields initial assumptions and requirements. The development of the central concepts of the building model has been performed based on Design Thinking (Brown et al. [3]) and Extreme Programming (Shore and Warden [15]): utilizing an iterative development process, implementation of each concept offers opportunities for feedback and re-evaluation of assumptions.

3. Research

3.1 Design

As mentioned earlier, design is a complex process. It is often described as iterative, cyclic or chaotic, and there have been many attempts to map the sequence of events in the design process. Although design often consists of separate activities such as analysis, synthesis and evaluation, the process generally does not occur in a fixed order (Lawson [10], Visser [18]). Designers, therefore, tend to focus on proposing solutions, rather than extensive problem analysis (Cross [7]). Another important characteristic of design are parallel lines of thought: several alternatives are often considered without an immediate attempt to resolve them (Lawson [10]).

When advancing the design, the designer works with his past experiences as well as knowledge of the current subject. This process can be seen as “reflection” (Schön [15]): each representation allows for reflection and development of design in a new direction. These new developments are then available to be re-interpreted (Gero [9]). Representations are integral to the design process: to describe problems and solutions requires representation. In engineering design, a variety of representations are
necessary describe analytical, geometric or quantitative models (Dym [8]). Design also requires models at different levels of abstraction and scale, and the designer frequently switches between them. Structural design justification consists of several different elements. One important aspect are (analysis) models. An important characteristic of structural design is the use of simple as well as complex models: simplifications can be used to make quick evaluations and gain confidence in results, while complex models can provide extra insight (Coenders [4]). Related to alternatives, cases and scenarios are another main aspect of the structural design justification, and are used to illustrate how the structure deals with a particular situation. Another characteristic is the progression from preliminary to detailed design: preliminary design is mostly concerned with the overall feasibility of structural systems (Baker and Fenves [1]); often only later in design every individual member is defined.

3.2 Conceptual building models
Most models in engineering practice focus on representing a structure precisely and in detail, for the purpose of (FEM) analysis or for representing the “ideal” structure for construction through documentation (CAD/BIM). As described earlier, these models are not very suitable for the early stages of design:

- In the conceptual design stage, precise information is often unavailable and exact results are not required.
- Constructing and modifying a detailed model is time-consuming. This is impractical in the early design stages when design changes are frequent.
- These models are limited in their ability to describe underlying design reasoning.
- Structural engineering is focussed on multiple boundary cases or worst-case scenarios while the above mentioned models often focus on the single “ideal” to be built scenario.

A growing trend in the building industry is Building Information Modelling (BIM). In the broad vision of BIM, computation facilitates information management throughout the entire design process, but this is currently not the case (Coenders [5]): in practice, BIM often means a single 3D-model in which information from various disciplines is combined or multiple singe-disciplinary models combined to a single coordination model. An important advantage, however, of this type of BIM-model is its object-orientation: while traditional CAD models often only contain geometric data, a BIM model consists of objects, which can contain various properties and design information. This allows complex models to be richly populated with information. However, because these detailed 3D-models require precise information and are complex to build and modify, their advantages for the conceptual design stages are limited (Sacks and Barak [14]).

Although most building models are aimed at the later design stages, a few representations have been proposed to support the early design process. One such development is SEED (Rivard and Fenves [12]), a software environment to support the early stages of building design. SEED’s modelling approach is object oriented, but aims to support design evolution as well as to provide a standardized building model to allow for case-based design, allowing designers to apply knowledge from earlier designs. Objects in the model are organized hierarchically and relationships can be defined between them. This conceptual model provides a standardized top-down hierarchical decomposition. This approach has been adopted by several other design tools (Mora et al. [11]).
Although this hierarchical division suits a top-down design process, enforcement of a standardized structural decomposition limits the model's flexibility: while the standard is extensible, it is by its nature not as innovative as design solutions themselves (Coenders [5]). In order to support conceptual design, it is important that models not inhibit the designer’s creative process: in order to provide sufficiently versatility, conceptual models should not focus on strict definitions, but allow room for interpretation. This can be achieved by implementing very generic components, incorporating both structured and unstructured data, and allowing different views within the same model (Turk [17]).

