Generative Interfaces and Scenarios - Interaction in Intelligent Architecture

A framework for Computer Integrated Buildings

Pia Fricker[1], Mathias Ochsendorf, Kai Strehlke
[1]Chair for Computer Aided Architectural Design CAAD, Institute of Building Technology, Department of Architecture , Swiss Federal School of Technology, Zurich, Switzerland.
http://www.caad.arch.ethz.ch, e-mail: fricker@hbt.arch.ethz.ch

Abstract. *New media and modern building automation have a strong impact on contemporary architecture. So far one could regard built architecture as static. These new technologies introduce a dynamic impulse to architecture. The objective of our research and teaching work is to demonstrate the impact of innovative systems on architecture in daily usage while providing building automation, multimedia integration and facility management services in intelligent networked buildings. These technologies, as described in this paper are integrated in our second year course for students of Architecture. By designing an interactive graphical interface for the lab they were asked to create a spatial scenario as a self running Flash animation. Thus real space is merged with virtual reality.*

Keywords. *CAAD Curriculum, Human-Computer Interaction, Web-Based Design Building Automation, Generative Graphical Interfaces*

Figure1. The setup of the lab called Red Hell. On the left is Red Hell with its actuators and its sensors. A scenographical spatial design exercise is showing the connection oft the real space with its representation. On the right is the control panel for Red Hell and the control space.

Introduction and context

This paper introduces a lab that is used both in teaching and research. The lab runs on the level of a state of the art functional prototype and comprises a complex infrastructure. This lab is called "Red Hell" was constructed as a portal for the teleimmersion research project 'blue-c' at the Swiss Federal Institute of Technology (http://blue-c.ethz.ch, 2003, http://www.caad.arch.ethz.ch). A red-colored wooden frame defines the architectural space of Red Hell and embeds the complete technical infrastructure. This infrastructure is composed of multimedia and building automation elements.

The operation of buildings is one of the main research focuses of the chair of Computer Aided Architectural Design (CAAD) guided by Prof. Dr. Ludger Hovestadt (Blue-c). Next to its primary task in the blue-c research project, Red Hell works as a demonstration prototype providing services like automation, security, media control, access control and data completely through Internet based technologies. This paper describes the systematical approach we have followed to build up a software framework which controls all these services and the implementation of this framework within the architectural curriculum of a second year course in the field of CAAD.

The computer integrated building

Until recently, in building technology the whole area of building automation has been differentiated from the area of communication. In both fields dedicated devices have been used to provide the respective functions. Building automation comprises light control, heating, ventilation and air conditioning (HVAC), etc. The telephone, the radio and the television, etc, represent the communication components in a building context. Until the mid-eighties dedicated devices were used to provide these specific functions. In the nineties bus systems like EIB in Europe and LON in the USA appeared and allowed the development of multi-functional systems that group different functions of building automation in one system.

These bus systems could not handle the required data traffic needed in the communication area. The appearance of Ethernet and the rise of the Internet created the base for integrated communication systems. Based on this new technology, services like Voice over IP, video on demand or web radios could be established and spread worldwide at low cost.

Frank Duffy predicted in 1991, that both fields of building automation and building communication would merge together to a computer-integrated-building (CIB) beyond 1995 (http://www.degw.com/about/frank_duffy.html). This vision has not yet been achieved and there is no standard available. We have implemented Frank Duffy's concept in a working prototype. This prototype is used in teaching and research and is constantly extended. We use Internet technologies to integrate automation and communication into the building. The focus of our work is to show from the architectural point of view the possibilities and the impact to architecture resulting from such a system

System Architecture

The framework implemented in the lab is a system architecture based on three layers. The bottom layer consists of generic hardware. The middle layer encapsulates the hardware into software services which can be accessed by the top layer. The function of the top layer is to allow the interaction of the user with Red Hell.

Hardware – multimedia and building automation

The bottom layer contains generic hardware, a group of sensors and actuators. Typical sensors are buttons, motion detectors or input devices. Typical actuators are lights, video devices, audio devices or projectors.

