SPOT

A 3D environment for direct sunlight visualization

DEA - MODELISATION ET SIMULATION DES ESPACES BATIS

Student : Sébastien BUND

Readers : E. Y. DO; J.C. BIGNON; P. BACH

CRAI

MAP - UMR CNRS n° 694

DMG-University of Washington (USA)

Centre de Rercherche en Architecture et ingenierie (France)

The Design Machine Group (DMG)

The Design Machine Group (DMG) is a "*collaborative research studio aimed at fostering and developing ideas that will shape the future of design and information technology*"
The DMG's research is based on the premise that new media is revolutionizing the practice of architecture and urban design. The DMG is analyzing and imagining how designers could better use new technologies in the design process. DMG's research projects can be sorted into five main categories: Back of an Envelope Design; Collaboration, Coordination and Constraints; Virtual Places and Smart Environments; Urban Design, and Physical Computing.

The projects in the Back of an Envelope Design category explore new methods for 3D model generation. Several projects also investigate how new technologies can help designers achieve a deeper understanding of the envelope they are shaping.
Several projects of the DMG are aimed at improving the communication between all the people involved in a project. These research projects belong to the Collaboration, Coordination and Constraints category. The Virtual Places and Smart Environments projects explore how 3D geometries are generated by defining the rules they should follow. For example, a composition system or structural constraints could be a base for shape generation or analysis.
The projects of the Urban Design category research how to improve the design of projects at the urban scale.
Physical Computing research projects address how the objects around us can be transformed into input devices which could also react to the environment. By embedding electronic into every day life objects, they could become smarter.

DMG was therefore one of the best places in the world to lead my research in the design computing field.

ACKNOWLEDGEMENTS

As a French student, doing my internship in an American research laboratory, the Design Machine Group- at the University of Washington was a lifetime experience for me. It would never have been possible for me to study abroad and go through the administrative steps without full support from both the DMG and the CRAI, where I completed my master's thesis on the "Modeling and Simulation of Built Spaces" (DEA "Modélisation et simulation des espaces bâtis").

Thomas Jung made this internship possible for me by helping me prepare and organize it. Thomas also gave me full access to his "Space Pen" project's code and assistance in implementing SPOT on top of it.

I would like to thank Ellen Do, Mark Gross and Brian Johnson who supported my ideas and hosted me while I was at the DMG. They provided me with the greatest work environment and gave me good counseling throughout my research. Ellen considered me as one of the DMG's students. She not only gave me clever and helpful advice to improve my research skills (coding, library research and documentation of SPOT), but she also gave me full financial support to extend my internship.

I also would like to thank Karen Hanson for her support in writing this thesis in English, which is my second language.

The Region Lorraine financially supported me with a grant. I never could have done this internship without this aid.

SPOT,

Interactive navigable 3D visualization of direct sunlight

SUMMARY

1- ABSTRACT

Daylighting is the most pleasant, economical and ecological source of energy for buildings [1]. Le Corbusier strongly advocated that, “Each dwelling must have a set amount of hours for daylight to enter” [2]. A simulation and visualization tool can help designers understand and recognize lighting problems and opportunities that should be considered during the early conceptual design stage. We present SPOT, a software tool that enables architects to quickly preview the amount of direct sunlight amount projected on specific areas over time in a navigable 3D web environment (Java3D).

Recent research enables quick generation of 3D models. VR SketchPad [3] interprets freehand sketches into 3D geometry. Harold [4] and SpacePen [5] support direct drawing in 3D spaces. However, none of these tools provide rapid direct sunlighting simulation in the 3D environment. Even though ray tracing computer graphic methods, such as “Radiance” [6], can simulate photorealistic lighting effects of a space at a given moment in a viewpoint, they do not provide real time feedback of lighting effects over time. Using them as a design analysis tool requires a great deal of trials to be performed on different days of the year, time of the day and from several viewpoints. This process is time consuming because it involves constant adjustments of the sun’s position and image rendering. Therefore, we built SPOT to enable architects to perform quick visualization and assessment of direct sunlight effects during the early stages of design.

