

ATMOSPHERES: PROOF OF CONCEPT FOR WEB-BASED 3D ENERGY MODELING FOR DESIGNERS WITH WebGL/HTML5 AND MODERN EVENT-DRIVEN, ASYNCHRONOUS SERVER SYSTEMS.

Timur Dogan¹, Christoph Reinhart¹

¹Massachusetts Institute of Technology, Cambridge MA, USA

ABSTRACT

This paper describes a prototype of a web-based 3D design and energy modeling platform called “Atmospheres”. A WebGL based 3D geometry modeling and data visualization interface that runs in any modern web-browser is coupled with an event-driven, asynchronous server that runs transient, 3D energy simulations. Besides an easy to use geometry modeler and an intuitive spatial visualization of the simulation results, the web based character of the platform allows easy and instant access to state-of-the-art 3D energy simulations and thus can potentially broaden the energy modeling user group.

INTRODUCTION

The field of building performance simulation [BPS] has reached high levels of modeling capability. The available tools are able to inform and predict key architectural qualities such as the thermal and visual comfort of occupants or the energy use intensity of a construction.

Common practice is that these tools are usually used in the middle of the design and planning process and only for larger projects due to cost reasons.

At this point the architects have already made key design decisions such as building proportion, allocation of spaces, facade design and the spatial interrelationship with the context. However, these decisions can make or break the energetic performance of the building. For smaller projects like small residential buildings the budget usually does not allow to perform a detailed energy simulation. To lure architects into BPS to inform their designs from the very earliest moment many researchers have worked on simplifying and speeding up simulations to make them more accessible for designers.

In 2006 Urban and Glicksman introduced the MIT Design Advisor, a non-geometric web based tool that performs a simplified server-side energy simulation. The tool allows the user to quickly identify tradeoffs between design solutions. While tools like DesignAdvisor have successfully lowered the inhibition threshold to use BPS due to simplicity and easy accessibility, energy simulation tools are still far from being accepted as supportive and generative architectural design tools [Morbiter et al. 2001,

Andre et al. 1994, Bauer et al. 1998, De Wilde et al. 1998, Robinson 1995].

The reason for the lack of acceptance of energy simulation tools in the early design phases is often believed to be due to their complexity and slowness and because they do not provide morphological feedback. However, the authors believe that these factors represent only a fraction of the problem. 3D visualization and animation software are highly complex and rendering images can be very time consuming - yet computer renderings have become common practice in all stages of the design process. The generated images transport a great amount of design relevant information and therefore are of great interest to architects. Energy simulation results also carry design relevant information, however, often in a hidden numerical form that is not directly accessible to a designer. Thus, it becomes hard to draw architectural and morphological conclusions from the results. Especially, simplified non-geometric tools like Design Advisor fall short in this category.

With respect to the afore mentioned the authors would like to present innovations in the following areas:

- Web-based 3D interface for EnergyPlus.
- Intuitive spatial visualization of the BPS results to strengthen the morphological feedback.

METHODOLOGY

Underlying Web Technology

HTML5 [W3.org, 2013] and WebGL [Khronos Group, 2013] are the underlying technology for the 3D interface on the client side. HTML5 is the markup language for presenting content in the World Wide Web. The fifth revision supports the latest multimedia technology and has improved consistency throughout different devices and browsers. WebGL is a young technology that allows high performance, interactive, 3D graphics within a web browser by utilizing the clients graphics processing unit [GPU] to accelerate image processing as part of the web site canvas. WebGL elements can be mixed with other HTML elements. This has been used extensively for the interface elements and the 3D viewport of

Atmospheres. Atmospheres WebGL program consists of a main program loop written in JavaScript and shader-code that is executed by the GPU [Khronos Group, 2013].

The main program loop is responsible for all user interactions such as polygon drawing and geometric transformations. It also handles all viewing and projection matrix calculations. It then fills the WebGL buffers with the geometric information. The shader takes over the lighting calculation, applies colors to the geometry and renders it. The main loop also has to report the user interactions to the server. Whenever there is geometry added or modified the main loop utilizes Socket.IO to broadcast the events to the server. This is also used for all other client interactions such as log-on or chat messages. Socket.IO is a JavaScript library to facilitate the making of real-time apps for browsers and mobile devices by blurring the differences between the various transport mechanisms available for web communication [Socket.IO, 2013]. It primarily uses the WebSocket protocol. As a fallback solution it can also use Adobe Flash sockets, JSONP polling, and AJAX long polling, while continuing to provide the same interface.