4. Results

4.1 Requirements
The conceptual building model must support:

- A unique process structure
- Multi-dimensional design problems and integrated solutions
- Design progression through reflection
- Parallel alternatives
- Different types of representations
- Representations at different levels of abstraction and scale
- Opportunistic switches between representations
- Various analysis models
- Cases & scenarios
- Evolution from preliminary to definitive design

4.2 Concepts
To structure the elements of the structural design story, the conceptual building model is subdivided into four domains. The component-domain describes the design subjects and contained design information: for example, a beam and its length. In the reasoning-domain, the justification behind this design information is described: for example, the reason behind a parameter value, or a design decision. The modelling-domain contains models and calculations used in this reasoning: for example, a beam model. The geometry-domain is used to describe component dimensions, shape and fit.

Concept 1: “Blank slate” components
The first and central concept of the building model is that of generic, customizable components: components which are not characterized by strict definitions, but are first and foremost adaptable to design needs. The goal is to allow for representation of design-specific components, as well as increase model flexibility. This is implemented by allowing components to represent any structural feature which contains design information (attributes). This not only allows straightforward modelling of nonstandard structural features, but also modelling at any level of abstraction or scale. Components can thus include traditional components (beam, column, etc.), high-level components (truss, core), and also project-specific components such as custom systems or groupings (see Figure 1).
Figure 1: the significance of custom object in describing building design: BIM and other traditional tools deal with low-level objects such as beams and columns. Tools for conceptual design include definitions for higher-level objects. However, often much of the “intelligence” of the design lays in components which are unique to the situation. Example from de Boer and Henkens [2].

**Concept 2: Hierarchical organization**

The second central feature is that of hierarchical component organisation: components may contain subcomponents. This allows representation at various levels of abstraction and scale in the same model. This is implemented by allowing a component to have any number of “parent” or “child” components. This allows breakdown of components as needed for each design phase and makes it possible for one model to represent both a conceptual design – where systems and representative members are designed – and detailed design where each element is checked.

**Concept 3: Reflective design reasoning**

The third main concept concerns modelling of design changes or additions as the result of reasoning from a certain design state or context. This methodology has been applied by allowing new attribute values to be added through reasoning, which in turn may be based on other attributes. This way of modelling reasoning aims to allow for design parameters to play a role in several lines of reasoning, as well as allow to integrated solutions.

**Concept 4: Analysis models as automated reasoning**

In the same way that manual logic (reasoning) can be applied in design steps, it is possible to apply models as a form of automated logic. The application of models is an important tool in design progression; many design parameters are the result of model outcomes. The goal is to allow the designer to determine what type of model to use, and when to use it. This functionality is implemented
by allowing the designer to add a model to any reasoning item. Input for the model is gathered from the reasoning context, and model output determines the value of the outcome.

**Concept 5: Alternative values per attribute**

As design information is stored in attributes, alternatives are represented as alternative values for this attribute. The StructuralComponents building model allows the designer to assign several different values to an attribute, with one being “active”. A value can be a reasoned outcome, but also an attribute from another component. The designer can switch between these values, and the model will respond to these changes. The idea of multiple values per attribute also makes the application of various models more straightforward by allowing the outcomes of various models to point to the same attribute.

**Concept 6: Scenarios and alternatives as component states**

Scenarios and alternatives are both represented as a component in different states. In each state the component’s attributes may have different values. In the StructuralComponents building model, the designer can create any number of alternative variants. Different alternative values can be activated and associated with this variant. When referencing this component, a specific variant or the “default” variant can be specified.

![Diagram](image.png)

Figure 2: Examples of components. Top and bottom left: a beam and floor component with (automated) design reasoning. Top-right: alternate values for an attribute. Bottom-right: one variant of a floor component.
4.3 Implementation
The conceptual building model has been implemented in a prototype application developed on the .NET platform. Aside from the building model, the prototype implements a user interface and a parametric engine. The parametric engine serves to keep the information consistent throughout associated elements of the building model and has been developed to support alternative values and component states.

5. Discussion
5.1 Validation
The concepts of the building model have been validated through application in several pre-determined use-cases, which illustrate the required functionality. Figure 3 shows a condensed example: the analysis of a “core” stability element in a simple building model. The examples have shown that the central concepts of the building model sufficiently address the high-level requirements. Notable limitations are the absence of geometric and visual representations, and lack of advanced analysis tools.

