Parametric Design in Protospace 1.1
A Collaborative Design Method

HUBERS Hans
Faculty of Architecture, Delft University of Technology, The Netherlands

Keywords:	3D modelling, collaborative design, design methodology, virtual design studio, parametric design

Abstract:	After a history of CAAD in the Netherlands, the author describes the CAVE-like facility that is being set up to support collaborative design. The first prototype, Protospace 1.1 has been tested. Protospace is a project of the Hyperbody Research Group, directed by Prof. Kas Oosterhuis. The state of the art of parametric design and some helpful tools to speed-up the design process are discussed. 3D multiplayer game software Virtools is used to develop synchronic and a-synchronic, local and inter-local collaboration.

1   A HISTORY OF CAAD IN THE NETHERLANDS

The use of CAD systems in regular architect offices started in the 1980's in the Netherlands. Most systems were not adapted to the Dutch situation. About 200 of the bigger offices therefore united in the VCA, the association of computer using architects founded in 1984, and the author was director from 1989-1995. Different CAD-groups (AutoCAD, CADVANCE, Arkey/Arcos) met at each other's offices to exchange information and to develop libraries of typical Dutch elements like isolated double brick walls and details of connections to e.g. window- and doorframes. In the beginning these CAD systems were 2D and only used for the production of drawings.

When the most important user issues were arranged, the CAD-groups were taken over by the software developers/resellers. VCA started a new overall CAD group called FANFARE, with a project to improve the exchange of data between different CAD systems and -more important- the interchange of CAD operators. It was very hard to find skilled personal for a particular system. They didn't get a budget for it, beside some money from the Ministry of Economy to write an inventory report.

Around 1990 a user group was formed for budget, estimations and specifications, the VCA-BBB group. The CAD systems developed in the meanwhile quantities output, but it was not in the way the cost experts could use it. E.g. they would need the circumference of casing of edges of the floor, but the CAD operators didn't draw a line with that label. And this was only one of many wishes. A big project was

needed with a considerable budget. The only way to get the budget was to participate in the national research project, obliging this rather practical VCA-BBB group to look into ISO/STEP and process- and product diagrams like IDEF and NIAM. Also the project had to broaden to installation engineers who were leading in the field of data-exchange and requested usable input from architects CAD drawings. The project, with an original budget of Euro 1 Million (guilders at that time), was a success. The prototype developed in cooperation with the University of Eindhoven was demonstrated, tested and approved by the VCA-BBB group, but then the software developers, who were in the comment group, decided not to implement the specifications in their software. The reasons they gave were that it was too much focused on the interest of architects and not enough on that of building contractors. But probably the real reason was that they didn't want to invest.

The software developers/resellers, united in the association FORUM, developed a CAD exchange format with the same name. It wasn't a success because some parties refused to accept it. Recently some members of this group started a project called The Digital House, representing 70% of Dutch architectural offices (40% AutoCAD and 30% Arkey). This group is planning to develop de facto standards for the exchange of data between different partners in the building industry: architects, construction advisors, installation advisors, clients, council and building contractors. But it will not be an open standard; members only.

The board of VCA decided in 1995 to integrate with the general foundation of architects BNA. The reason they gave was that offices now paid double contributions, but maybe they also thought that this kind of developments are not in favour of architects. They have to do more work and invest in systems while the other partners in the design can take profit of the improved data exchange. This would explain why since then the developments in this field have been very little.

We can learn from this that, while buildings become more and more complex, data-exchange is not soon without problems. We expect buildings to become more and more interactive, thus complicating the design even more. It is a well-known fact that most influence on the feasibility of a building lies in the beginning of the design. To cope with this growing complexity and to improve design we propose real-time collaborative design in a virtual reality environment. We call the project *Protospace*.