The server side runs Node.js with a Socket.IO extension and listens for client side events. "Node.js is a platform built on Chrome's JavaScript runtime for easily building fast, scalable network applications. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient, perfect for data-intensive real-time applications that run across distributed devices." [Node.JS, 2013] Whenever an event is received, the server processes the data and manages it. The data can be geometric data, energy model parameters, user

data or messages. The data is then stored in a database called mongoDB. MongoDB is an open source document-oriented database system that belongs to the NoSQL system family [mongoDB, 2013]. Instead of storing data in tables as done by "traditional" database systems, MongoDB stores structured data as JavaScript Object Notation [JSON] documents in order to make the data integration easier and faster for web applications. The JSON format is a text-based open standard designed for human-readable data interchange. The server also performs the energy simulations.

Atmospheres uses EnergyPlus as underlying simulation engine. "EnergyPlus is an energy analysis and thermal load simulation program" [DOE, 2013]. The authors chose EnergyPlus because it is available for free. However, any other simulation software that reads in a text based input file can be used.

Whenever a user requests a simulation - the client side aggregates an EnergyPlus input file [IDF] and posts it to the server. The server then executes the simulation and broadcasts back the numerical results that are then further processed on the client side. The modus operandi is depicted schematically in Figure1. The IDF file can also be downloaded and saved locally.

The 3D interface, geometric modeling and energy model setup

One goal was to make modeling of the 3D geometry of thermal zones as fast and easy as possible. The tool follows a draw-poly-line and extrude methodology. For non-convex polygons a simple ear-clipping algorithm has been implemented. [ElGindy, 1993]

