Chapter 7
Feedforward

ABSTRACT

Feedforward refers to the transition from briefing to designing in ways that enrich the design with information from the brief (one assumes that a better-informed designer can achieve higher performance than an uninformed one). The first possibility for feedforward that is discussed is the use of the requirements graph in designing, including the development of schematic designs in Visio. After that, another form of feedforward is examined: the connection of the brief database to design representations. This is considered with respect to the various facilities on offer in CAD (AutoCAD) and BIM software (Revit and AutoCAD Architecture).

STARTING POINTS FOR DESIGNING

Several myths concerning design creativity suggest that an architect (a great one, at least) can produce a good, innovative, or even impressive solution to a problem just on the basis of a deeper, almost mystical understanding of the problem. While intuitive, creative leaps are not to be underestimated, it pays to prepare by analysing a problem explicitly and creating an informative and informed framework for designing. The brief with its goals, requirements, and constraints is an obvious source for this framework. The important thing is to express brief information in a practical, usable way that helps us comprehend and solve problems in an unambiguous manner. The brief and its satisfaction may offer no absolute guarantees for a good design, but it is a usable basis for a design that is minimally adequate for its purpose.

There are two complementary paths we can follow. The first involves the use of the requirements graph. The relationships and clustering organization indicated in this graph are important for the design and their graph representation allows for direct comparison to the design. The second is the enrichment of design representations with brief requirements (i.e., the connection of CAD or BIM files with the brief database so as to

DOI: 10.4018/978-1-4666-4647-6.ch007
Feedforward provide a detailed and comprehensive picture of what the design should satisfice, including criteria for evaluating the design). In practical terms this means that when dealing with the accommodation of an activity you can see all requirements on the activity in the brief and compare them to corresponding spatial properties and relationships (e.g., required minimum space height to actual space height in the design). You can use such requirements to guide design actions, evaluate their results, or adapt and refine their products.

Closely related to the second way of linking the brief to the design is the transfer and application of brief constraints to a design representation. However, by virtue of their combination of feedforward and feedback as well as their immediate and automatic impact on design behaviour, constraints form a distinct subject and are therefore covered separately in this book (Chapter 9).

**GRAPH AND DESIGN**

Using graphs to represent a brief is a clear first step in its translation into a spatial form and hence a useful bridge to designing. It is therefore not surprising that similar representations have been used as a first step in attempts to automate design processes, as mentioned in Chapter 4. However, there are serious limitations in such attempts, primarily due to the highly abstract character of a graph, which eliminates most of the geometric aspects we need in a design (see section From requirements graph to floor plan in Chapter 4). So if you use a requirements graph as a starting point for designing, keep in mind that it should be primarily used in comparisons to spatial design representations – do not try to make a design around a graph.

Importing a graph you have made in Visio in CAD or BIM software is quite easy if you export it first as a DWG file. An increasing number of programs can also insert a Visio file (e.g., as an OLE object, but in most cases there is little you can do with such an object except view it). If you import a DWG version of a Visio graph in, for example, AutoCAD, vertex symbols are translated into polylines (even if they were originally circles). Straight edges become lines and curvilinear ones splines; in short, all turn into objects you can process in AutoCAD. Visio layers are translated into AutoCAD layers, giving you easy means of clustering imported graph components.

In Revit, things are similar with two notable exceptions. The first is that an imported DWG file becomes a single symbol. In order to be able to work with the graph, you have to explode it first (both full and partial explode have the same effect). After exploding you encounter the second difference: Some vertex symbols are exploded in several graphic objects. In particular, the circular vertex symbols we have used in Chapter 4 become two half-circle arcs (interestingly, in AutoCAD they become a polyline consisting of two such arcs – a common problem with exporting circles from vector graphics, even in the age of interoperability).

The main limitation of imported graphs in CAD and BIM software is that the edges are no longer dynamic connectors as in Visio but mere lines that do not glue to vertices. Consequently, Visio graphs in CAD or BIM programs are not as easy to manipulate (e.g., when overlaying a graph on a design so as to analyse and evaluate conformance to brief requirements). The obvious alternative for such comparisons is to import the design in Visio, where you can adjust the graph with more ease. AutoCAD drawings can be imported directly (although Visio is not always compatible with the latest DWG version), while Revit models must first be exported as DWG files. You can scale the DWG drawing down to customary Visio sizes (A3 or A4) or choose to enlarge the graph to the size of the imported drawing.

In either case it is advisable to place the drawing on a background page so that you can use multiple graph overlays (e.g., consider different interpretations of the brief or brief adaptations following a design proposal). You can also have different