

VISUALISATION OF BUILDING MODELS

Karl Grau¹, Kim B. Wittchen¹, and Christian Grau Sørensen²

¹Danish Building and Urban Research, DK-2970 Hørsholm, Denmark

²High School Student, DK-4684 Holmegaard, Denmark

ABSTRACT

In an existing integrated building design tool, *BSim2002*, a 3D building model can be created. The model can be used for thermal simulation, but also for other purposes, e.g. for calculation and visualisation of the solar penetration into a building. The model can be exported to other programs, e.g. for daylight calculations and visualisation. Finally the model can be visualised using the OpenGL technology, which is an integrated part of the Microsoft Windows operating system, allowing the user to have a virtual tour around and inside the building.

This paper describes how this approach has been implemented and shows some examples of the advantages, especially when dealing with complex models, compared to the normal wire frame views used in this type of programs.

INTRODUCTION

The most common way to visualise building models in thermal simulation programs is wire frame drawings as plans, elevations and 3D. It is a more or less tedious job to create a 3D-building model from paper or CAD drawings. When the 3D building model has been created it is thus obvious to reuse the same model for other purposes than thermal simulation.

A model can be used to visualise the solar penetration into the building, so the user can see how a solar spot from a window or opening move over the floors, internal surfaces and even passes through doorways and internal windows.

Another use of a model can be to export it to other visualisation tools or calculation programs. Many different file formats and 3rd part commercial, free or share ware programs exists. This makes it possible to create more realistic images of building models created for thermal simulation purposes.

Using build in graphics capabilities in Windows, e.g. OpenGL makes it also possible to create small and fast programs for visualisation.

BUILDING SIMULATION 2002

BSim2002 (Wittchen, Johnsen and Grau, 2000-2003) is a user-friendly, flexible computer program for analyses of indoor climate, energy consumption and daylight performance of buildings. *BSim2002* contains different tools and links to applications making it possible to carry out a complete thermal, moisture and daylight analysis of a building.

The core of the system is a common building data model shared by the different design tools, and a common database with typical building materials, constructions, windows and doors. Figure 1 illustrates the user interface of *BSim2002*.

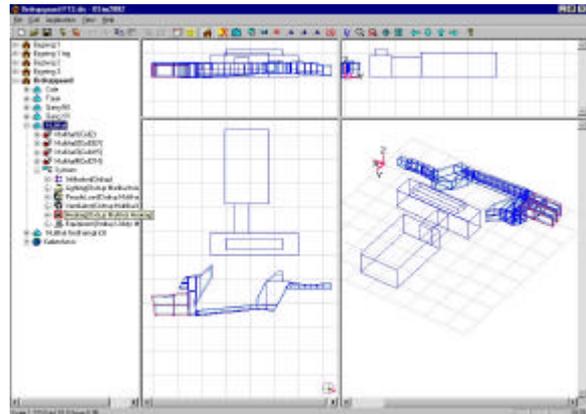


Figure 1. *BSim2002* model of an extension to Ordrupgaard art museum north of Copenhagen.

Applications integrated in *BSim2002* are:

- *SimView* for editing and visualisation of the building model geometry.
- *XSun* for analyses of solar distribution and shadows in and around buildings.
- *tsbi5* for thermal simulations. The program is based on the finite control volume method. It has been widely validated and employed in several international research projects, e.g. EU COMBINE (Augenbroe, 1995). Recently the program has been extended with a model for synchronous calculation of transient hygrothermal conditions (Rode and Grau, 2001).

- *SimLight* is a simple tool for estimation of the daylight conditions in convex spaces.
- *SimPv* is a tool for calculation of the potential electrical production from building integrated PV-systems (Wittchen, 2003).
- *SimDB* is the user interface for the common database with building materials, constructions etc.

From within *SimView* it is possible to link to a few external programs:

- *SimDxf* allowing the user to create the geometry of the building model by using a CAD-drawing (DXF-format) as the basis (Grau and Wittchen, 1999).
- *Bv98* can import the building model from *SimView* to carry out a heating demand calculation according to the EN832 standard and a building code compliance check.
- The building model can be exported to the *Radiance* program for visualisation and detailed calculation of the lighting and daylighting conditions.

