Complex Multiplayer Urban Design System – Concept and Case Studies

Tomasz Jaskiewicz

Delft University of Technology, faculty of Architecture, Hyperbody department
Julianalaan 134, 2628 BL Delft, The Netherlands
t.j.jaskiewicz@tudelft.nl

Abstract. This paper explores the idea of creating a software and hardware system supporting collaborative urban planning and design. It demonstrates several working case studies of various parts of such system. Using these examples a selected strategy for a computer supported cooperative work for the field of architectural and urban design and planning is illustrated. Proposed strategy is part of the Protospace system and laboratory development at the Delft Univesity of Technology, faculty of Architecture.

1 Introduction

Urban planning is an inherently collaborative activity, where multiple decision makers, stakeholders or even general public may participate. Numerous, complexly interlinked factors influencing the plans are involved in the project process. It is not uncommon for urban plans to develop into several versions and multiple scenarios. Creation of an urban plan is never a predetermined process. Urban plans and strategies are often revised, corrected or even entirely replaced before their final implementation [1]. This is primarily due to the very high degree of complexity of urban systems, resulting in significant unpredictability of occurring activities and effects that interventions into such systems may cause [2]. It is also acknowledged that urban planning can be approached as an act of creating and/or modifying complex (adaptive) systems [3].

The state-of-the-art in design support systems in the field of urban planning is underdeveloped in comparison to architecture or other design fields. Design support provided by existing systems is very limited and facilitates only selected tasks. GIS [4] (Geographical Information Systems) applications are relatively common tools that can be employed to map various features of urban and other plans to geographic locations. Nevertheless, basic drafting software such as Autodesk Autocad or Adobe Illustrator, or even manual drafting techniques are to this day the most widely used tools for creating urban plans, typically accompanied by vast textual descriptions. All those approaches leave collaborative aspects of the design to take place without any form of computer support.

Many attempts were made to increase the employment of digital technologies in urban planning in order to facilitate dealing with the complexity of those undertakings. The first group of these attempts includes development and use of spatial decision support systems (sDSS) [5] aiming at improving the collaborative aspect of urban

H.X. Lin et al. (Eds): Euro-Par 2009 Workshops, LNCS 6043, pp. 300–309, 2010.

planning and design. The second group brings together extensions of traditional CAD (computer aided design) drafting techniques or GIS systems, by actively involving agent-based simulations, parametric urban modeling, genetic algorithms, various applications of neural networks, data mining and many other techniques. Nevertheless, no tools or standards in either category have yet been commonly accepted by the wide community of urban planners.

Case studies presented in this paper are aimed at investigating possibilities of a synergy between these two trends, while allowing further exploration of new possibilities on both tracks by pursuing an integrated approach towards creation of an extensible urban design support system. Its development is based on design system architecture formulated by the team including the author and other members of the research group Hyperbody in the scope of the ongoing Protospace [6] project.

Protospace constitutes of a software system (Protospace software) and physical environment (Protospace lab) that are being developed to facilitate collaborative designing in architectural and urban design context, using novel computer design tools and multimodal interfaces. Certain collaborative design activities of Protospace, such as design sessions involving selected specialists may, happen in the physical space of the laboratory. Nevertheless, there is large demand for facilitating online connectivity to the design model for a wider group of specialists, authorities and last but not least the general public.

2 Approach Strategy

Most commonly, urban and architectural design systems are based on hierarchical models, where hierarchies follow the scale of (usually nested) design components. Traditionally, the main canvas for such systems is established by the meta-hierarchy of a: region - city/landscape - building/street/square - interior/finishing/furniture. This approach requires revision, because in reality dependencies between design components, occurring also across different scales, are bidirectional [7] and form multiple feedback loops. For that purpose investigations were made into the concept of behavioral modeling in urban and architectural design. Behavioral modeling [8] involves creation of virtual models that are composed of multiple autonomously operating components, dynamically related to each other. In this way models can be created that: a) exhibit dynamic properties b) may natively include agent-based systems c) their different components can be modified or manipulated simultaneously, without concern for the hierarchical dependencies d) are open for further extensions, also throughout a particular design or planning process. At the same time, however, such strategy requires substantial amount of computing power, exponentially increasing with the amount of elements and users involved in the design process.

In this context, behavioral modeling needs a larger system to be embedded in. Due to the nature of urban design and planning, such models need to be formulated collaboratively and need to be connected to other models, depending on the specificity of the project. In that process, ideally all participants would be present at the same location. In reality this is often not viable and, therefore, a wide range of possibilities for remote on-line access to the design model is required.