2   PROTOSPACE 1.1

Protospace is a new kind of CAVE supporting collaborative design in real-time (Figure 1). It is a project of the Hyperbody Research Group, directed by Prof. Kas Oosterhuis (http://protospace.bk.tudelft.nl). It is the vehicle for our research, which is aimed at *interactive architecture*, architecture that acts and reacts on changes in the environment. Already many parts of buildings are interactive: elevators, entrance doors, heating ventilation and air conditioning, sprinkler installations, toilets, and not to forget: the doorbell. But we expect more and more parts of a building to

become interactive, thus complicating the design process. Therefore the design of an interactive building needs input from different experts of architecture, cost, construction, environment, installation and more recently, interaction. Collaborative design in the early conceptual phase is needed.
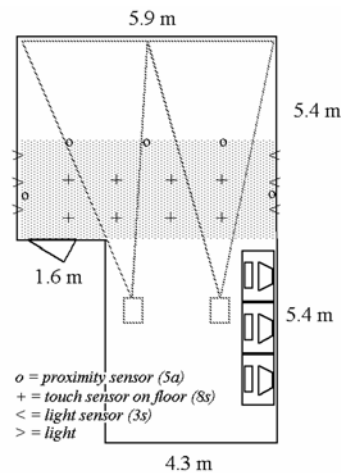
Figure 1 Protospace 1.1

We believe that all stakeholders should participate in the development of the concept. To limit the cost of the design process with all these expensive experts, we propose efficient design methods and Virtual Reality. Parametric design (see next chapter) and handy tools improve the efficiency.

Our VR system will consist of projection of the design on the walls around the team, surround sound, sensors on the "play field" which enable real-time interaction of the stakeholders with the design. We start in Protospace 1.1 with two screens to find out the solutions for creating one virtual world using two render machines and one server and the special VR pack of the 3D game development software Virtools. We have setup the system with 11 switch sensors (s) and 5 analogue sensors (a) (Bongers 2002).

3   PARAMETRIC DESIGN

Parametric design is design in relation to one or more parameters. The final product emerges from variable parameter functions and equations. By defining an object parametrically, it facilitates rapid editing, re-adaptability, and enables useful libraries of scripts and reusable parts for future use.

De Luca and Nardini identified three approaches: the Algorithmic Parametric, the Variational, and the Artificial Intelligence Approach (DeLuca 2002).

Parametric Design in Protospace 1.1

In *Algorithmic Parametric Approach* dimensions and overall geometry are automatically determined by reading from both user input and preset instructions, or rules, which processes this input. An example of this approach is the design of the Waterloo International Railway Station, UK (Moore 1993) by Nicholas Grimshaw & Partners (Figure 2).

Figure 2 Telescopic steel tubes in Waterloo International Railway Station

Given the constraints imposed by the site and the number of trains, the roof varies in span starting from 32.7m at one end, and tapering outwards to 48.5m on the other.

The *Variational Approach* generates a particular object by defining both geometric (e.g. angle, length, etc.) and non-geometric constraints (e.g. volume, mass, etc.). An example is the Web of North Holland pavilion by Oosterhuis.nl (Figure 3).

Figure 3 The Web of North Holland pavilion (source: www.oosterhuis.nl)

Only one parametric detail was developed where the ribs came together, allowing for file-to-factory processing (Oosterhuis 2002).

Lastly, the top-down *Artificial Intelligence Approach* utilizes an "expert-system" which takes an overall master view of the object. The expert system implements a plan based on estimates of local constraints. Subsequently, the global constraints are then checked against the overall plan after the system fulfils all local constraints and completes a design task (Brewer 1987). The advantages of using the A.I. approach become obvious when frequent updates and changes need to be implemented throughout the design process; it essentially reduces the need for the thorough understanding and analysis of complex series of parametric equations and algorithms. Multi-user collaborative design is also facilitated by allowing every member of the design team to modify individual component details without necessarily having to trace back to the origin of the parametric script. An example of A.I. is the ADA project (Figure 4).


Figure 4 ADA project
(source: http://www.ini.unizh.ch/~expo/_download/ada_vp_de.pdf)

ADA is a project by a research group from the Neuroinformatics department of the university of Zurich (Eng 2002) and is probably the only interactive structure that implements AI into architecture. The project is a pavilion for the 2002 EXPO exhibition in Zurich. The structure is conceived as an artificial organism that can interact and communicate with her visitors. It is based on current research in neuroinformatics. By means of her senses, i.e. vision, audition and touch, ADA was able to locate and identify visitors. Her effectors, i.e. lights and sounds, allowed her to provide cues to visitors and express her internal states in an emotional language. Key functionalities of ADA are: to balance visitor density and flow; to identify, track and guide "interesting" visitors; to group selected visitors in space; and to play games with visitors. ADA had to achieve these behavioural goals by establishing active interactions with her visitors.