We prefer devices which can be controlled di-

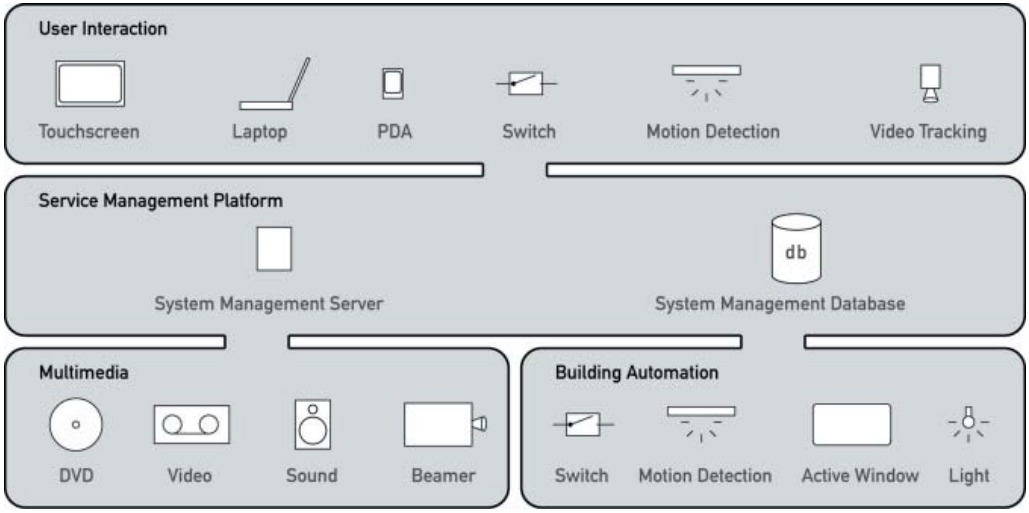

Figure2. System architecture

rectly via Internet protocols, based on TCP/IP. If a device does not support direct control via IP we use embedded systems to map the functions of the device and to provide them via IP to the system. For instance, Red Hell's projector can only be controlled via Infrared or RS-232. In this case we use a computer and its RS-232 interface to integrate the projector into the system. All devices of building automation are managed with a RaumComputer building automation unit (http://www.raumcomputer.com). The RaumComputer handles ten dimmable spotlights, five dimmable neon lamps and a window which can be electrically switched from opaque to transparent. The RaumComputer unit itself provides an TCP/IP interface of all managed devices to the system.

Service Management Platform

The middle layer, often called middleware, is a software layer. Software components control and manage the hardware elements of the bottom layer. As far as possible the communication of the software with the underlying hardware layer is based on Internet protocols, Internet technologies and bus systems. The approach used in the lab decouples the hardware elements from one another. A switch is no longer connected to a series of lights. It is a sensor in the hardware layer that sends only impulses to the middleware. The middleware receives the impulses from all sensors and is the only instance that controls the actuators.

This decoupling allows a free configuration and programming of the hardware and therefore enables an individual reaction to each impulse from the hardware layer. The middleware is written in Java and is embedded in the OSGi framework (http://www.osgi.org, Hall and Cervantes, 2004).

The middleware also provides an interface to the top layer. This service application interface uses the exchange of XML messages for the communication between users and the middleware. The framework allows multiple users to be connected to the system at the same time. Users send requests to the system. The middleware reacts and sends the appropriate control commands to the hardware and a notification message to all listening clients.