As a working concept, it has been agreed that urban design system should support the entire process of collaborative and multidisciplinary design. It should allow for parallel work on the main design model, real-time connectivity to other models, and it should provide a possibility of creation, simulation and validation of multiple design variations. The design model should be open for extensive modifications and adjustments throughout the entire design process and after project completion. In addition to that, it was decided that all urban activities should be performed in a real-time navigable 3d environment instead of a two-dimensional one, the latter being still commonly practiced in urban design and planning.

3 System Vision

Proposed solution model for the problem consists of three generic layers. The server(s) host all project data. Various tools (client applications) connect to project database. Access rights vary among them and depend on their role in the system. Client applications may include web applications, dedicated specialist tools or collaborative design support systems dealing with multiple user teams. The third layer includes a flexible set of interfaces that can be used to control client applications. Typical mouse/keyboard/screen interface can be replaced by CAVE (cave automatic virtual environment) systems, alternative pointer devices, gesture and speech recognition and other interfaces facilitating work in a collaborative environment.

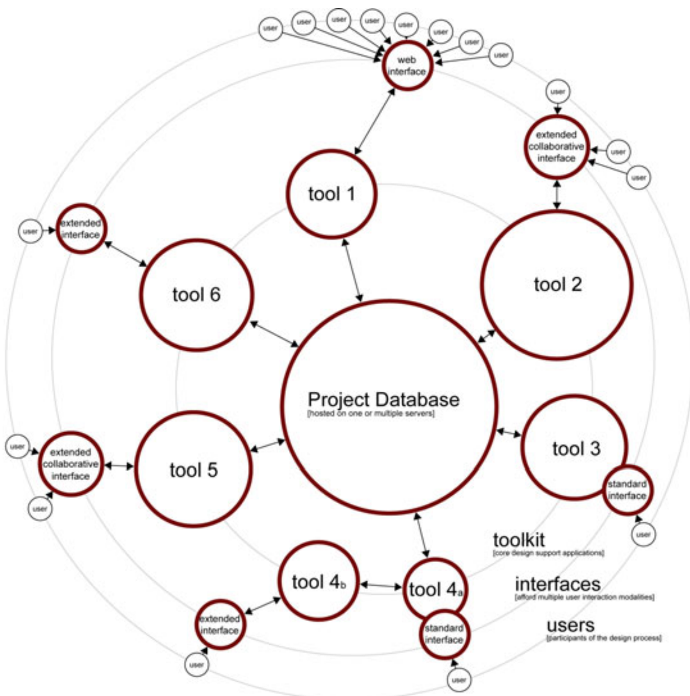

Fig. 1. Complete range of proposed system architecture variations for creating a multiuser design system model. Branches of this system were independently explored in case studies presented further.

Although there are no analogous systems in existence, client applications available to members of the general public willing to participate in the urban design process or obtaining information about developed plans could be compared to features currently available in version 5.0 of the Google Earth [9] service. In a similar way in which 3d building models can be uploaded and visualized in that application, variants of buildings and other future development plans could be displayed in the client application of the proposed system, allowing general public to view, express opinions, debate or vote for most desirable solutions (optionally using real-time controlled avatars). On the other end of the spectrum of possibilities, existing specialized applications such as road network design or traffic simulation software (e.g. OmniTrans International) could be connected to the system through custom APIs (application programming interfaces).

4 Case Studies

On the path of agile development and identification of the specificity of problems that are expected to be faced when creating the proposed system, several research projects and case studies were conducted at the Delft University of Technology, faculty of Architecture, the Hyperbody group, under the umbrella of the Protospace laboratory development, in which the author has actively participated. Several other projects by the author that are also presented in this paper are satellite projects of this development, carried out in connection to educational activities of Hyperbody group.

All explorations were strongly rooted in ideas influenced by computer games. Most important concepts were this of a multiplayer game system and real-time interaction in a 3d environment. The main structure of proposed systems involved a server application (MySQL based, or custom Virtools Dev game server solution) and several diverse client applications or multiple instances of one application. Virtools Dev (currently 3DVIA Virtools), visual programming environment for prototyping 3d computer games, was used as a development platform.

4.1 Paracity Project

The Paracity [10] project continued on the ideas developed in an earlier project: Protospace Demo 1.1 case study. Protospace Demo 1.1 was developed by the author in a team directed by prof. Kas Oosterhuis, involving cooperation of dr. Nimish Biloria and Dieter Vandoren. It was meant as a conceptual prototype study for an architectural design support system. The development was based on the concept that a structural engineer, project manager, architect and material expert would work together on one design model, having each a different interface to it. Thus, the architect would insert and deform functional volumes. The structural engineer would control the structural topology, individual lengths and sections of structural struts. The material expert would choose materials and control sub-surface deformations and the project manager would work with a spreadsheet overview of all materials and involved costs. Due to short project timeframe, the four views were switched in a sequence, yet it would have been possible to provide them simultaneously. The software included an internal project database that was simultaneously accessible by different client applications.