Parametric Design in Protospace 1.1

There is a fourth approach that exists as a subset (and as such cannot be a single category by itself) of the A.I. Approach: the *Knowledge-Based Approach* (Figure 5). The knowledge is normally stored in rules in the code of the software. These rules often concern the topology of the design, the order in which parts fit together. But they can also relate to any other attribute or relation between objects. Values for these parameters are derived from earlier loops through the system, databases or user input. An example is the ICAD system, in the way it is used by the SIA group at the Faculty of Aerospace of Delft University of Technology in The Netherlands (La Rocca 2002).
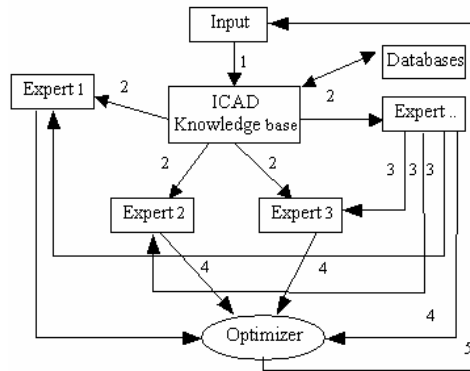
Figure 5 Knowledge based design at SIA

The whole process starts with filling in (1) the input parameters (this is after the engineering rules are programmed). Then the ICAD system sends relevant documents and other output to the software of the Experts(2). The Experts carry out all kind of evaluations and following this may propose other inputs with which to run the system. They might communicate (3) that to the other experts to get an agreement on the next set of inputs. Also this might be automated through an optimizer that collects (4) all evaluations and then starts the process again (5). During our introduction lessons in ICAD we came to the conclusion that it has a negative influence on creativity in architectural design. For redesigning an airplane wing it is very suitable.

Another more recent approach is called A*daptive Parametrical Design*. This is described in (De luca 2002) but more clearly in (Wymer 2002) with next example (Figure 6).
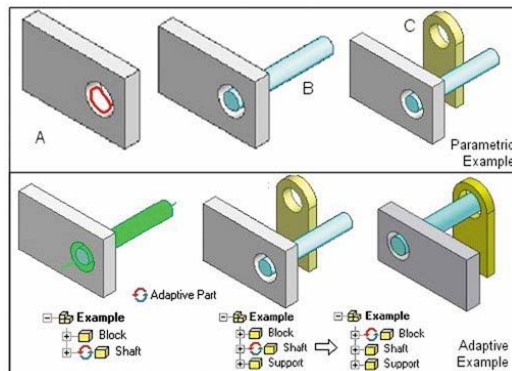

Figure 6 Example of adaptive parametric design
(Source: Wymer 2002)

The upper half of Figure 6 illustrates that with straightforward parametric design, if the hole in part A changes, then shaft B will follow, but not the other way around. With variational design it will work both ways, but if part C comes along then part A and B will not automatically be adapted. With adaptive parametric design, as illustrated in the lower half of the above figure, an intent driving part can become an intent driven part. Unfortunately Wymer does not describe exactly how this works, but it can be inferred that inverting the parent-child relationship is essential.

Our research group gives special attention to *Genetic Algorithms* (GA). It is a type of evolutionary algorithm that is useful for multidimensional optimization problems, a technique that uses computers to simulate evolution through natural selection. GA generates every individual from a peace of code that is known as a "chromosome" or "genome". Chromosomes can be combined or mutated to breed new individuals. In a GA cycle, the fitness of all the individuals in the population is evaluated. The fittest will be used to breed the next generation. This process continues until the operator is satisfied. An example is the BRAIN by Sven Blokker as part of our Virtual Operation Room Project (Oosterhuis 2003).