The following calculations can be made on most buildings using *Bsim2002*:

- Heat gains from solar radiation, people, lighting, and equipment
- Solar radiation through windows
- Heating, cooling and ventilation
- Floor (inside constructions) heating
- Power and energy balances
- Temperature, moisture and CO₂ conditions
- Temperatures and condensation risks in and on constructions
- Perceived indoor air quality
- Heat and air exchange between zones
- Shading conditions
- Variable infiltration and venting
- Several different ventilation systems simultaneously
- Air exchange in connection with infiltration and opening of windows
- Natural ventilation
- Heat and refrigeration recovery in ventilation plants
- Supply and exhaust air temperature in ventilation plants
- Power consumption in heating and cooling coils and humidification in ventilation plants

MODEL EDITOR

Models can be created using the build-in model editor *SimView*, but can also be created on the basis of a CAD plan drawing. The geometry is defined in a local 3D coordinate system, with the X-axis facing east, the Y-axis north, and the Z-axis facing upward. Normally a building is not oriented according to this local coordinate system, but the model can finally be rotated according to the actual orientation. A model consists of one or more buildings, but only one building can be analysed at a time.

The geometry of the buildings is represented as shown in the NIAM diagram (Nijssen and Halpin, 1989) in Figure 2.

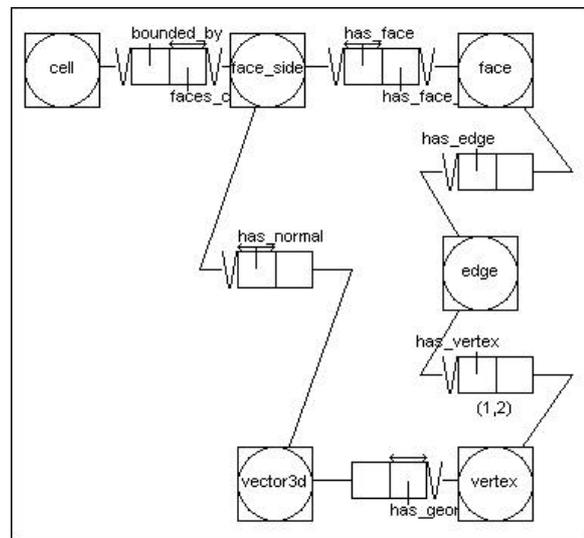


Figure 2. Geometry and topology model for BSim.

Cells bounded by face sides represent rooms. Each face has always two face sides, one facing the cell representing the room, and the other facing the neighbour cell or the outside. Faces are represented as polygons of edges, and two vertices holding the 3D local co-ordinates define the edges.

The user interface (see Figure 1) reflect this structure in a hierarchical tree shown in the left part of the view, while a graphical view is shown in the right part. The graphical view is split up in four different views:

- X-Y plane view at lower left part,
- X-Z elevation view at upper left part.
- Y-Z elevation view at upper right part, and
- 3D wire frame view (parallel projection) at lower right part.

These wire frame representations are a common way to visualise building models in thermal simulation programs. For small models it is good way, but for big and complex models it is difficult to get an over-

view (see Figure 3) and to locate a specific face, edge or vertex.

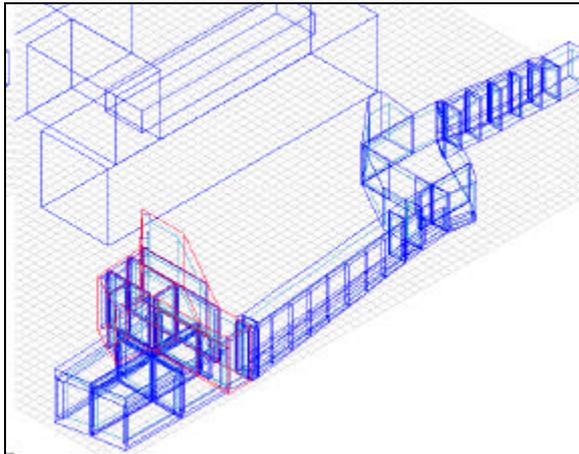


Figure 3. Wire frame view of a complex building model.

SOLAR PENETRATION

Apart from making a thermal simulation the model can be used for analyses of solar penetration into the building using *XSun*, which takes into account shadows cast by other buildings and the building itself.