Direct sunlight visualization in navigable 3D space is a multi-dimensional problem. The sun angle variations with the diurnal and annual cycles need to be considered because the sun’s position fluctuates with time. Consequently, SPOT needs to visualize a five-dimensional phenomenon. Several research

projects address the complexities of lighting visualization. For example, to display multiple dimensional data on a single 2D display, “Space Series” [7] uses a focus-plus-context technique. The illumination levels in a “workplane” are generated for different times or dates in a 2D display where the X and Y axis respectively represent the annual and diurnal cycle of the sun path. However, this system can only display a single 2D space, and the target audience for the project is lighting experts who have specialized knowledge. SPOT, on the other hand, focuses on easy visualization and interaction techniques for architects. “Ecotect -V 5.0” [8] visually displays the pattern of multiple shadows projected during a period of time on a single diagram. However, it does not quantify the amount of light received.

SPOT features two distinct but complementary components: (1) “Time Projection” and (2) “Navigable Animation.” SPOT employs Java 3D technologies to enable virtual interaction in a navigable 3D world. Below we briefly explain these two functional modules of the system.

The “Time Projection” module of SPOT accepts the following inputs: (a) a 3D model, (b) a period of time (e.g. 8AM-5PM during January-March), (c) a geographical location (selected from a list or the latitude, longitude, and time zone inputs) as well as the orientation of the model. The “Time Projection” module consists of two views of data: (a) the spatial lighting distribution displayed on a selected surface in the navigable 3D world; and (b) the temporal distribution of light represented in graphic calendars.

Designers first sketch a boundary shape on the 3D model indicating the area for simulation. SPOT then generates a representation of the spatial distribution of the illumination level on a selected surface over time. The selected surface is then rendered with colors of varied gradients. The color for each pixel of the surface is a result of the calculation of the number of times it is illuminated at regular intervals during the selected period of time. Designers can adjust several parameters such as the size of pixels, and their transparency. To inform designers of the percentage of illumination for each pixel over time, SPOT tests each pixel for illumination twice an hour and twice a month per pixel. Designers can also adjust the frequency of this calculation.

SPOT also enables designers to visualize the temporal information of light distribution over time for a given point. For each point clicked on the 3D model, SPOT generates a calendar diagram of a chart where the X and Y axis represent the months of the year and the time of the day. The color of each cell of the calendar is the result of the calculation of the light amount reaching this specific point.

The “Navigable Animation” module of SPOT facilitates interactive navigation through both space and time. SPOT supports time dependent phenomenon of sunlight display in the 3D space. To navigate the temporal variables, designers can click the forward and backward buttons to change the time or date. A simplified ray tracing function is implemented in SPOT to rapidly render the shadow cast effect on the selected area for a given date and time. Designers can also move their view positions in the 3D environment to examine the lighting effect from different locations. Informal user study shows that designers welcome the ease-of-use of 3D and animation interfaces.

SPOT is implemented on top of Space Pen, a 3D geometry-editing engine that has many functions and features to support collaboration over the web. As a result, designers can make simple modifications on the model’s geometry using SpacePen’s 3D editing functions. Daylight performances on these modifications can then be assessed in SPOT. Future work would include functions for generating elaborated design elements such as shading devices. SpacePen also provides annotation functions to document design rationale. We are also improving SPOT to facilitate special annotation of the lighting simulation result for remote collaboration over the web.

2- DAYLIGHTING AS A DESIGN FEATURE

2.1/INTRODUCTION

The sun provides the most pleasant and ecological energy source to light and heat buildings. The use of sun in architecture dates far back in history, but modernist architects developed a significant interest in daylight at the beginning of the 20th century. Aware of the hygienist movement's ideas, they considered it not only a necessary requirement to provide healthy environments, but also a plastic element to compose with. In the *Charte d'Athène(1933-1942)*, the Swiss architect Le Corbusier declared, "Each dwelling must have a set amount of hours for daylight to enter." In an interview, he further declared, "I use light abundantly, as you may have suspected; Light for me is the fundamental basis of architecture. I compose with light". Because efficient daylighting helps save energy and is beneficial to health, sunlight remains an important issue for architects and urban planners today.