Another way of parametric design is the S*warm* or flock. Hereby every member of the population reacts on the other members in almost the same manor. We successfully applied this technique in the above-mentioned Virtual Operation Room Project, where Christian Friedrich used it to model a bobbling sphere. Each vertex on the sphere reacts if an avatar is within its proximity, and attempts to maintain a specific distance from neighbouring vertices at all times (Figure 7).

261
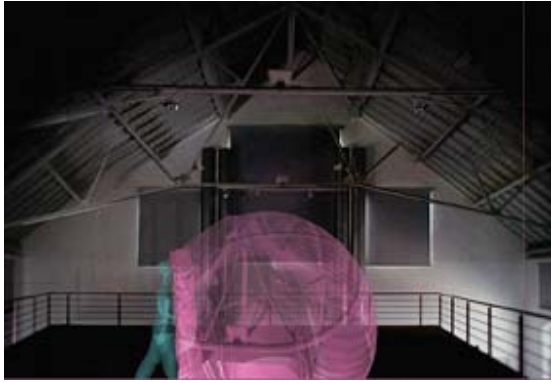
Figure 7 Virtual Operation Room
(Source: Protospace Lab Research, 2003)

For our own experiments we choose the variational approach, with genetic algorithms and swarm behaviour. The directional parametric approach has the drawback of limited ways of input, while the AI approach becomes so complex that for the time being it has a negative influence on the creativity in architectural design.

4   TOOLS

Besides an efficient VR system and design method we also need software tools, in order to speed-up the collaborative design process. The first tool is to import the *Program of Demands* from a spreadsheet and convert it into an array (Figure 8). The script simply iterates through the array and copies a cube, scales it to the volume cell of the actual row of the array, sets the material to the colour cell of the same row and places it next to an earlier copy. Then some functions can help to divide, scale and swarm the volumes over the site, keeping constraints (e.g. maximum height). The volumes serve as a reference layer to design the project. We develop these tools with *Virtools*.

Virtools is a virtual reality development software program of the Paris company with the same name. It won the IST Grand Prize of the European Commission's Information Society Technologies research programme 2002 and can be used for e-marketing, e-learning, gaming etc. We use it for creation and simulation of interactive architecture. We selected it because of the user-friendly interface and the extended functionality. A free player can be downloaded at www.virtools.com.

The interface gives the possibility to have a 3D view of the game where camera's, lights, move around objects, zoom, orbit etc. can be added, and at the same time the developer can work in other windows on the script. A script is the code of the game, but it looks more as a flow diagram.
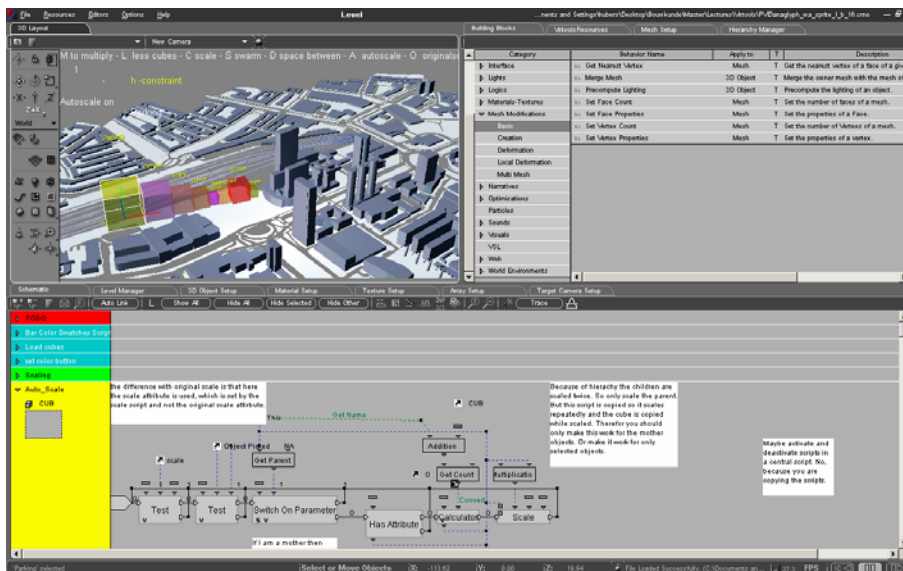
262