The method used to perform the calculation of shadow- or sun patches is developed in the context of hidden surface removal within computer graphics, and is based on clipping a concave polygon to the boundary of another concave polygon performed in 2D (Weiler and Altherton, 1977).

Such an algorithm to perform polygon clipping has been developed (Grau and Johnsen, 1995). Polygons are represented as circular lists of vertices. The orders of the lists are important, and can be determined by the sign of the area A of the polygon:

$$A = \sum_{k=1}^n (x_k y_{k-1} - y_k x_{k-1})$$

Where n is the number of vertices, and x_k and y_k are the 2D co-ordinates of vertex k . If the area is positive, the list is ordered clockwise, otherwise the list is ordered counterclockwise.

Two sorts of clipping can be performed:

- Calculation of the polygons inside one subject polygon and outside another clip polygon, referred to as *exterior clipping* (see Figure 4). In this case the subject polygon is ordered clockwise and the clip polygon is ordered counterclockwise.

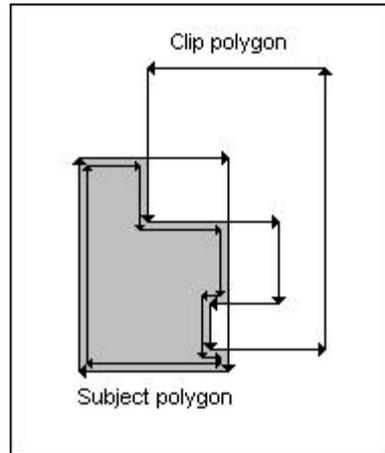


Figure 4. Exterior polygon clipping.

- Calculation of the polygons inside one subject polygon and inside another clip polygon, referred to as *interior clipping* (see Figure 5). In this case both polygons are ordered counterclockwise.

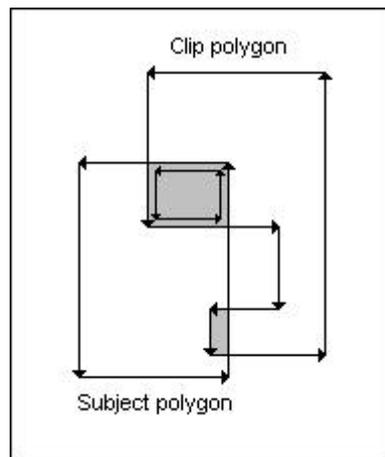


Figure 5. Interior polygon clipping.

The boundaries of the subject polygon and the clip polygon may or may not intersect. If they intersect the intersections will occur in pairs. One intersection occurs when one clip polygon edge enters the inside of the subject polygon and one when it leaves.

When an entering intersection occurs by the clip polygon, the boundary of the clip polygon is followed until it leaves. From that point the subject polygon is followed until another entering intersection occurs by the clip polygon. This process can be repeated until the first entering intersection is reached again.

Special cases has to be considered, e.g. when the clip polygon is completely inside or outside the subject polygon, or when the clip polygon has common edges with the subject polygon.

This algorithm for polygon clipping is the core of *XSun*.

Using *XSun* to calculate the sun patches entering a building at a given time, polygon clipping is being used in the following way:

- The outside faces of surrounding buildings that can cast shadows on the faces of the building are projected in the direction of the sunray onto the actual face as shadow polygons. The same is done for faces of the building that can cast shadows on the building itself, e.g. L-shaped buildings.
- All external windows and openings that can be hit by the sun are clipped to polygons outside the shadow polygon, giving polygons receiving sun.
- These sun polygons are projected in the direction of the sunray onto the inner face of the wall, taking into account the wall thickness. The internal sun polygons are clipped to be inside the opening boundary.
- The inside sun polygons are projected in the direction of the sunray onto the internal faces of the room. These polygons are clipped to be inside the boundary of internal windows and openings. In the case that sun polygons has been found, the previous task is repeated recursively until no more sun polygons are found.

Figure 6 shows how the sun enters a building, and how the sun is passed on to a neighbouring room through an opening. Figure 7 shows an example of shadows cast from a neighbouring building. To analyse which buildings obstruct the sun, the model can be viewed from the sun in the direction of the sunray (see Figure 8).