2.2/ ENERGY SAVING

In designing sustainable considering daylight related issues results in reducing the building's overall energy consumption. For this purpose, the use of natural lighting should be maximized and heat gains generated by the sun need be controlled.

Artificial lighting consumes approximately 25 percent of the electric energy used in the United States. Heat generated by the electric fixtures also increases the loads imposed on the cooling equipments and indirectly results in an additional consumption of energy.

The study "*SKY LIGHTING AND RETAIL SALES-An Investigation into the Relation Between Daylighting and Human Performance (1999),*" [2.1] conducted by Pacific Gas and Electric demonstrate that efficient daylight design results in energy saving (between 30-60% in commercial and institutional buildings). A daylight efficient design reduces the building operating costs by minimizing the use of artificial light.

In addition, heat gains resulting from the sun shining through the apertures of a project can contribute to the heating of interior spaces. On the other hand, uncontrolled sun penetration can result in unwanted heat gains. In temperate climates, capturing the energy of the sun during the winter and keeping the sun out during the summer has significant impacts on energy consumption. By saving energy, implementing smart design will also have a part in reducing gas emissions in the atmosphere.

2.3/ SUNLIGHT AND PRODUCTIVITY

In addition to energy saving, sunlight also provides a healthier and more stimulating environment. In "*Daylighting-Performances and Design*, [2.2]" Gregg D. Ander explains how, "daylighting creates a more pleasing and productive atmosphere for the peoples within [it], provides a link to the outdoor environment, [and] supplies a perpetually evolving pattern of space illumination."

“*A Study into the Effects of Types of Light on Children – A Case of Daylight (1994*) [2.3]” carried out by Robbery Warren E. Hathaway, Ph.D., demonstrates the influence of full spectrum light upon students. The physicians, educators, social workers, nutritionists, and dentists that conducted the study found that students under full spectrum light with a trace of ultraviolet light learned faster, tested higher, grew faster, had a third fewer absences due to illness, and even two-third fewer cavities than expected.

The study “*Sky Lighting and Retail Sales,* [2.1]” also examined the correlation between occupant productivity and exposure to daylight within retail and school buildings. The study found that students performance on tests significantly improved (10-20%) and sales increased by an average of 40% when working spaces were exposed to daylight.

2.4/ SUNLIGHT AND HEALTH

Daylight within buildings not only improves the occupant’s productivity, but also influences their health. There is no life possible without light. By placing 2000 mice in the same environment but under four different types of light, John Ott, PhD (“*Health and Light*” 1976), proved the importance of natural light. The mice developed diseases and died in 7.5 months under pink fluorescent lights; under cool white lights (usually used in standard office environments), they survived 8.2 months; and under natural sunlight, they lived during 16.1 months.

Natural light also reduces stress, anxiety, depression, migraines, and sleep disorders. It improves immune system functions and metabolism by lowering cholesterol levels and increasing calcium utilization. Exposure to daylight also results in concentration and visual acuity improvements. Cardiac outputs are improved as well. Natural lighting is not only a pleasant feature in a building, but it has a significant influence on one's wellbeing.

Some people have a lack of light sensitivity that is severe enough to provoke sadness, anxiety, irritability, and violence. These are the symptoms of the Seasonal Affective Disorder (SAD). In the article "*Seasonal Affective Disorder (SAD): About Light, Depression & Melatonin* [2.4]", Gila Lindsley, Ph.D., explains that this disease, sometimes labeled as "the blues of wintertime," results from a lack of exposure to bright light. This typically occurs during winter under temperate latitudes when the daylight hours decrease. Daylight within a building is important because it provides the occupants of a building with healthy spaces which help prevent this disease.

2.5/ CONCLUSION

The sun affects all buildings. It is probably the most enjoyable, economical and ecological energy source that architects will ever use to light or heat a building.

It not only provides the users of the spaces with a motivating atmosphere, but also enhances their physical and mental health. It helps conserve energy and enhances one's productivity; therefore, maximizing daylight penetration in a project is beneficial for both the manager and the occupants of a building. However, certain lighting problems and opportunities that are amplified under a clear sky should be controlled.