User Interaction

The top layer builds the interface between the

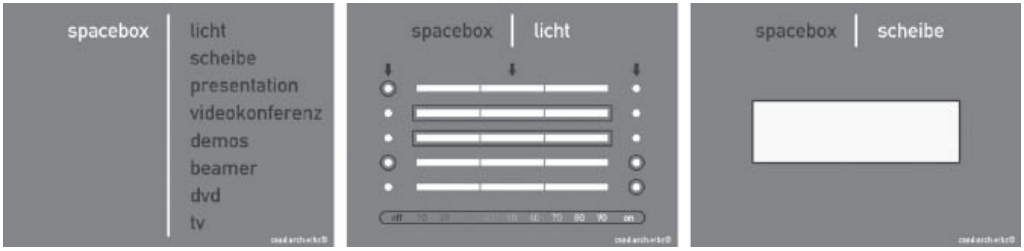

Figure3. Touch screen interfaces of the Red Hell

users of the architectural space of Red Hell and the system implemented in the space. Arbitrary applications and therefore an arbitrary number of users can be connected to the system at the same time. The user interaction layer defines a communication and interaction interface between the users and the system implemented in the lab. This communication interface is based on TPC/IP, one of the core Internet communication protocols (http://en.wikipedia.org/wiki/Transmission_Control_Protocol). For example, each time a lamp is switched, the impulse is transmitted to the middle layer. This event triggers the middleware to notify all connected/online user interfaces. As a consequence, all applications are up-to-date at any time. In an open and ubiquitous environment like this, an application can react to incoming notifications and interact with the lab individually.

This approach allows the development of generic interfaces for the architectural space. The complete functionality of all devices in Red Hell can be combined into, and accessed through a single user interface. The total control of the interface allows one to define the complexity of the interface. The projector itself has more than a hundred different functions. The interface designer can decide which of these functions will be accessible to the user. Another advantage is to be able to eayily combine different functions into scenarios within the interface. For example an interface button could be created which switches the projector on and dims all the lights down.

Although a single interface encapsulating all these functions can be realized, it is also possible to create a multimodal interface. The middleware has the possibility to simultaneously receive inputs from multiple interfaces as well as from different input types like video tracking, gesture recognition, motion detection and ordinary light switches.

The choice of XML messages as intelligible exchange medium to the middleware permits the detachment of the front-end completely from the system. This allows an approach for developing a wide range of possible front-ends for the interaction. These can be web applications or stand-alone solutions. A connection to the Internet and the ability to process XML data are the only two requirements a front-end application has to fulfill.

For Red Hell we have used Macromedia Flash because it allows us to create XML socket connections and it is a simple development environment for building appealing graphical front-ends (http://www.macromedia.com). Another advantage of Flash is that it creates interactive vector graphics, which can be played on various numbers of devices and operating systems. We have implemented an interface for a touch screen, which is installed as a control panel inside Red Hell.

Course Description

Besides research, the lab has been used for the last two years to teach every year second-year Architecture student in building automation. For the last few years the architectural computer field of work has been enlarged by the possibility

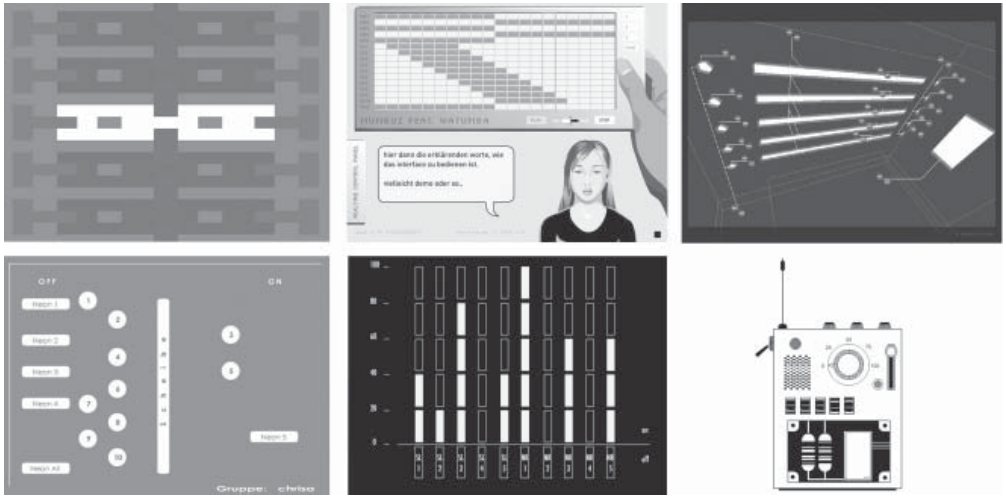

Figure4. Examples of student works 2004-2005 showing the different interfaces for Red Hell (Interfaces).

of programming space. The aim of the course is to teach the concept of the computer-integrated-building (CIB) to architects. We teach them the underlying technical principles and let them discover the architectural potential of the realized prototype – the Red Hell lab. The course is divided in two main parts: a theoretical and a practical part.