Figure 8 POD as volumes loaded in 3D environment, Virtools

The rectangles are called Behaviour Building Blocks. Pick one from a list and drop it in the script, connect the BB's, adjust the input parameters and you have a game. There are different modules: the development software DEV 3.0, a Behaviour and Multiplayer server to develop multiplayer internet games, a VR pack for CAVES and other virtual reality setups. There is also a Software Development Kit, to work in C++ and a scripting language that looks a lot like Java script. Virtools comes with many examples and free to use objects and characters.
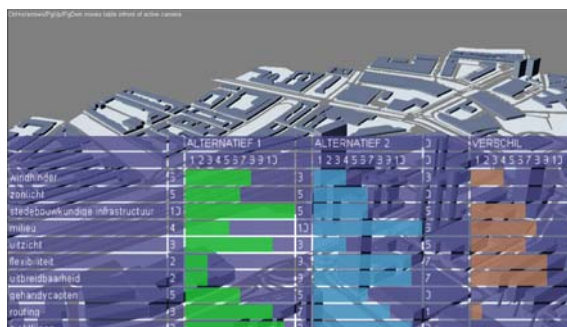

Figure 9 Criteria overlay matrix in Virtools

A second tool we developed is the *Criteria Overlay Matrix* (Figure 9). This helps to speed-up the discussion. Coloured bars show where the evaluation of criteria differs between the design team members.

5   DISCUSSION

Interactive architecture needs collaborative design, because it becomes too complex for one person. The collaborative design team needs a space with virtual reality equipment that we call Protospace. The design process in essence comes down to: developing alternatives and evaluating them (Hamel 1990). Both sub processes can be accelerated, the first with *variational parametric design*, *Genetic Algorithms* and *Swarm* behaviour, the second with a *Criteria Overlay Matrix*. Some handy software tools can further support the design process, e.g. to import a *Program of Demands* as functional volumes.

REFERENCES

Bongers, B. 2002. Interactivating Spaces. In *Proceedings of the Symposium on Systems Research in the Arts: Music, Environmental Design and The Choreography of Space*, ed. Jim Rhodes. Windsor: The International Institute for Advances Studies in Systems Research & Cybernetics.

Brewer, F.D., and D.D. Gajski. 1987. Knowledge-Based Control in Micro-Architecture Design, *In 24th ACM/IEEE Design Automation Conference*, ed. A. O'Neill, and D. Thomas. New York: ACM Press.

De Luca, F., and M. Nardini. 2002. *Behind the Scenes: Avant-Garde Techniques in Contemporary Design*, Basel: Birkhäuser Publishers for Architecture.

Eng, K. 2003. ADA: Buildings as Organisms. In *GameSetAndMatch*, ed. J.C. Hubers: 33-44. Delft: Faculty of Architecture, TU Delft.

Hamel, R. 1990. *Over het denken van de architect.* Amsterdam: AHA Books.

Koza, J.R. 2003. *Survey of Genetic Algorithms and Genetic Programming*. Internet. Available from www.genetic-programming.com; accesed 18 January 2005.

La Rocca, G., L. Krakers, and M. van Tooren. 2002. *Development of an ICAD Generative Model for Blended Wing Body Aircraft Design*. Internet. Available from www.lr.tudelft.nl/sia/publications.htm; accesed 18 January 2005.

Moore, R. 1993. *Structure, Space and Skin: The Work of Nicholas Grimshaw & Partners*, London: Phaidon Press.

Oosterhuis, K., J.C. Hubers, C. Friedrich, S. Blokker, M. Bittermann, and C. Kievid. 2003.Virtual Operation Room. In *Proceedings of E-activities in design and design education*:103-113. Paris: Europia.

Oosterhuis, K. 2002. Vloeiende ribbenkast. *Bouwen met Staal* (August): 26-28.

Wymer, J. 2002. *Simplify Using Adaptive Design*, Autodesk, Manufacturing Division; Novi, USA. Internet. Available from *http://www.widom-assoc.com/html/simplify_by_using_adaptive_des.html*; accessed 1 Feb 2005.