3- DIRECT SUNLIGHT RELATED ISSUES CONTROLLED DURING THE BEGINNING OF THE DESIGN PHASES

3.1/ INTRODUCTION

The quantity of light that penetrates a space is an issue. In situations with high light intensity, our eyes can adjust in order to avoid discomfort, but a lack of light intensity reduces the human's ability to see. The more light enters a space, the better our vision is. Certain illumination values are expected depending on the activities of the occupants. To design a light efficient building, the architect's concern is to maximize the amount of light that enters a project and make it penetrate interior spaces as deeply as possible.

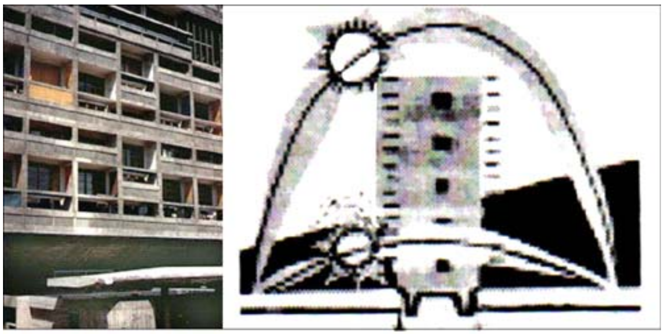
Light intensity depends on geographical location and spatial variables. Sky conditions also influence the nature of the sunlight that enters a project. An overcast sky results in a uniform light; in this case, the sun's position does not have a significant influence. A clear sky is brighter, resulting in a more intense light. The illumination levels also depend on the project's surrounding objects such as buildings, vegetation, or the relief of the site. Sky condition and sun position depend on the climate zone and the geographical location, whereas external obstructions are purely geometric issues.

Increasing the aperture's size and number maximizes the quantity of light that penetrates the building. Certain problems that might occur when daylighting is employed are strongly amplified under clear skies and these problems must be controlled. I will demonstrate that direct sunlight-related issues depend on design variables that are usually treated during the earliest, most conceptual stages of the design. The purpose of this research will consist in providing architects with a tool that will help them preview those issues.

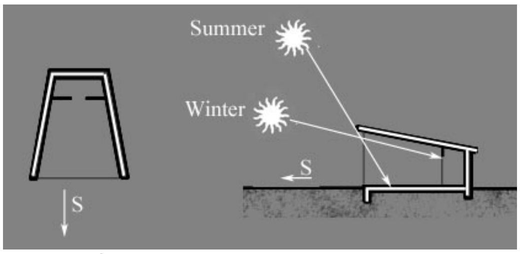
3.2/ SOLAR HEATING

In a building, the sun emits heat energy that travels through the glass of the openings and heats up the interior spaces. The energy accumulated inside the building radiates heat of a longer wavelength which is no longer able to escape through the glass; this energy is trapped in the room that gets warmer. This phenomenon is called the Passive Solar Gain or the greenhouse effect.

In temperate climates, a primary design goal is to capture the energy of the sun during winter but not during the summer. For this purpose, architects want to optimize the geometry of their project to take advantage of the sun's path variation over the seasons. Le Corbusier's solar principle became a reference, although the paper "*Les ambiguïtés d'un dispositif solaire de référence : la loggia de la maison radieuse (1999)* [3.1]*,"* reveals that he failed to implement it in the "Unités d'Habitation de Marseille." His concept was to protect the facades that contain openings with overhangs that cast longer shadows on vertical surfaces during the summer, when the solar altitude is increased (Fig.3.1).



(Fig 3.1): Façade of the Unités d'Habitation-Marseilles (Left) and Le Corbusier's solar diagram (Right).



(Fig 3.2): Socrates' solar house principle.

In his lecture "*The Evolution of Solar Architecture* [3.2]," S. Robert Hastings found that the concept of a passive solar house has existed for a long time. According to him, using the solar energy within buildings was a know art for Socrates (400BC) who advised orienting a transparent façade toward the south to maximize its dimensions (Fig 3.2).

