DESIGN AND CONSTRUCTION AS COMPUTER-AUGMENTED
INTELLIGENCE PROCESSES

GERHARD SCHMITT
Architecture and CAAD
Swiss Federal Institute of Technology, ETH Zürich
schmitt@arch.ethz.ch

Abstract. The purpose of design is no longer restricted to the construction of physical artefacts, as the emergence of virtual architecture and information architecture demonstrates. Methods and instruments developed for the design of physical architecture are also useful for the development of virtual architecture. Vice versa, important lessons can be learned from design and construction methods for information architecture. Certain quality criteria - such as the Vitruvian firmitas, utilitas and venustas - apply to both kinds of architecture. The computer can directly augment selected tasks by combining human intelligence and machine reasoning to arrive at new design results - both for physical and virtual architecture. This augmented human design and construction intelligence will be a key for future design processes.

1. Physical and Virtual Environments

The emergence of virtual reality techniques has renewed the discourse about the relation between reality and virtuality. Physical and virtual worlds are subsets of the new reality (Fünfschilling, 1992). The physical environment consists of objects that are subject to gravity, that are three-dimensional and that can be felt or seen. Objects in the virtual environment are not subject to gravity, can be n-dimensional and may result in visual presentations. In the world that surrounds us, the importance of the virtual environment is increasing rapidly. The time may not be far away that some physical environments will need a protection similar to what national parks offer to natural or original habitats. Museum exhibits of physical instruments whose function is now superseded by virtual instruments are a first indication.

Operations in virtual environments require abstract instruments or software, whereas operations in physical environments require physical instruments or hardware. The boundary between the two types of instruments is becoming blurry: most of the physical artefacts are produced
by software driven hardware, and hardware collects data to develop software. Design and construction for the physical environment have a long history and a highly developed instrumentarium. The increasing use of computers has changed the design and construction processes and has lead to the discovery of a new design territory: the new material of this territory is information.

2. Design and Construction of Virtual Architecture

The information territory is a special subset of the virtual environment. In this territory, virtuality is the only reality. It is the environment in which the object and its representations can be identical, whereas in the physical environment a representation is always an abstraction. It is also the development and construction site for virtual architecture. Because of the identity of representation and object, design and construction of virtual architecture in the information territory are closely interlinked and can be achieved by the same person. This could lead back to the unity of architect and builder. Context and infrastructure of the information territory are radically different from the physical territory, yet the metaphors of the latter re-appear. Enabling concepts in the information territory are networks, data bases, interfaces, and agent technology.

2.1. NETWORKS

Networks establish connections between people, between computers and between people and computers. The World Wide Web creates a network of documents in the form of links on pages (HTML), or a network of objects in the form of links on elements (VRML). This is an opportunity to become aware of the fact that information is being linked, not just machines. The network should make designers aware of each others’ work and enable them to react to changes in the environment their work is contained in.

We use the WWW to teach these aspects of design. The WWW provides for the design and maintenance of complex interfaces and crosses the boundaries between machines of different architectures. We apply languages such as Java or JavaScript whenever we need to do more than simply turn pages or if we want to break out of the WWW interface restrictions.

2.2. DATA BASES

Data bases can act as building memories, containing all data and information related to the building. As such, the data base structure in the information territory is related to the physical and functional structure of the building in the physical territory.
Data bases help to control and manage the complexity of systems based on input from various users or groups of users. They also serve to extract different views from one set of data. As our systems are built using Internet technology, data bases are always placed on the server side and contain the structure, the skeleton of the teaching environment they serve.

2.3. INTERFACES

Interfaces establish the link between the representation of the machine and representations that humans can understand. The evolution from command line interfaces to interactive desktops will continue with the development of fully three- and more-dimensional interaction spaces between designers and machines.

Architects can contribute to better interface design, as their discipline already requires them to develop the ultimate interface - environments for people. Interface design, sometimes described as information architecture (Wurman, 1996), involves skills from different professions. It will gain more attention and importance in the future as information and data sets accumulated need appropriate processing and the results need better and denser presentations than today in order to highlight content and to distil underlying relations. In our experiments, we discovered that two basic categories of views on one set of data are helpful in teaching networked, decentralised design thinking (Wenz and Gramazio, 1996): The InWorld view is the view centred on one’s work with links to environments surrounding it. It is the work view, focused on the project and representing the view a user works in. The OutWorld view provides an overview of the entire system of activities and projects. It shows the connection between groups of students and may cluster works according to criteria, computed from life data. Both InWorld and OutWorld views have related, but less powerful analogies in the physical world.

