# Modelling as Communication

## *The use of 3D models for developing architectural ideas*

*M. Helena Rua[1], Pedro A. Alvito[2], Duarte Ramos[3], Bruno Fernandes[4], Susana Martins[5]*
*[1]IST - Department of Civil Engineering and Architecture, TU-Lisbon, Portugal,*
*[2,3,4,5]Architecture students, IST - Department of Civil Engineering and Architecture, TU-Lisbon, Portugal*
*[1] http://www.civil.ist.utl.pt/~hrua/*
*[1]hrua@civil.ist.utl.pt, [2]pedro.afonso.alvito@gmail.com, [3]duarte.framos@gmail.com, [4]bmmfe@ist.utl.pt, [5]susanabmartins@ist.utl.pt*

***Abstract.*** *Regardless of its size or scope, the preparation of an Architectural Project, is perfectly well defined. The first phase of conception, drawn up according to the statement of intentions provided by the Owner of the Work (preliminary Program), usually is the only one in the whole process which draws on the abilities of the Author of the Project.*
*Thus, with the aim of speeding up this process, an experimental model which can be used as a tool for negotiation between stakeholders in a project of Architecture has been developed. This implies giving priority to the graphic aspects of the model, such as visualisation, texturing, manipulation and automatic data updating. What is left to be developed is the desire to associate sensorial information to these scenic models – sound, smell and time – and to study the means of converting the results of this informal communication into documents that may come to be used in the subsequent phases of the Architectural Project*
***Keywords.*** *Architecture; modelling, virtual reality; visualization.*

## SUMMARY

The preparation of an Architectural Project is perfectly well defined and managed by legislation[1] involving the use of drawings, usually of Technical quality. In the execution of a project, the sequence of phases corresponds to the relationship between the three main stakeholders – the 'Designer', the 'Owner' and the 'Inspector of the Buildings' of the mayor.

Sketching is the first phase of conception and the only, amongst the whole process, where drawing depends on the abilities of the Author of the Project [FIGURE 1]. However, this informal state of objectives and forms of the Architectural Project always depends from the approval of the client: the more subjectively these are expressed, the more difficult it is to achieve them. It is for this reason that the normal procedure at this juncture is to offer alternatives and variations of the base-concept.

Aiming at speeding up this process, a model has been developed which can be used as a tool for negotiation between the three actors in the process,

[1]Ordinance nº 35/72 of February 11(Instructions for Fees Calculation Relating to Projects and Public Works).

bringing into play their individual interests and contributing towards the establishment of the final form of the project, preferably in real time.

This means giving more concern to the graphic aspects of the model, such as visualisation, texturing, manipulation and automatic updating of proposed alterations and adaptations, to the detriment of the technical information associated with the models so exploited by Building Information Model (BIM). The goal is to have a modelling and visualisation tool which allows the manipulation of geometry and materials in real time, propagating the updates of changes and adaptations automatically throughout the whole project but accompanied by the different attempts to sketch variations of the initial idea.

Figure 1
*Comparison between initial author sketches and the finished construction: the 2005 Serpentine Gallery and the Brasília Cathedral (Siza and Moura, 2010; Niemeyer, 1967)*

## INTRODUCTION AND AIMS

Education in Architecture puts at the forefront communication through drawing in all its formats. However, even more than the latent disappointment which exists when we make mental comparisons between what is imagined and what is later constructed, is the rather unattractive way in which drawing software do represent objects.

Graphic computation (in vector format), published as an alternative to the process of traditional design[2] , makes tools for drawing three-dimensional models immediately available. The problem lies with the representation of solids as a material. The evolution of the algorithms used in these programs allowed overcoming these initial limitations, and even to adding texture to those surfaces. There are numerous three-dimensional modelling tools specifically targeted for architecture, but, essentially, they develop the technical design component aimed at construction projects (Stouffs, Beirão and Duarte, 2009). This is the case of BIM, a particular type of CAD systems, which was the answer for the need to link written references to the drawings or models. This made possible to automatically obtain reports on the specificities of the building's operation.

In spite of the importance of these developments for the building industry, representation in Architecture falls short of the Designers' requirements. New means of visualizing their projects are necessary, that is: alternatives to pre-rendering or to video, a fact that is especially critical when the project calls for particularly high graphic quality. To some extent, it can be said that the technology available for visualisation of Architectural models, such as outlined below, is still an early stage development, especially when compared with the technology for real time rendering, already available in other more developed areas such as in videogames.

In an initial phase, the problem of overcoming these limitations was solved by personalising games' graphical engines for the project purposes. However, this procedure is not totally satisfactory because as it is not possible to control the importing of the scenarios, nor even to access the programming language editor.