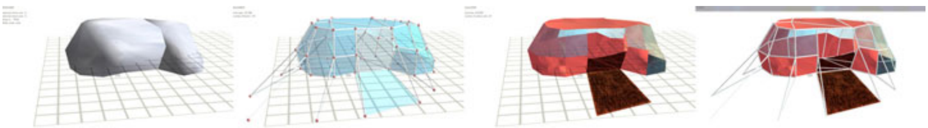

Fig. 2. Screenshots of architect, structural engineer and material expert interfaces of Protospace Demo 1.1 application, followed by the real-time render of a combined model

The later Paracity project was built on a similar principle, yet its aim was to support urban scale design. However, the application did not consist of several interfaces to the same tool, but went beyond that concept by creating a distributed system of multiple autonomous applications, each corresponding to a different aspect of the proposed design process.

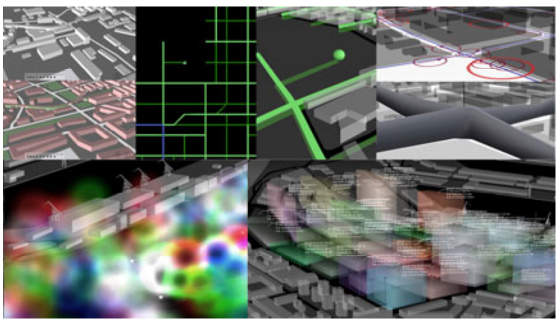

Fig. 3. Screenshot collage of all consecutive client applications of the Paracity project

The prototyped design system was consisted of seven "layers", each layer operating as an autonomous application. First layer was providing a context of the project by defining the state of neighboring areas. The second layer was establishing the topology of all spatial connections within and around the designed area, including roads and pedestrian pathways. The third layer was evaluating that topology based on program distribution and connectivity, using an agent-based simulation. The fourth layer was calculating the potential intensities of various types of user movements through the project area and estimating probabilities of different types of functions to spontaneously appear in an urban environment. Based on those probabilities a program distribution was generated in the fifth layer, whereas the sixth layer allowed for top-down insertion of space organizing points and lines that interactively modified the generated layout in three dimensions. The last, seventh layer, dynamically generated a half-abstract 3d project model. The model is a representation of all parameters constituting the plan, including specifications of functional volumetric masses, urban block envelopes and program distribution shown as color gradients and numeric data. Streets and urban topology are shown schematically as remaining spaces between the blocks.

Unlike in other computer aided approaches to urban design, all operations in the system were performed in real-time, at more than 15 iterations (including rendering) per second. Similarly to the Protospace Demo 1.1 project, the Paracity application did not offer any possibilities of multiuser operation, other than sequential switching between different layers of the process that were meant to be operated by different design process participants, however that functionality could have been potentially added to the system if the project had been developed further.

4.2 Protospace Demo 1.2

In terms of content, Protospace Demo 1.2 project is analogous to the 6th layer of the Paracity project and can be considered to be a detailed elaboration on the activity of program distribution during the urban design/planning process. The project team included same participants as Protospace Demo 1.1, with additional support from dr. Bert Bongers and Maaslab office specializing in interaction design. The main purpose of the project was to test a possibility of multiple user interactions with the design system using one display and multiple controllers. For this, a large projection display was used in combination with two wireless game controllers, a standard OSX speech recognition system, pressure sensor embedded floor mat and an IR beacon tracking system computed using the MAX|MSP platform. The system was designed to be operated by a team of four design process actors. Each of the team members was controlling a different cursor. An algorithm was developed to control camera position and direction based on the location of cursors, causing the virtual camera to move backwards if the cursors were far from the center and forward if the cursors were in the inner part of the screen. Additionally, the view would rotate if an average position of all cursors was close to one of the screen edges. In this way 3d navigation was performed intuitively and collaboratively.