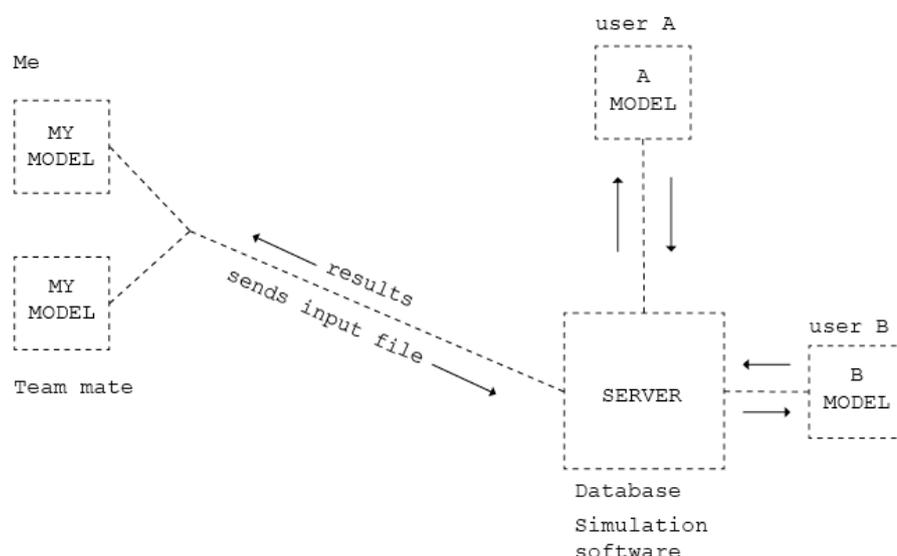


Figure 1 Schematic modus operandi for the "Atmospheres" prototype

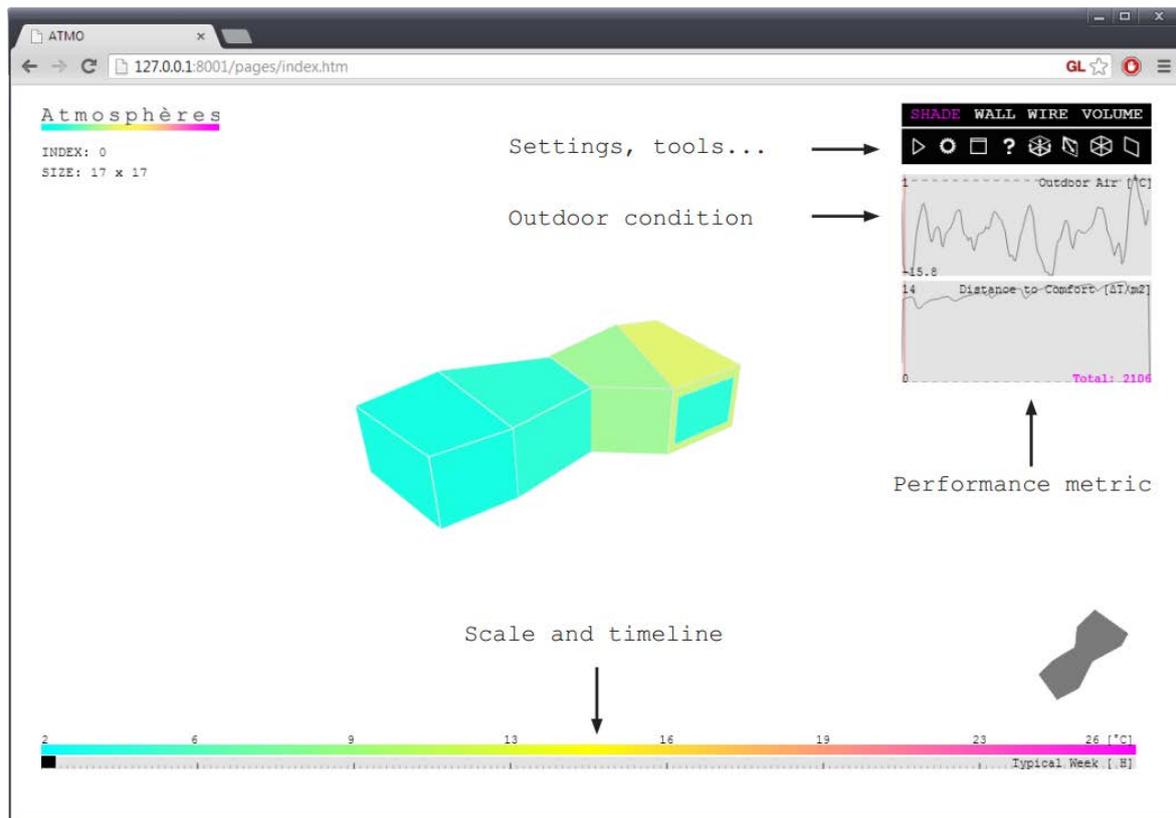


Figure 2 Screenshot of the interface with annotations.

For the drawing as well as for the extrusion process various popular construction aides such as object or axis snaps have been implemented. Once the zone is drawn one can modify the base polygon by simply dragging the corresponding points. Pushing and pulling of the vertical or horizontal faces is also possible. Windows/openings are modeled through a separate command that allows the user to draw a polygon on any face of a zone. Three-dimensional allocation of the mouse cursor, object identification for picking and finally rendering the geometry are done by utilizing WebGL buffers (index buffer / depth buffer / frame buffer).

With more than one zone present in the scene the software has to establish a connectivity graph for inter-zone heat/mass flow automatically. A very basic surface intersection test has been implemented to detect coincident faces (connections). Once the geometry is modeled the user still has to specify certain inputs for the thermal simulation. Through an object selection command the user can set loads and schedules for a zone as well as physical properties of slabs, walls and windows. With the request of a simulation the client-side software decomposes the 3D model into EnergyPlus specific elements. Each modeled volume is mapped to an EnergyPlus zone. The zone specific data such as loads and schedules that have been attached to the volume are also translated to their IDF-syntax equivalent. Then all

faces of the volume are traversed and mapped to the appropriate EnergyPlus Element according to their normal orientation. The face orientation and previously generated connectivity graph is then used to apply the correct boundary condition to the face. User defined materials are managed in a dictionary and are written to the IDF file after all zones and faces have been traversed.

Since the tool is tailored for early architectural design it hides the complexity of a heating, ventilation, and air conditioning [HVAC] system and tries to shift the focus to spatial implications of energy in buildings. However, it would be easy to add more complexity to the simulations such as HVAC or airflow network functionality at a later point, since it would just require an expansion of the input masks for the thermal zone object. Figure 2 shows a screenshot of the interface with some text annotations. The toolbars, sliders, legend as well as the graphs are HTML canvas elements that float on top of the WebGL viewport. The image shows a 3D model with four EnergyPlus zones and simulation results loaded. A video that shows Atmosphères in use can be found under the following link:

<http://vimeo.com/58514267>

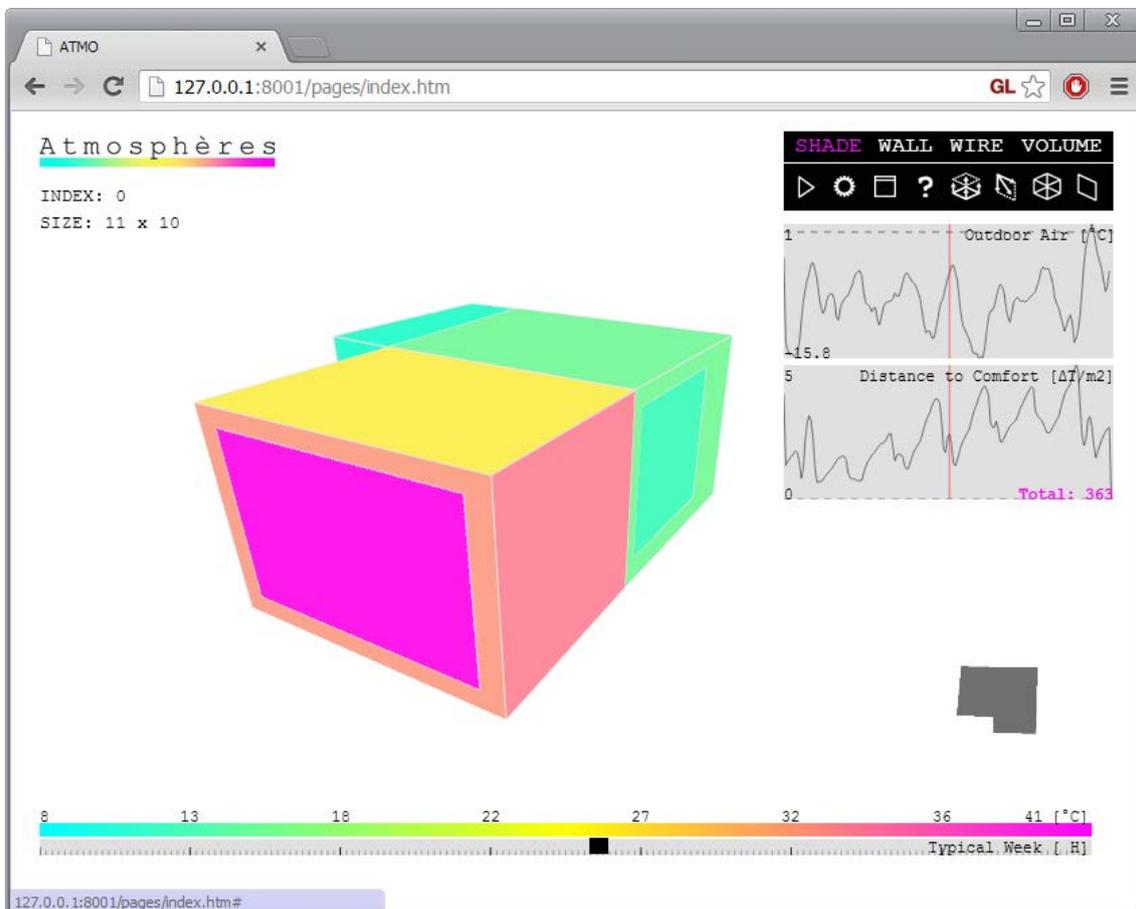
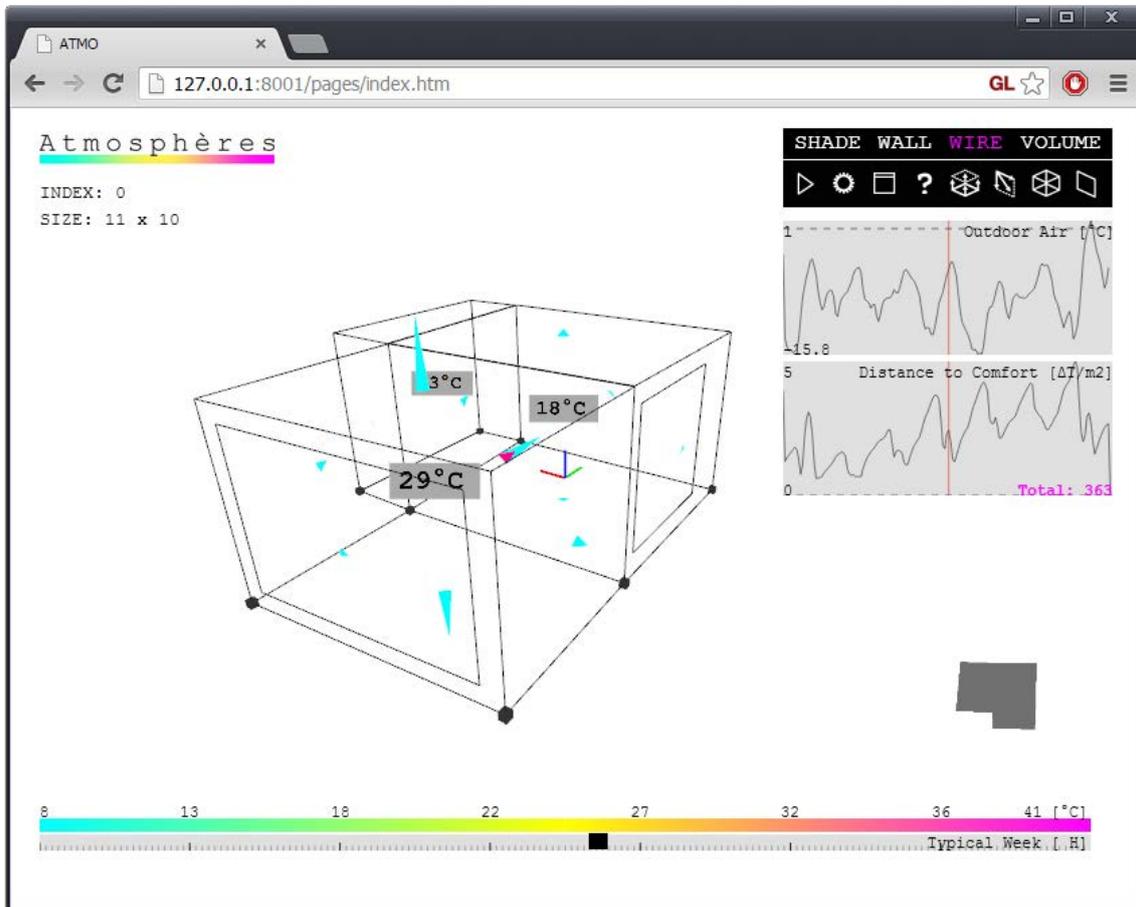