At present, and relative to the aims of this paper, new forms of realistic representation in Architecture are being sought. Research is focusing on specific tools which meet the needs of communication and which can provide an alternative to the renderings and videos used at present, without any detriment to the graphic quality of the models [FIGURE 2].

[2]Related to normative architecture, (Márton, 2010).

The work presented here is a research into the modelling tools most appropriate for Architectural Projects, in terms of visualisation of the final result, and, where relevant, as an alternative even to the traditional planning process. This process is based on procedures which are the most natural, and the most compatible with the method of construction itself. In other words, the objective is to create realistic virtual models which perform well in meeting the demands of an Architectural Project, also allowing users to manipulate the scenarios created by the exploratory programs in real time and in parallel. A further aim is to find a modelling process that can evaluate the construction and planning sequence to be used in Architecture.

Finally, future developments can be expected from the pursuit of these two broad objectives, namely the evolution of technical aspects related to the visualisation and presentation of the architectural work. These can be sensorial sophistication, that is, adding details of sound, smell, volume and impact. Nevertheless, evolution can also be expected in technical aspects related to planning, such as remote team cooperation, where each individual technician is responsible for a section of the project, which s/he develops simultaneously and together with the others involved.

In order to do this research, a model of an existing building, well-known to the study group, was developed as a case study. Furthermore, group of volunteers carried out evaluation of its impact with the purpose of checking the results.

## Process Modelling

### Context

Mainly there are three kinds of computer graphic software available when one needs virtual models:

1. CAD software modelling, oriented for design performance of two and three-dimensional objects, with emphasis on rigorous constructive design;
2. Model and visualization software, which allows one to customize objects with textures, for instance; and
3. Virtual reality software or real-time simulation programs, such as game engines that allow the manipulation of objects.


*Figure 2*
*Example of what can be achieved by manipulation and image processing – Frank Lloyd Wright Falling Water project (Kasper, 2006)*

Compatibility between these software is required for quality reasons. It is also a know issue because data conversion often implies the redefinition of the models. Alternatively, it is possible to use interface tools (known as plug-ins), but with the drawback of these are programs that become out dated very easily.

**Modelling:** the strategy varies according to each type of CAD, and it requires the designer to think differently in the case of a 2D or 3D model, or if other kinds of information should be associated (such as BIM – Metadata), or if a parametric or generative model is needed. Parametric models allow dynamic exploration and fast shaping of a variety of solutions, The final form is reached more quickly when compared to traditional methods, since there is no need to redo the entire model whenever there is a slight change in the project. In turn, associative models also match constraints but with autonomous components, i.e., as corresponding to the combination of different parametric models in the execution of a whole object.

The association of technical descriptors (metadata) to graphical components permit automatically work reports and a faster search for a specific element in a database which is constantly updated. However, the objective of this application restricts the ability of reaching a free-form object. And the modelling procedures are dependent on the used software (Sketch Up, ®AutoCad, Revit, Rhinoceros, Bentley) and on the set strategy.

**Edition:** most models support different visualization and analysis techniques. In the case of performative design there is a goal, which is to improve one or more aspects in order to maximize the design's efficiency. This purpose can be related to sustainability in architecture (where for example ®Ecotect Analysis could be used), to structure improvement, etc. These are models that, in order to be handled with some operability, are usually constructed in specific environments that have some limitations therefore sometimes simplification of the architectural form are required and not always visual previews of the results are available in automatic mode. The final product is obtained by experimentation, a posteriori, when a satisfactory image is achieved.

Basically, at this stage it is necessary to handle two types of software: one that allows texturing and superficial treatments that are overlaid to the model, such as GIMP (GNU Image Manipulation Program), ®Adobe Photoshop, ®Illustrator (FreeHand e Corel Draw) Flash, Paint, ®InDesign; and others to assign the desired image to the model, such as ®3D StudioMax or Blender.

**Production:** manufacturing – virtual or real – is the expected outcome of any modelling project, and more important than evaluating the results exclusively from the aesthetic point of view, is the fact that it allows to confirm the articulation consistency of the project, to test the quality of the structure and to implement automated series of production (CAD/CAM). Software for animation is usualy used to this end (real-time rendering), or in alternative editing games cards such as Cinema 4D, Maya, ® ActionScript 2.0 (basic), ® Premiere. The main drawbacks of this procedure are related with the necessity of reducing the quality of the models in order to be able to display its manipulation, and also the need of study the attribution of post-prodution (physics, sound, etc.) to their performance – ®AfterEffects, ®SoundBooth, ®SoundForge. Finally, adjust the model for publication through the use of ®Dreamweaver, ®HTML (HyperText Markup Language), ®XML (Extensible Markup Language), ®CSS (Cascading Style Sheets), among others.