The factors that have an effect on the building's solar energy performances are designed during the conceptual design phases. These include the orientation of the edifice and its mass, and its relations with the surrounding elements. The case of the south façade is certainly not a unique technique architects could put into practice to produce sun energy efficient constructions. To develop innovative designs that make use of the sun's path variations over a year, the amount of direct sunlight penetration that enters a project over the seasons must be quantified.

3.3/ GLARE

Glare or excessive brightness contrast within the field of view is responsible for visual discomfort. In his paper, "*TOWARD A DEFINITION OF GLARE: Can Qualitative Issues be quantified?* [3.3]," Marc Schiler explains that the main difficulty of previewing glare lies in the fact that it is not only a quantitative issue; contrast ratios are useless to predict whether an occupant of a space will experience glare or not. However, a great deal of research has tackled the problem more subjectively and concluded that there are certain factors that generate glare discomfort.

The dominant component is the luminance of the light source that could be estimated at the beginning of the design phase. The luminance of the background and the size and number of glare sources depend on details such as the colors and texture of the surfaces, as well as the number, proportions and size of the apertures. Glare problems can be fixed by filtering daylight by using elements such as vegetation, curtains, or lattice systems. These filtering elements are often added during advanced phases of the design process. The architect should be aware of glare problems in a project as soon as possible in order to integrate the cost of shading devices in the project's budget. The problem can be fixed early on by considering direct sunlight penetration.

Orientation, size and proportions of the openings, the building's shape and its geometric relation to the surrounding objects are factors that could be considered at the beginning of the research if the tools were available to quantify the amount of direct sun light penetration. Although the quantity of light entering a project is not the only factor, glare issues typically occur under clear sky. It can be controlled during the beginning of the design phases by avoiding direct beam light penetration in spaces where the occupants perform critical tasks. SPOT would help users identify potential areas in the project where glare could occur by revealing areas that get a lot of beam light and the length of time during the day that it receives the light.

3.4/ PROTECTING FRAGILE OBJECTS FROM DAYLIGHT.

Light can be responsible for destroying fragile objects. In her article "*The Effect of Daylight*, [3.4]" Rebecca Ellison, conservator at International Fine Art Conservation Studios in Bristol explains that daylight might damage vulnerable objects, especially those made of organic materials. Solar radiation causes materials to be damaged and colors to fade. Heat fluctuation, which causes expansion and contraction of materials, is responsible for the deterioration of rigid objects.

(Fig 3.3): The new Bibliotheque National. The transparent towers protected by wooden shutters.

Ellison emphasizes the importance of controlling exposure levels to daylight, especially direct sunlight which is the most intense. Failure to consider this phenomenon had consequences on the late French President Francois Mitterrand's last “grand projet,” the new Bibliotheque National (National Library). The article “The *Bibliotheque National*” – published in “*Bonjour Paris* [3.5]” by M. Andrew in December 1998 divulges the errors that architect Dominique Perreault committed when designing this project. Because Perreault’s original concept was to store them in four transparent towers, the books in the library were to be exposed to daylight. He did not achieve his goal, “because protective wooden shutters [would] have had to be installed on them to protect the sun-shy books.” Besides, those devices were useless in controlling heat gains generated by the sun. For a project like this, direct sunlight illumination must be controlled at the conceptual phases of the design, because it might be incompatible with the purpose of the building.

3.4/ OVERSHADOWING

Buildings in urban settings might block the view of the sky from adjacent properties. Shadow casting reduces the solar energy available for the surrounding structures. The city codes of many cities in the United States attempt to tackle this issue. San-Francisco’s planning code (ARTICLE 1.2-SEC. 147) indicates that “The amount of area shadowed, the duration of the shadow,” are factors that must be taken into account to reduce substantial shadow impact on public spaces. The City of Seattle’s planning code requires architects to “provide for adequate spacing between existing and proposed towers in order to minimize blockage of views from public places, and to minimize casting of shadows on public places (SMC 23.45.066 High-rise - Structure height. A. Maximum Height)”. The City of Los Angeles CEQA Thresholds Guide

gives more precise recommendations. A project’s impact is considered significant when “shadow-sensitive uses would be shaded by project-related structures for more than three hours between the hours of 9:00 a.m. and 3:00 p.m. Pacific Standard Time (between late October and early April), or for more than four hours between the hours of 9:00 a.m. and 5:00 p.m. Pacific Daylight Time (between early April and late October)."