Figure 3 and 4. Wireframe visualization with heat flow arrows and colored faces by temperature

Spatial Data Visualisation

The data table output of EnergyPlus [DOE,2012] is tied more closely to the morphology by visualizing the data spatially. Like first introduced by the EnergyPlus interface OpenStudio [Guglielmetti, 2011] faces of the 3D model can be colored according to their inner face temperature [Figure 4]. In addition to the above heat and mass transfer are displayed in a 3D flow diagram that allows the user to understand the flow direction and order of magnitude of the flows. The exact numerical values can be visualized as arrow labels.

As a side effect this visualization style reveals the nodal resistor network character of the underlying simulation software. Thus, an inexperienced user can easily understand the capabilities and limits of the software. Figure 3 shows the flow diagram visualization.

Since the above mentioned phenomena are varying over time the authors also included a slider element in the interface lets the user intuitively explore the temporal resolution of the simulation results.

To expand the spatial resolution of BPS the authors implemented a volumetric rendering of the mean radiative temperature [MRT]. For each point in the cloud a view factor based temperature is calculated. In order to allow interactive scrolling through the temporal resolution of the EnergyPlus results the time-consuming view-factor calculations had to be accelerated. Exploiting the client-side GPU processing power does this. Since the face temperatures are already mapped on the walls and slabs the MRT can be approximated by simply counting the colored pixels each point sees. In a separate frame buffer a low resolution 360 degree panorama is rendered (cube map) [Greene, 1986]. Then the image is converted into a WebGL texture. Through the WebGL Mip-Mapping capability (fast texture reduction) [Williams, 1983] the average color/temperature can be approximated.

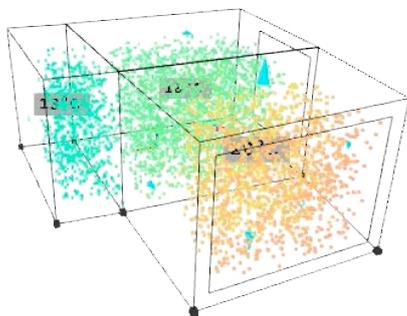


Figure 5: Mean radiative temperature cloud. The visualization enhances graphs and data-tables with spatial information allowing the user to discover “local” details of their designs.