The historical background of the media in architecture and knowledge of the computer program Macromedia Flash MX and the messaging language XML are taught within the lectures.

The practical part consists of two tasks and is supported by weekly consultation hours at the chair of CAAD. To lead the students through the complex topic, these tasks are constitutive in their context and strongly related to the work field of an architect. As a first assignment, the students are asked to create an interface for Red Hell. The second exercise consists in a spatial scenographical design. Macromedia Flash is used as development tool to carry out both exercises.

First task: the interface design

The students are asked to design a graphical user interface for the lab. They are given a template which contains the XML communication with the Red Hell system. An example shows them how to control the devices in Red Hell. They can now communicate from their own computer with the actuators of Red Hell and control, for example, the light situation. The focus of the exercise is to think about interface issues. This comprises the design of an interface logic and the adoption of an aesthetical position. Within this task they are also trained in programming, by using the programming language ActionScript. The range of results varies from strictly graphical interactive images to ergonomic approaches and to playful solutions.

Second task: spatial scenographical design

A spatial scenographical design has to be done as second final exercise. This is achieved in a self running Flash animation which uses the lab as a stage and which is projected on the screen inside Red Hell. In this exercise we want students to grasp the potential of the idea of the computer-integrated-building. They have to play with all the devices of the space and create a dynamic, temporal and spatial scenario. Two technical approaches can be taken to realize this task.

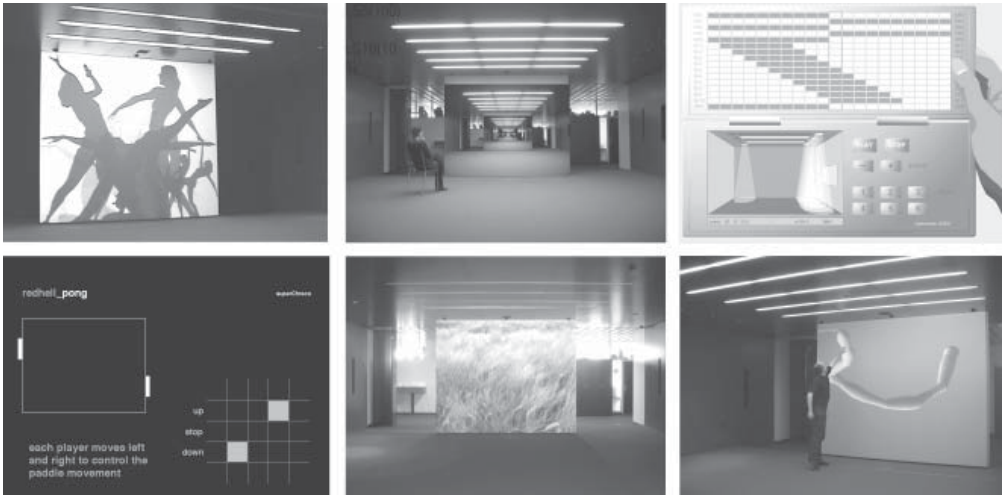

Figure5. Examples of student works 2004-2005 showing the different spatial scenarios for Red Hell (Scenarios).

The first approach is to create an ongoing animation and send commands to the system on defined key frames. In this way it is possible to design an animation and, at the same time, control the different devices inside Red Hell. Most students take this approach, generating a wide range of spatial scenarios. Some students represent the lab in their animations and merge therefore the real space with its virtual representation. Other students create a small story and include the space of Red Hell into this narration.