2.4. AGENT TECHNOLOGY

We see agents as programs that operate on behalf of the user, that react to events and that can act autonomously. In our teaching environment, an individual agent accompanied each student from the beginning of the first semester in the form of a digital assistant, guide, representative, helpful spirit and communication beacon in an unknown environment (Engeli, 1996). Agents can also take on the role of presenters of information and mediators between partners (Stouffs et al, 1998), a necessity for establishing virtual firms. Agents have an important role in augmenting human design capabilities by taking care of low level or difficult tasks that are not directly in the competence or time budget of designers.
3. Support and Augmentation of Human Tasks

Physical design instruments support and augment human tasks - they give immediate feedback and they can stimulate thought. Virtual design instruments are more oriented towards memory support and augmentation. Networks, data bases, interfaces and agents are useful both for the design of physical architecture and virtual architecture, as the following three examples demonstrate.

3.1. AUGMENTING CONVENTIONAL DESIGN CASE BASES - PHASE(X)

Architectural case bases are fundamental for the development of new designs. We consider the selection and adaptation of complete cases as a counterpart to planning based on first principles of design. When working with cases, the size and quality of the case base as well as the capacity to select and adapt the appropriate case are essential for success. Our experiments with case-based design demonstrated that the visual browsing of cases or case elements is at least as important as the capability to search through indexed case properties (Dave et al. 1994). But it was not before the combination of the appropriate interface (PhP), network (WWW) and data base (MySQL) instruments, that we could prove the benefits of such a system over conventional approaches.

Figure 1. Two views of the Phase(X) database, containing more than 700 individual student projects. Projects ordered by authors and connections (left), and by authors, connections and time (right). Each horizontal line describes one distinct work or case, diagonal lines show connections between projects. The course begins with Phase 1 or time=0 at the left of the figure.

The purpose of Phase(X) is to teach fundamental principles of computer aided architectural design (see http://space.arch.ethz.ch/ws97/). By participating in this course, students become authors in a network oriented process that is administered by the system, but guided by the collective actions of the students. In each phase, the student works on a design from a different author. Immediately after a model is deposited in the data base, it
can be seen and checked out as a complete case by other students. This allows for a natural selection of the most attractive designs. The fundamental responsibility of the student is the analysis of the formal idea in the design product of another person, before she or he selects it for further work. At the same time, the system combines individual contributions into high quality design products with collective authorship. The value of the individual contribution is measured according to a range of criteria. One of them is the number of offsprings or memes that a design produces. Underlying the teaching of geometry and composition is the principle of Types and Instances (Madrazo, 1996).

Figure 2. Snapshot of the Phase(X) InWorld interface. Parent design from the previous phase on the left and children designs on the right of the object. Bottom: Display of data base content.

Students select from a large number of designs displayed in visual form. By clicking on the design, it appears in full resolution in the main window, together with its predecessor on the left, and, if already available, its children on the right. Students can then check out the design, work on it, and check it back into the data base from where it is immediately visible. The system consists of a web based interface, a data base that stores and organises the individual contributions, and the actual designs distributed throughout the network. Students interact asynchronously with the data base through their home page from where they observe their personal project status, order the exercise they are interested in and deposit the result of their design. This interface also allows for monitoring the evolution of the models.
Compared to earlier courses with identical CAAD content, but based on the traditional paradigm of working only on one’s own design throughout the semester, three observations were striking: In Phase(X), the process was (i) much faster, (ii) produced better results and (iii) in spite of the changing authorship, some designs kept their inherent identity from the first through the last phase. The analysis of these observations is not completed and we are exploring specific aspects in follow-up projects.

3.2. MULTIPLYING DESIGN TIME

With increasing globalisation and specialisation in the design and building industry, collaboration between partners in remote locations becomes crucial. Ideally, all of them could work on a building design at any place, simultaneously together (synchronously) or separately (asynchronously), with the current state of the design available to all team members. The Multiplying Time project allows simultaneous and continuous work on a
design or a set of designs through different time zones around the world (see http://space.arch.ethz.ch/VDS_97/).

The concrete task was to design a residence for a painter and a writer on an island west of Seattle, USA. In a first step, three partners from ETH Zürich, the University of Hong Kong, and the University of Seattle agreed on a common project. The interactive modelling program Sculptor by David Kurmann (Kurmann, 1997) was installed in all three locations to enable synchronous and asynchronous design. Data exchange was enabled through a database directly connected to the Internet, similar to that of Phase(X).

Students in Hong Kong started with the design. At the end of their 8 hour working day, they deposited the results in the common database in Zürich that could be seen by all partners through the browser interface. At that time, students from Zürich began their work and could thus base their decisions on the results achieved by their Hong Kong partners. They also placed their designs in the common database, so that students from Seattle were able to explore the designs from Zürich and Hong Kong by the time they started to work. In addition, video conferences took place every 8 hours, during which students could share and explain their ideas. The set-up thus created an intense global think-tank, operating 24 hours a day.