The main features and limitations of Computer Graphic (CG) software used for architectural purpose can be summarized in the following [Table 1]:

This table suggests that any software is a support tool for Architecture but where there isn't a concern for an active communication with the stakeholders of a project. Thus, we developed a model process with this quality, namely:

1. Where the geometry is related to the different phases of a project, enabling interactive evaluation of different projective hypotheses, since its early stages;

*Table 1*
*Results of the software, based on tests made by Martins (2010)*

| Characteristics | | | |
|---|---|---|---|
| Revit | Lack of Boolean operations (only subtraction is possible) complicate the modeling task. | It is hard to recognize which restrictions have already been applied. | Lack of abstraction. Method is based on technical drawings, depending on orthogonal projections (views). |
| Grasshopper | The arches were a geometry challe nge as there was no curve to answer the needs. | Data issues: dealing with lists of lists of elements is not obvious with the given tools. This struggle caused a change in strategy more than once. | The constant visualization and the programming concepts involved in the procedural modelling helped understanding written programming concepts. |
| Blender | Exclusive for visualization, real time visualization and rapid prototyping. | The existing 2D drawing documentation is not suitable for technical drawings. No 3D data can be added. Therefore it is not suitable for architecture. | Conversion for another formats is possible with eventual loss of quality. |
| VisualScheme | More generalization and higher abstraction lead to higher reuse of the written functions. | Less interactive than grasshopper, less empirical. It is harder for the user to evaluate each individual function. | Independency from the software as the user is capable of designing his own tools. |

2. Where also the architectural finishes, either from the object and its environmental could be seen and tested in an interactive way; and,
3. Where the model could be easily 'used' in real time.

Basically, it is the application of film animation concepts to Architecture.

**Methods**

The selection of informatics was focused on the choice of software that best fits to the purposes of an Architectural Project, since there was no intention of using more sophisticated equipment than a simple laptop to express ideas.

The software choice was the freeware open source Blender, from Blender Foundation[3], also by economic reasons, but mainly because it was the best of all for modelling/visualization/manipulation without having to convert files. In fact, Blender software is one application that integrates the actual game engine (with real-time display) with by other tools of modelling, animation, texturing, and sometimes also allows programming intelligent events.

In turn, the case study for modelling was the Pavilhão Central (main hall building of Instituto Superior Técnico in campus Alameda) by the architect Pardal Monteiro, conducted from 1927 to 1937 [FIGURE 3]. Mainly this choice is because it is a singular building, well documented and easily accessible, as it is on our faculty. Being a building with a permanent occupation of 70 years and with a large number of modifications, we set up as our goal to draw up the initial project and, according to the results, study their adaptation to its more representative epochs.

*Figure 3*
*Instituto Superior Técnico, Lisbon*

[3]Blender it's an open source software for 3D modelling (animation, rendering, post-production, interactive criation and copy/paste) available for most operating systems in general public license format (GNU).

The graphic data of the project was divided into three groups, independent but referenced and developed separately by different designers. Each of them also corresponding to different levels of detail, namely:

1. Pavilhão Central building (nº 4 in the picture);
2. Topography and surrounding;
3. Exterior and interior architectural details.

The building model was based on the creation of faces (limited by edges – fill); on the subdivision (cut) of those faces until the desired geometry is achieved (assuming, in Blender, at most 4 vertices per face) and also with the use of the common tools and operators, such as move, rotate, and extrude, to position the geometry in its place [FIGURE 4]. Thus, the final model of Pavilhão Central results from the union of simple entities, to achieve the interior and exterior complexity of the original building, benefiting from its symmetry.

The same procedure was made to model the surrounding scenario and the architectural details, each one at its own scale. In order to fasten the whole process and to easily interact with it, the three different models of the building, of the surrounding campus and of the architectural details, were developed in separate files and then referenced to each other.

Then we proceeded the overlay of texture images (UV Unwrap) in harmony with the type of material involved: diffuse (diffuse texture), shine (specular color), reflective (reflection or ambient map) or detail (normal maps).