Figure 3: Analysis of a “core” component. The value for total vertical load is dependent on whether the office tower is constructed in steel or concrete. A “soft” foundation scenario can be used to obtain a sufficiently conservative stability calculation. For the stability calculation itself, both a simplified model and a more exact model can be used. Sketches obtained from the preliminary design of the RAI Elicium in Amsterdam (Arup, 2009).
5.2 Background

Compared to design tools in current engineering practice, StructuralComponents takes a fundamentally different approach. Firstly, by providing a tool for early stage-design exploration. Additionally, design reasoning, lacking in most traditional models, plays a central role in the StructuralComponents building model. Compared to existing tools, however, the developed prototype still lacks functionality for geometry and visualization.

As mentioned earlier, StructuralComponents has been developed based on the Structural Design Tools approach (Coenders and Wagemans [6]). This research continues in this direction by providing a tool which is highly adaptable and controllable. This is achieved by letting the engineer determine the components and organization of the building model. Also, because the user can define their own underlying reasoning, engineers can use their insight into structural behaviour to select the correct means for design justification, instead of a “one-click-calculation”.

5.3 Next steps

This section discusses possible development directions following observations of the prototype validation.

5.3.1 Opportunities for standardization

Although generic components are essential to cover unique design situations, some measure of standardization is required to allow for interoperability and prevent repetitive definition of routine situations. It is important, however, to implement this in a way which does not compromise the benefits of highly adaptable components. One way to do this is to implement standardized component interfaces and attribute types. These interfaces would define certain required attributes. Design flexibility would be maintained, as the designer is free to define which interfaces to apply, the reasoning and models behind these properties, as well as define any additional attributes as needed.

5.3.2 Reflection and visualization

The conceptual building model implements a simple type of design “reflection”: reasoning from a state of component attributes. In design practice, reflection often takes many forms, often aided by various (visual) representations such as drawings or sketches. Supporting this type of reflection requires not only visual representations, but also a broader conceptual model for design reasoning context.

5.3.3 Geometry

It is clear that geometry is an important aspect of design, especially in the later design stages. One way to implement geometric representations in an organized manner is to support definition of a geometric breakdown, along with the hierarchical component breakdown. In this way a component could contain a geometric representation of the composition of its subcomponents. A geometric representation could be built up from nodes and lines, with each linked to a component (e.g. a beam component is associated with a line and a connection component with a node).
5.3.4 Other enhancements
Possible enhancements include better handling of complexity through replication, implementing reasoning behind alternative choices, expanding functionality for alternatives and scenarios, component validation through automated checks, and links to more advanced analysis tools.

6. Conclusion
The study has been set out to develop a new implementation of the StructuralComponents design tool, focussing on representing the various aspects of structural design justification. The study sought to investigate the elements of structural design justification, and develop a conceptual building model in which to represent these elements.

Core characteristics of the design process and the structural design justification have been identified. These have established high-level requirements for modelling design in general, representations, and structural design. A conceptual building model has been developed to address these requirements: several core concepts have been proposed, partly derived from existing concepts and partly evolved from feedback during iterative development. As a proof-of-concept, a prototype has been developed which implements these concepts. Application in practical examples has demonstrated that these concepts address the high-level requirements, and have given insight into directions for future research.

As a significant step towards a comprehensive model of structural design justification, the developed building model offers several opportunities. Firstly, by making design reasoning explicit, it could reduce miscommunication, and potentially promote re-use of design knowledge. Also, the flexible nature of the model allows for application of computational analyses earlier in the design process, leading to more informed decisions. Additionally, the inclusion of scenarios allows for more targeted use of optimisation.

Acknowledgements
This research is part of a collaboration of the faculty of Civil Engineering and Geosciences of the Delft University of Technology, the BEMNext Laboratory, and White Lioness technologies. The authors would like to thank Prof. dr. ir. J.G. Rots. and Ir. J.W. Welleman (department of Structural Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology) for their feedback during research and development. The authors would also like to thank Sander Hofman (Arup Amsterdam) for his feedback and for providing the practical examples for validation of the prototype.

References
Chapman, W., Rozenblit, J., and Bahill, A., System design is an NP-complete problem, in *Systems Engineering*, 2001; 4(3); 222–229.


Visser, W., Designing as construction of representations: A dynamic viewpoint in cognitive design research, in *Human–Computer Interaction*, 2008, 21(1); 103–152.