The Multiplying Time experiment demonstrated that it is possible to work from a common database, taking advantage of different time zones and special capabilities of particular sites: Seattle provided the site, Hong Kong the first design models, Zürich the modelling program. The resulting designs are of shared authorship, but the individual contributions are clearly identifiable, along with the evolution of the design. Precondition - as in Phase(X) - is the capability to modularise the design process and its results and the willingness to share design knowledge with project partners.

3.3. DYNAMIC INFORMATION SPACES - FAKE.SPACE

While Phase(X) augments the organisation and retrieval of cases and the Multiplying Time project helped to expand time constraints, fake.space augments the human capacity to build and traverse complex, dynamic information spaces (see http://space.arch.ethz.ch/ss97/).

fake.space is a node system, consisting of tanks - as containers for spatial descriptions and statements about space - and pipes - the connections between the individual containers. Both tanks and pipes are represented as HTML documents, but with different functions. Tanks describe a spatial situation using a combination of floor plans, perspectives, animations, interactive 3D models, light simulations, text, images and sound. Pipes have the function to demonstrate the content connections and the paths between spaces. This important role as a mediator between different spaces is achieved with text. If one decides to add to a space in the fake.space system,
one first has to add a pipe to an existing tank. After this, a new tank can be added. fake.space is therefore constantly growing. The InWorld view is that of moving through tanks and pipes. The OutWorld view is that of visually exploring the data base.

![Figure 4. fake.space OutWorld view with differently scaled views of the data base and selected node contents. The starting point of the course - time=0 - is in the centre.](image)

In a first exercise, students connect an interior space, represented as a model, and an exterior space, represented as a path description. A second exercise deals with insights and views to the outside, represented by 3D models and photographs. A third exercise focuses on the realistic simulation of light and atmosphere in the models to support the design story (Hirschberg and Streilein, 1996). A fourth exercise explores the type and quality of the circulation space. The last exercises are dedicated to animation, bringing visitors directly to interesting places and suggesting paths through the database.

Whereas in Phase(X) the database consisted of a simple table and the navigation was implemented in Perl and HTML, fake.space builds an additional layer on top of this structure. Collaboration with the work of other
students still occurs through the data base and in asynchronous mode, the need to attach a pipe to a tank with a meaningful continuation of a story ensures that the continuity of the entire system is higher than that of Phase(X). The impact of visual navigation within data bases has not been explored sufficiently in AI, but operating in the growing fake.space information architecture in the form of a nervous system is certainly more suggestive than entering data in a hierarchical file structure (Koutamanis 1995). The ability to see the overall information structure grow in the OutWorld view while the content of the individual tanks and pipes is visible in the InWorld view, provides designers with additional control and responsibility.

4. Firmitas, Utilitas and Venustas in Information Space

While it seems obvious that computers can assist the process of designing physical architecture, the Phase(X), Multiplying Time and fake.space projects demonstrate that information created in the process crystallises into another type of composition that we call virtual architecture. The working environment of the students has changed entirely: They no longer design on drawing tables, but on desktop and laptop computers; they do not produce drawings anymore, but more-dimensional models displayed on screens and in VR environments; they no longer need space to build and store physical models, instead they carry these models on a CD or DVD. The courses themselves have created a virtual architectural environment that only exists on the network, organised through the data base structure, and that can be experienced only through the interface we created. We claim that this information architecture can fulfil many of the quality criteria applied to physical architecture.

Firmitas or firmness is not an attribute exclusive to physical structures. The interface and structure of the Phase(X) data base can be used as a stable foundation for many other courses and applications, in analogy to Vitruvius’ original description of firmitas. In accordance with Vitruvius’ definition of utilitas or usability, information architecture can augment physical space: not only does it provide a new functional working environment, but by using different time zones, it can create virtual offices that operate around the clock. Venustas or beauty, finally, was originally defined as the correct proportioning of space and buildings. The information structures created in the fake.space experiment propose a different, yet balanced kind of proportioning and beauty.

The Phase(X), Multiplying Time and fake.space projects aim at the exploration of a collective human-machine intelligence. Instead of producing synthetic machine intelligence, our systems are based on the
collection of human intelligence during the design process. The result of this collected and collective intelligence can be shown, for example as the OutWorld views in Phase(X) or fake.space. The generation of those overviews is one of the most „intelligent” and interesting capabilities of a computer. Humans can interpret those maps and draw their own conclusions.

Thus, computer augmented design supports and expands the traditional design and construction processes. While it is a welcome tool for physical structures, it is indispensable for the design and construction of virtual architecture.

Acknowledgements

I want to thank Florian Wenz, Cristina Besomi, Fabio Gramazio, Urs Hirschberg, Patrick Sibenaler, Bige Tunçer and Malgorzata Bugajski for their innovative and enthusiastic work in the Phase(X), fake.space and Multiplying Time projects.

References

Wurman, R. S. (ed.): 1996, Information Architects, Graphic Press Ltd., Zürich, Switzerland