DISCUSSION

Web versus Offline

Within a feasible time-frame the authors were able to proof the successful implementation of the aforementioned modeling, simulation and visualization functionality in a WebGL/HTML5 context and created the prototype software “Atmospheres”. The potentials for web deployed software is great. It can substantially improve the accessibility. Without the hassles of installation or downloading regular software the tool grants easy and instant access to BPS. It also facilitates the maintenance of the software for users and programmers. Other advantages of the web-based software like collaborative design, crowdsourcing and collective memory can easily be added at a later point. The authors believe this approach can substantially increase the value of a tool.

The authors are aware that performing a transient energy simulation on the server-side can create significant computational loads for the server under heavy use. This problem would have to be encountered with more server capacity as the service grows. However, the authors believe that this is not a problem with respect to the recent growth of on-demand infrastructure offers [Azure, 2013].

Neglecting HVAC systems

The use of heating and cooling systems has not been implemented. Although, HVAC systems are a crucial part of energy simulations and can help to reduce overall energy consumption, the authors wanted a more intuitive metric that also the inexperienced user can put into context. The delta between the operative temperature to a user defined comfort range was chosen. Further, the choice to exclude conditioning systems reveals the true behavior of a building design. A space that heats up to 50°C or more in the summer is intuitively regarded as an absurd design proposal whereas a very high cooling demand expressed in Kilowatt-hours might not directly lead to the same conclusion. Especially for the layman it is not easy to contextualize these energy figures.

Spatial resolution of BPS

The volumetric rendering of the MRT was an attempt to extrapolate the results of a typical EnergyPlus simulation in order to reveal local details of a design. The approach is capable to point out unexpected hot or cold spots especially close to windows and was able to show areas of higher radiative temperatures next to a trombe-wall. From an architectural perspective these local characteristics are valuable since they can give hints for the appropriate use of the adjacent space. Despite the interesting observations the authors believe that they have reached a limit of current BPS software such as EnergyPlus. To provide more spatial resolution one would need a different simulation engine.

CONCLUSION

With Atmospheres the authors were able to create a prototype of an intuitive 3D building energy modeling and design tool that runs in a web browser and is thus accessible to everyone with an internet connection. Atmospheres runs serverside EnergyPlus simulations visualizes the simulation results in the 3D browser viewport. It is the authors hope that the technology introduced in this paper can broaden the energy modeling user group.

FUTURE

The authors are on the web deployment and public beta release of the software.

REFERENCES

- 10gen, 2013, mongoDB, mongoDBNoSQL database, software version v2.2, URL: <http://www.mongodb.org/>, last accessed 2013
- Azure, 2013, Microsoft Windows Azure, URL: <http://www.windowsazure.com/en-us/>, last accessed 2013
- Chronic Logic, 2000, Bridge Builder, Bridge Builder Game, software version unknown, URL: <http://www.chroniclogic.com/>, last accessed 2013
- DOE, 2013, EnergyPlus, Energy analysis and thermal load simulation program, software version 7.2, URL: http://apps1.eere.energy.gov/buildings/energyplus/energyplus_about.cfm, last accessed 2013
- ElGindy, H., Everett, H., and Toussaint, G. T.. 1993 "Slicing an ear using prune-and-search," Pattern Recognition Letters, Volume 14, Issue 9, Pages 719-722, ISSN 0167-8655
- Guglielmetti, R., Macumber, D., Long, N. 2011. OpenStudio: An OpenSource Integrated Analysis Platform. Proc. 12thConference of International Building Performance Simulation Association, Sydney, AU.
- Greene, N., 1986."Environment Mapping and Other Applications of World Projections" ,Computer Graphics and Applications, IEEE , vol.6, no.11, pp.21-29.
- Khronos Group, 2013, WebGL, Web Graphics Library, software version 2.0, URL: <http://www.khronos.org/webgl/>, last accessed 2013
- Node.JS, 2013, Node.JS, Node.JS platform for fast, scalable network applications, software version v0.8.20, URL: <http://nodejs.org/>, last accessed 2013
- Urban, B., Glicksman, L. 2006. The MIT Design Advisor-a fast, simple tool for energy efficient building design.In IBPSA-USA Conference Proceedings.
- Socket.IO, 2013, Socket.IO, Socket.IO JavaScript library for realtime web applications, software version v0.9, URL: <http://socket.io/#home>, last accessed 2013
- W3.org, 2013, HTML5, Hyper text markup language, software version v5, URL: <http://www.w3.org/html/wg/>, last accessed 2013
- Williams, Lance. 1983. Pyramidal parametrics. In *Proceedings of the 10th annual conference on Computer graphics and interactive techniques (SIGGRAPH '83)*, New York, NY, USA