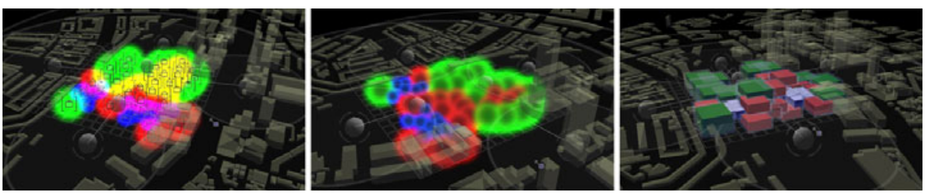

Fig. 4. Screenshots of the running Protospace Demo 1.2 application

Each of the team members used a different aspect of the interface. The urban designer would insert space organizing objects into the system by guiding the cursor using a game controller. The controller was additionally equipped with an IR beacon, allowing for 3d tracking of its position. In this way selected elements could be manipulated not only by controller sticks and buttons, but also by movements of the entire device. Similarly, the other team member, the planner, would manipulate particles representing specific parts of the functional program, drag them into desired areas and allow them to find the most appropriate location in that area on their own, based on their pre-programmed behavior. The cost expert would operate in a radically different way, by walking around the pressure-sensitive floor, which represented the project area. In this way his position in real space was mapped to the virtual environment. Based on that input, location specific information was displayed and cost calculations for chosen areas could be adjusted. The last team member, the session leader, was only using a headset to switch between different phases of the process and save states of the model, while freely moving around, discussing with other team members and supervising the progress of their work.

Fig. 5. Interfaces of the Protospace Demo 1.2 project

The resulting installation verified the possibility of having a small team of specialists working collaboratively on one model from the same location. The concept proved successful. However, four team members was the maximum number that was feasible without significantly decreasing team productivity. Further experiments on interfaces included use of PDA-based input, custom built controllers and augmented displays for role-specific data display.

4.3 751 Project

The 751 project was supervised by prof. Kas Oosterhuis and executed by the author as part of the Hypebrody Master of Science design course. It was an attempt to validate a possibility of a very large team of designers working on one urban design project. The approach covered a different problem than the previously described projects. In this case all design group members had the same role of urban designers, but were assigned to different zones of the design site. The provided site and given assignment were set in a way to force designers to be strongly dependent on each other's decisions. To motivate them to work in an out-of-the box manner, without any design method preconceptions, the site was defined as a three-dimensional volume in place of traditional two-dimensional plot. Design sites were also distributed in three dimensions. This meant that some of the designs had to be located above or under other, often without any direct access to the outside boundary of the design area. In this way projects were mutually dependent, forcing designers to collaboratively solve problems of accessibility, structural support, light access, connectivity, transportation and many other.

To facilitate that functionality, a database system was developed for the use of the project, which, in this case, was not a usual repository of project parameters, but was dedicated to managing only the data being exchanged between projects. Every surface separating any two adjacent zones was mapped to a different table in the database. Each record consisted of 2d position coordinates on that surface, a 3d vector of direction of occurring exchange, its value, units and most importantly, category of exchanged information. In this way it was left open to designers to decide what information was to exchange in the system. Both flows of people and structural forces could be equally well expressed using this data model and it was up to individual designers to decide how this data were to be interpreted in their individual projects. Originally, the communication with the database was performed using a standard online interface. Throughout the duration of the project an interactive application was developed that worked as a viewer of the entire site and allowed for more intuitive selection and definition of information to be exchanged.

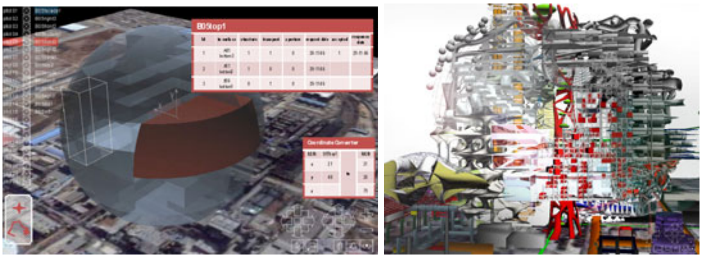

Fig. 6. 751 project, from the left: partially implemented custom database interface application and the assembled model in a real-time viewer application

Workflow progress was twofold. It consisted of working from remote locations and personal meetings in subgroups, rarely involving the entire group, which included 23 designers in total.

The process of development of the project in many ways resembled the growth of real city structures. Since exchanged parameters were forming multiple feedback loops, individual sub-projects were continuously being reconfigured and after five months of the duration of the project, no complete equilibrium was established. However, throughout the process, the project as a whole has evolved into a rich, interesting and potentially well functioning city-scale structure of an unprecedented scale and form. Despite a certain degree of design task abstraction, it was proven that complex designs can be created in an entirely distributed manner, since the used database was storing only locally exchanged parameters.