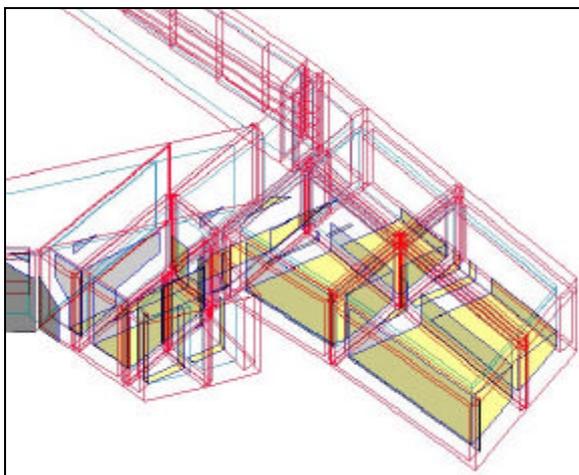


Figure 6. Incident sun calculated by *XSun*, August 23, 2003 at 17:00 in Copenhagen.

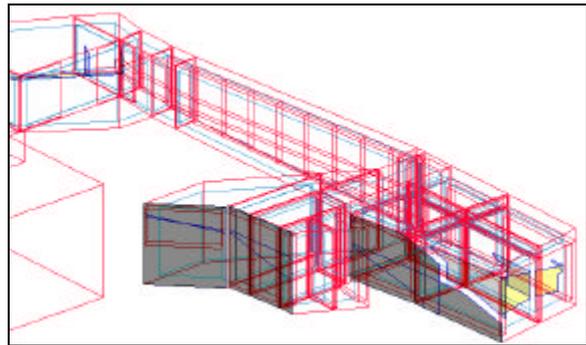


Figure 7. Shadows cast from a neighbour building, August 23, 2003 at 18:30 in Copenhagen.

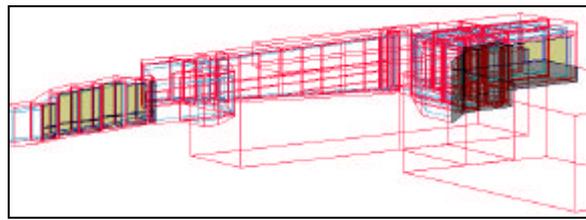


Figure 8. Shadows cast from a neighbour building as seen in the direction of the sunray, August 23, 2003 at 18:30 in Copenhagen.

RAYTRACING

The views shown until now has been wire frame transparent views, which are not very realistic and can be difficult to overview. In order to have photo realistic views and to perform lighting calculations the model can be exported to *Radiance* (Ward Larson and Shakespeare, 1998).

Radiance employs a light-backwards ray-tracing method, extended from original algorithm introduced to computer graphics (Whitted, 1980). Light is followed along geometric rays from the viewpoint into the scene and back to the light source. The result is equivalent to follow the light forward, but the process is more efficient because most of the light leaving a source will never reach the point of interest.

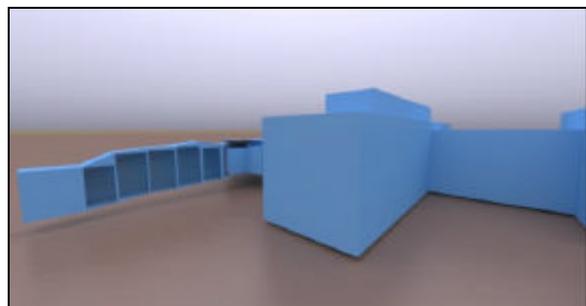


Figure 9. Radiance picture of the extended art museum with an outline of the existing building in front.

Figure 9 and 10 show pictures made by *Radiance* of the extension of Ordrupgaard art museum seen from different viewpoints.

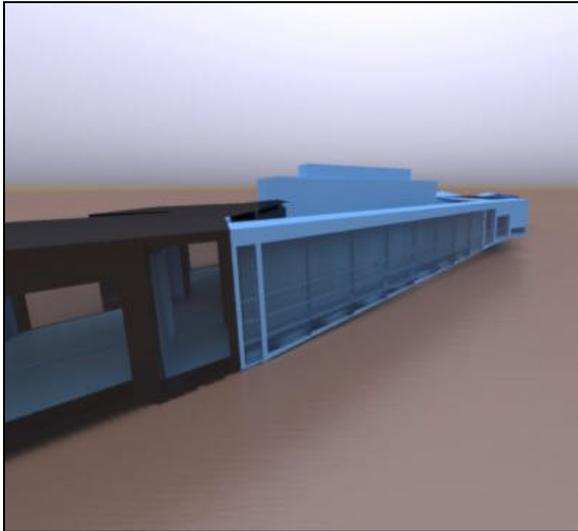


Figure 10. *Radiance* picture of the extension to the art museum and the existing building in the background.