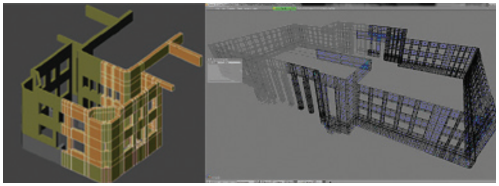
The diffuse texture controls the basic features of the object, such as direct colour while the specular controls the gradient of brightness and colour; the ambient maps provide some additional reflections (currently, there are just a few tools with the ability to calculate reflections in real time). The normal maps allow adding some detail to the object without making the system slower, and without the need of adding new geometry. They work by controlling the normal (perpendicular) to the faces, namely tag along the variation of one face's texture, giving the illusion of finer details.

Usually, for this purpose, first an equivalent object is modelled separately (prototype) with a great level of detail in order to obtain pictures of geometric textures in the desired position (Bake), and later more detail results from applying the image of this more complex geometry (high-poly) on the more simplified model (low-poly).

Realistic lighting should simulate in computer graphics what happens with real light – ambient-light (indirect) and spot-light (direct). The creation of shadows makes the scenes more realistic, but also allows evaluating the effects of textures. The complexity of indirect lighting is simulated using a combination of spot-lights and environment-lights (Hemi). Spot-lights spread shadows over other objects shadowmaps format and environment-lights produce a comprehensive and uniform light, brightening the scene and softening the shadow/light contrasts. In turn, the direct light is obtained with individual spot-lights located at strategic points, specifically targeted some of the scene objects, managing to clear and control its own lighting [FIGURE 5].

## DISCUSSION AND RESULTS

The option for a known and accessible building provides a comparison between model and reality. Beyond the obvious lacks of equipment and human use of space – because, at this stage, the interactivity



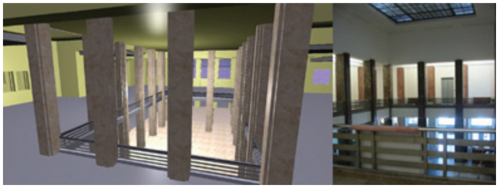


*Figure 4 (left)*
*Example of different phases of modelling Pavilhão Central*

*Figure 5 (right)*
*Pavilhão Central Atrium – model and reality*

and intelligence components available in Blender not yet been activated. At this point, there are obvious differences that may have negative connotations (such as the variation of architectural finishes) but, on the other hand, there are very favourable aspects, including the freedom of choice to 'walk' the space and points of view.

Blender provides some advantages over other software when used as a real-time architecture visualization tool, however this advantages are at the same time its greatest strength and also its greatest weakness.

The internal render engine that powers Blender's viewport visualization is a powerfull Game Engine, based on OpenGL with GLSL shaders. This allows an experience more visually appealing when compared with other more generic real time renderers that power common CAD tools, like AutoCAD, Revit, Rhino, etc. GLSL stands for Open GL Shading Language and is a C based open programing language that allows the customization of the virtual materials applied to the scene objects. These allow greater visual fidelity, and contribute to the overall "photo-realism".

The built-in dedicated Game Engine also means that real-time interaction tools, navigation methods and interactivity can be infinitely better than any viewport based software. The open-source nature of the application also allows the creation of standalone executables, that can be easily distributed and visualized by anyone, without installing any applications, and eliminating the need to have a dedicated viewer.

This means that even with little additional work a preview simulation of the model can look moderately good, some generic textures can be applied, and adding some lighting and shadows can add depth to the model. However, to bring it to a production level quality, some extensive texturing and backing work has to be done, which can be a lengthy task. This necessary work defeats the whole purpose of having a quick visualization tool, even if necessary for quality visualization, because of all the overhead and time consumed preparing the model.

This is can be aggravated if the model is exported from another application, since editing and reimporting the geometry might imply loosing work and backtracking.

Another limitation of a Blender model (or other mesh modeling software for that matter) is that this "dumb" models carry no data with them, as opposed to models created in a BIM environment. Polygon models are in a way "useless" for construction or fabrication, since not only they don't have construction related information, they are mesh-based, meaning that there is no curve or geometry precision, everything is triangulated or at best converted to quads, rendering them somewhat inaccurate for fabrication processes. Even bi-dimensional CAD drawings can be hard to obtain from polygon or mesh geometries.

Also, even though polygon based tools have many useful automating or "parametric" tools, they are mostly directed for animations, character modelling and rigging, not for architecture, engineering or construction. These tools may be used to reduce some work and create so called "parametric" or adaptive components; or at least dynamic models that can be re-shaped and reused, some times even with more success than dedicated tools; but they are largely unadapted for rigorous CAD work and fabrication purposes.