4.4 A2 Design Studio

The A2 design studio was taught by dr. Nimish Biloria with H.C.Friedrich and the author assisting and developing the design support systems. The system developed for the A2 design studio was in certain aspects similar to the 751 project. Yet, it involved a more realistic design assignment. In this case the individual zones were areas along the A2 highway in the Netherlands, not forming a 3d structure, but a more conventional two dimensional plan. In this case each zone had only two neighboring plots on its two sides and a straight line as a connection. The collaborative design support system was created in a similar manner, yet this time more attention was given to the software prototype, including the functional distribution behavioral modeler which became the core part of the project.

The application was embedded in an on-line website that upon opening, requested the plot number to be edited and in this way allowed for working on all design zones simultaneously, from any location where a standard PC computer and an internet connection were available. Additionally, the system included another client application, which was overlooking the entire project development in a top-down manner, establishing global parameters, accessible from all different zones. These parameters were: overall program values for specific functions to be distributed throughout the whole design site and included guidelines for building heights along the highway represented by a three dimensional curve. In this way each point on the ground plane in the entire site was mapped to a specific preferred height value.

Fig. 7. Screenshots of the A2 application for distributing functional program elements

In order to simulate program demand distribution occurring in real life economy, insertion of each element of the functional program was causing an increase of demand for other functions. This demand was spreading to adjacent zones, while its value would also exponentially decrease. A special matrix model was used to calculate demands for different functions.

5 Conclusion

Presented case studies explore different components of what could together form a fully operating multiplayer urban design and planning system. Most of verified technical possibilities are being currently applied to the Protospace system framework, which in the near future will serve as a platform for implementing proposed solutions in their full potential and validate them by testing the system on applied projects executed in collaboration with the commercial sector.

The outcome of presented case studies has been highly influential on the development of the Protospace System. An additional developed feature of that system, which has not been explicitly demonstrated in presented examples, is integration of client applications of external developers. In shown projects, multiple "views" on the design are created, either by specialized interfaces, or by connecting multiple applications to one project database. For this, presented systems, such as in the Paracity project, consist of different specialized modules (client applications). In reality, these modules need to perform much more complex tasks than in presented examples and it may not be feasible to develop them within one research group or small company. On the other hand, many commonly used commercial applications allow scriptable connections to external servers and/or connectivity to their APIs. In this way those applications could replace selected modules of proposed systems. Several solutions were tested and results are promising.

However, amount of data being exchanged increases as the system approaches real-life applications and with each additional module being introduced. For this H.C. Friedrich has introduced a XiGraph [11] data structure and protocol concept, which has the potential to become an additional layer of the system, mapping and controlling all connected parameters, allowing for multiple databases to be integrated along with a possibility of flexible connectivity between modules, evolving throughout the project. XiGraph protocol supports flexible creation of dynamic connections and dependencies between data sets as well as additionally solves potential conflicts between applications when simultaneously editing same project data.

The development of Protospace system has been put on hold as a result of the disastrous fire of the Faculty of Architecture in Delft, on May 2008. The project has been recently resumed with new hopes for further developments.

References

1. Hopkins, L.D.: Urban Development: The Logic of Making Plans, 1st edn. Island Press (2001)
2. Batty, M.: Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals. The MIT Press, Cambridge (2007)
3. Portugali, J.: Complex Artificial Environments: Simulation, Cognition and VR in the Study and Planning of Cities, 1st edn. Springer, Heidelberg (2006)
4. Wilson, J., Stewart Fotheringham, A.: The Handbook of Geographic Information Science. Wiley-Blackwell, Chichester (2007)
5. Power, D.J.: Brief History of Decision Support Systems (DSSResources.COM, World Wide Web), `http://DSSResources.COM/history/dsshistory.html`, version 4.0 (March 10, 2007)
6. Oosterhuis, K., et al.: Protospace Software. In: Second International Conference World of Construction Project Management, presented at the Second International Conference World of Construction Project Management. TU Delft, Delft (2007). `http://www.wcpm2007.nl/`
7. Kilian, A.: Design Exploration through Bidirectional Modeling of Constraints. MIT Press, Cambridge (2006), `http://www.designexplorer.net/newscreens/phd2006/index.html`
8. Reynolds, C.: Flocks, Herds, and Schools: A Distributed Behavioral Model. Computer Graphics 21(4); SIGGRAPH '87 Conference Proceedings (1987)
9. Google Earth, `http://earth.google.com/`
10. Jaskiewicz, T.: Paracity – A Digital Urban Design Process. In: Game Set And Match II. On Computer Games, Advanced Geometries, and Digital Technologies. Episode Publishers, Delft (2006)
11. Oosterhuis, et al.: Protospace Software