The second approach is to work with the notifications sent by the middleware which are broadcasted constantly to all listening clients. In this case the notifications are controlling the flow of the Flash animation. Any change in Red Hell would alter the projection displayed in the space. These scenarios could than easily be controlled from the outside, for instance, with the touch screen interface which is permanently installed in Red Hell. An example of this approach is the work of a group of students of who mapped the status of each light with the image of a dancer (figure 6). The light's intensity was translated into the alpha value of the correspond image. In this way each light condition would generate a different image on the projection screen. Another approach is the usage of the tracking system. By using the cameras, integrated in the lab, a system is developed to locate each position of a person or any item in Red Hell. This information is sent to the Raum-Computer and can be used by the students. Applications of this technique results in student work which integrate the tracking system in a spatial game, where the output is constantly shown on the projection screen. This approach is a realtime interaction with the given space. The tracking of persons make up another possibility for users to interact with the system. Applications can scan the positions of persons and depending on their position, for example, control the lights or change the audio settings.

Conclusion and outlook

The systematic approach to merge media and building technology is one of the main research topics of the Chair for CAAD. It is linked to several ongoing research projects. We believe that this concept of a computer-integrated-building will considerably extend the potential of architectural

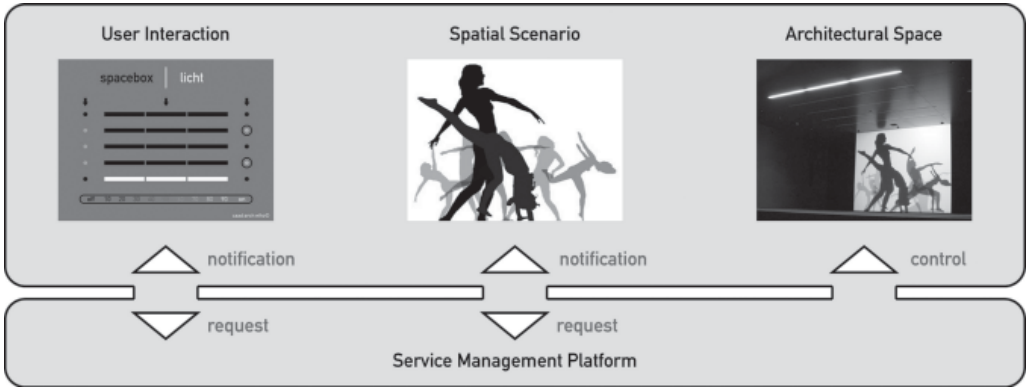

Figure6. Request and notification of the connected interfaces and spatial scenarios

design. The architect will be able to include spatio-temporal considerations into a design project. Architects will have the facility not only to design physical spaces but also to design the interfaces for them.

Acknowledgements

We would like to thank all the architectural students of the second year course in CAAD 2003/04 and 2004/05 for their participation and their work. A special thank-you goes to Christian Spagno from the center of product design who helped with the construction of the Red Hell and to Prof. Dr. Ludger Hovestadt head of the CAAD chair at ETH Zurich.

References

Blue-c : Gross M., Wuermlin W., Naef M., Lamboray E., Spagno C., Kunz A., Koller-Meier E., Svoboda T., Van Gool L., Lang S., Strehlke K., Vande Moere A., and Staadt O. 2003. Blue-c: A spatially immersive display and 3D video portal for telepresence. In Proceeedings of the ACM SIGGRPAH 2003 Annual Conference. ACM Press, New York.

Hall Richard, Cervantes Humberto. 2004. An OSGi Implementation and Experience Report. In Proceedings of CCNC 2004. Las Vegas, Nevada USA

Interfaces: The Interfaces shown as examples have been designed by Basil Georg Spiess, Marc Zürcher, Tobias Klauser, Tim Sergej Kop, Simon Martin Edelmann, Boris Gusic Daniel Abraha, Michael Martin Bühler, Willy Urs Stähelin, Christoph Kühne, Salome Fravi, David Dalsass, Deborah Wyss, Lorenz Stieger, Nils Wagner

Scenarios: The spatial scenarios shown as examples have been created by Tobias Klauser, Tim Sergej Kop, Philippe Markus Lacher, Lukas Peter Raeber, Reto Giovanoli, Franziska Pfyffer, Damaris Baumann, Sabine Laura Herzog, Julia Annika Sulzer, Fabian José Valverde, Christopher Stephan, Gregor Bieri, Nino Soppelsa, Michael, Gugg