DIRECTX

Making photo realistic pictures with *Radiance* are very time consuming. Depending on the complexity of the model it can take several hours, or even days.

In order to make visualisations of models fast, but of less quality, they can be exported to the DirectX file format.

The DirectX file format is defined by Microsoft, and is supported by many programs for visualisation and animation. The DirectX file format provides a rich template-driven format that enables the storage of meshes, textures, animations and user-definable objects. In *BSim2002* the complete geometry and the material properties can be exported to a file that can be opened from 3rd part visualisation programs supporting DirectX.

Figure 11 and 12 shows samples of pictures produced by the program *Deep Exploration*, a professional visualisation tool from the company Right Hemisphere.

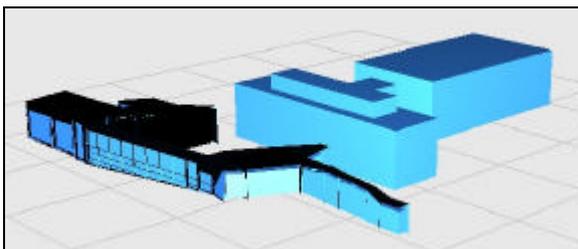


Figure 11. The museum showed as a solid view exported as a DirectX file.

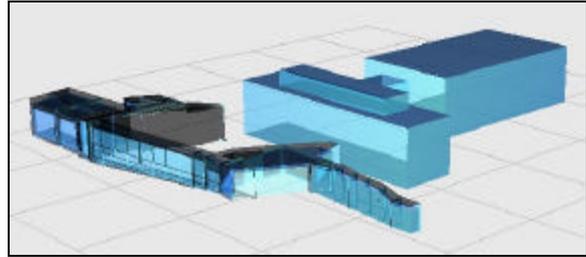


Figure 12. The museum showed as a transparent view exported as a DirectX file.

OPENGL

OpenGL is a three-dimensional graphics software interface, which is an integrated part of the Windows operating system. Originally OpenGL was developed by Silicon Graphics, Inc., but has become an open industry-standard independent on operating system and hardware, and the standard is maintained by an independent group called the Architectural Review Board. The OpenGL software interface allows graphics programmers to create high-quality, three-dimensional colour images with graphic effects such as shading, lighting, texture-mapping etc.

A small program *BSim3d* has been developed in Visual Basic based on OpenGL to make visualisations of building models. The models can be viewed from outside as well as from inside the building, allowing the user to have a virtual tour in the building.

The building models are imported from XSun via a file in DirectX format to get the geometry and material properties, but also the sun patches at a given time as calculated by XSun.

Figure 13 - 15 shows sample pictures of the new extension to the art museum seen from outside as well as from the inside of the building.

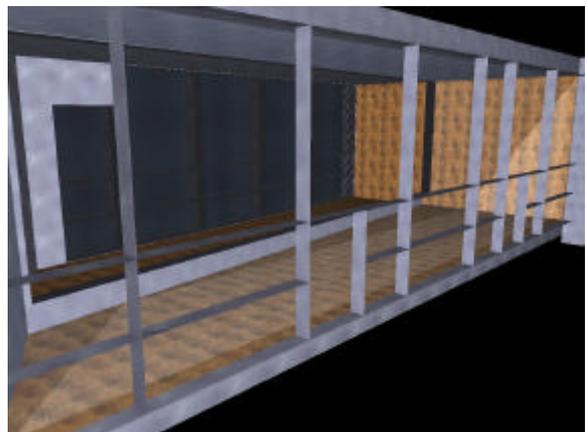


Figure 13. *BSim3d* outside view of the new extension to the art museum.