Software like 3DSMax, Revit, AutoCAD or Rhino have the advantage of being able to construct accurate NURBS based models, with construction information (in the case of Revit or Archicad or other Building Information Model) and at least allow accurate bi-dimensional drawings like elevations or sections. Most of these software already allow dynamic elements which may function as modules or components adaptable to different situation, this automating the drawing task.

However, most lack the graphical computation performance required from a dedicated visualization tool. Real time render engines are usually chosen to optimize viewport editing and functional clarity, rather than visual fidelity or (photo) realism, therefore the visual quality of the model is far from what it could be.

Also, since this is not the primary function of these render engines, they lack a method for distribution, meaning it is generally impossible to distribute a lightweight version of the model/viewer that allows a client to navigate a 3D model without a license and installation of complex and expensive editing software.

If the result of a talk between stakeholders leads to the adjustment of the preliminary program provided by the owner, for instance, the increase of the living room's area or a change in a bedroom configuration then this software allows some modifications, including the displacement of walls and/or equipment.

This is an excellent tool to help decisions about the architectural finishes as well the decorative elements. Since it is neither a parametric tool nor does it allow association of related information (such as details from the others specialties of an architectural project), then changes are not easily upgradeable.

One possible application of these visual models can be rapid prototyping, especially when the object, either by its scale or its characteristics, does not need to be detailed in the constructive system point of view. Depending on the used hardware, with greater or lesser detail, the models can result in another form of communication and become a political tool.

Finally, it is possible to think about implementing this type of procedures to modelling/visualization of small projects, such as advertising and marketing of Real State properties for instance, and analyse the 'user's' reaction.

## CONCLUSION

Any program that bases the presentation of its results on visual display has quick didactic applications as it is a form of communicating and representing ideas. Consequently, there are countless fields where experimentation with these types of programs took place: art, cinema, design, construction, education and research amongst others. There is nothing better than an image to explain a concept, when it corresponds with a design or a model. The aim of this work was to explore its usability in the field of Architectural Planning, not only where the visualisation of interior space is concerned but also for the interface with the user (keyboard, mouse and so on) and, later to be used in the exploration of the programming of logic and animation in the model, that is, Artificial Intelligence.

## REFERENCES


Castle, H (ed.) 2009, *Closing the Gap: Information Models in Contemporary Design Practice,* Architectural Design, Vol 79, No 2. Wiley, London.

Castle, H (ed.) 2009, *Digital Cities*, Architectural Design, Vol 79, No 4. Wiley, London.

Galanter, P 2003 'What is Generative Art? Complexity Theory as a Context for Art Theory' *Soddu C (ed.) Proceedings of 6th International Conference on Generative Art,* Milan, pp. 216–236.

Kolarevic, B (ed.) 2003, *Architecture in the Digital Age* – Design and Manufacturing, Spon Press, New York, London.

Martins, S 2010 *Digital Project Design methodology experimentation, Master's thesis*, Instituto Superior Técnico.

Sakamoto, T, Kubo, M, Ferre, A 2008, *From Control to Design – Parametric/Algorithmic Architecture.* Actar Edition.

Schumacher, P 2009 'Parametricism - A New Global Style for Architecture and Urban Design' in Castle, H (ed.) *Digital Cities – Architectural Design,* Vol 79, No 4, London, pp. 14-23 (DOI: 10.1002/ad.912).

Stouffs, R, Beirão, JN, Duarte, JP 2009 'Grammars of designs and grammars for designing – grammar-based patterns for urban design' in T Tidafi and T Dorta (ed.) *Proceedings of Joining Languages, Cultures and Visions: CAADFutures*, Canada.

Van der Zee, A and de Vries, B 2008 'Design by computation', in: C Soddu (ed.) *Proceedings of 11th International Conference on Generative Art*, Milan, pp. 35-52

Zellner, P (ed.) 1999, *Hybrid Space: New Forms in Digital Architecture,* Rizzoli International Publications, Inc., New York.
Weblinks:
[1] AKT (experimental research team at Adams Kara Taylor - http://www.akt-uk.com/), available at: http://scriptedbypurpose.wordpress.com/participants/akt/ (September, 2010)
[2] Kasperg: 2006, Frank Lloyd Wright Falling Water in Half-Life 2, http://www.cstrike-planet.com/maps/969-fallingwater?p=3, consulted: 25/05/2011)
[3] Márton, E.: 2010, 'The Evolution of Design Techniques' – http://corpora.hu/en/pressmaterials/catalogue/eniko-marton-the-evolution-of-design-techniques/# (consulted: 27/01/ 2011)
[4] Stellmach, T.: 2007, "Associative design at berlage." http://www.dysturb.net/ 2007/associative-design-berlage/ (consulted on September 2010).