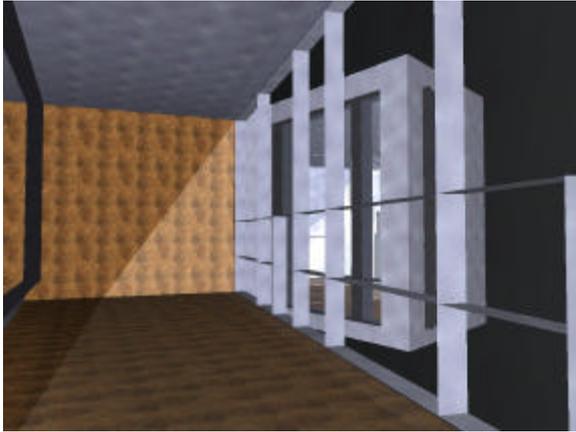


Figure 14. BSim3d inside view of a corridor in the new extension to the art museum.

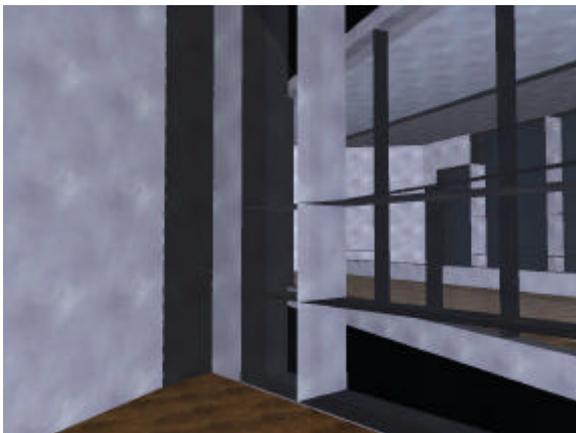


Figure 15. BSim3d view from inside into a corridor in the new extension to the art museum.

CONCLUSIONS

A thermal simulation program can create a complete detailed 3D-building model. The model can be used to predict the indoor climate and energy consumption. In addition the model can be used to calculate the solar penetration and distribution within the building, to be used for thermal simulation and for visualisation.

To obtain more realistic visualisations of a building, the model can be exported to other tools: to *Radiance* for photo-realistic visualisation and detailed lighting calculation and to other visualisation tools based on the DirectX file format. Finally the model can be exported for having a virtual tour in and around the building using a simple tool based on the OpenGL graphics interface integrated in Windows.

Using more photo-realistic methods to visualise building models for thermal and simulations makes it easier to explain and show the model for a non-technical audience. It also saves the time needed to create new models in other applications for solely visualisation purposes.

ACKNOWLEDGEMENT

The consulting engineering firm Birch & Krogboe A/S has supplied the building model of the extension to Ordrupgaard art museum, and it is gratefully acknowledged.

REFERENCES

Augenbroe, G.L.M. (Editor). 1995. COMBINE 2, Computer Models for the Building Industry in Europe, Final Report, Delft University of Technology.

Grau, K., Johnsen, K. 1995. General Shading Model for Solar Building Design, ASHRAE Transactions, Vol 101, Part 2, San Diego, USA.

Grau, K., Wittchen, K.B. 1999. Building Design System and CAD Integration, Proceedings of IBPSA Building Simulation '99, Kyoto, Japan.

Nijssen, G.M., Halpin, T.A. 1989. Conceptual Schema and Relational Database Design, Prentice Hall.

Rode, C., Grau, K. 2001. Synchronous Calculation of Transient Hygrothermal Conditions of Indoor Spaces and Building Envelopes, Proceedings of IBPSA Building Simulation 2001, Rio de Janeiro, Brazil.

Ward Larson, G., Shakespeare, R. 1998. Rendering with Radiance, Morgan Kaufmann Publishers, San Francisco, USA.

Weiler, K., Altherton, P. 1977. Hidden Surface Removal Using Polygon Area Sorting, Computer Graphics, Vol. 11, USA.

Whitted, T. 1980. An Improved Illumination Model for Shaded Display, Communications of the ACM, 23(6).

Wittchen, K.B., Johnsen, K., Grau, K. 2003. BSim - User's Guide, Danish Building and Urban Research, Hørsholm, Denmark.

Wittchen, K.B. 2003. Building Integrated Photo Voltaic in a Thermal Building Simulation Tool, Proceedings of IBPSA Building Simulation 2003, Eindhoven, Netherlands.

INTERNET SOURCES

Danish Building and Urban Research: www.dbur.dk

BSim2002: www.bsim.dk