These articles reveal that there is no standard representation to assess this issue.

Shadows cast by structures vary in length and direction with the seasons and with the time of day. The factor that determines the geometry of the shadows cast by an edifice is its mass. Planners, designers, politicians and citizens need to have the right tools available to quantify the direct daylight available for a property over time.

3.5/ CONCLUSION

Since natural light is the best form of illumination for interior and exterior spaces, architects should use daylight abundantly to create sustainable buildings. Indeed, besides being an extremely pleasant source of light, daylight is a necessary element to keep us healthy, and motivated to perform our activities.

Thermal heating control, glare, overshadowing and the destruction of fragile objects are issues that are exponentially emphasized under a clear sky.

The characteristics of a project that will have an influence on direct sunlight issues are typically decided during the beginning of the design process. These include the structure orientation, its general volume, and its relation to the natural or built environment.

The strategies employed to design sunlight efficient buildings are most effective when considered during the early phases of the design process, when the design is still being analyzed and decisions regarding geometry, and orientation can still be changed.

In the next part, we will discuss the existing strategies and methodologies architects could implement to assess the direct sunlight within a project.

4-THE RIGHT TOOL AT THE RIGHT TIME

4.1/ Introduction

The following will illustrate the requirements for a tool that allows designers to assess the sunlight efficiency of their projects during early design phases. An understanding of the cognitive strategies employed to generate ideas would indicate ways to help designers design. A range of commercial CAD modelers supports shadow cast projections, and the sunlight effects on a project can also be simulated through physical models. However, they do not provide real time feedback of lighting effect over time. Using them as a design analysis tool requires a great deal of trials to be performed on different days of the year, time of the day and at several viewpoints.

4.2/ The basic requirements for a tool aimed at studying direct sunlight.

An understanding of the strategies employed by designers in generating ideas edifies that rapid assessment of a sketch's quality is of utmost importance during the beginning of the design process. The articles "*Design Protocol Data and Novel Design Decisions* [4.1]" by Omer Akin and Chengtah Lin and, "*On the Analysis of Intuitive Design Processes* [4.2]" by Charles M. Eastman relate investigations into the cognitive strategies designers employ. Both of these articles consist in studying a subject's behavior in the form of a protocol to access their design thinking.

In "*Design Protocol Data and Novel Design Decisions*, [4.1]" the activities observed while a designer is working are: thinking, writing, examining, and listening. To enable the experimenter to interpret the subject's activities, the subject is instructed to think aloud. Although designers usually do not speak while working, the speaking activity is an echo that obliges them to structure their reasoning.

Even though the design process is very individualistic, the experiments resulted in the recognition of quite a few patterns of the strategies designers employ. One of them is of particular interest for our purpose. Eastman points out what he called the “trial-and-error process,” which consists of sequentially processing data by drawing an idea and then evaluating it. This process is comparable to the loops implemented in computers- information is continually processed until a satisfying solution is found.

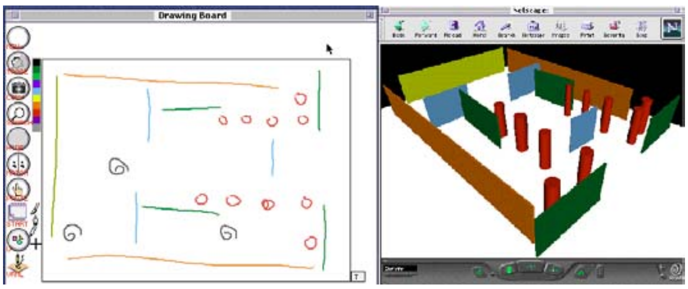
The aim of designers is to explore a variety of concepts and assess their relevance. Our purpose is to enhance the quality of this evaluation which will enable them to identify what Eastman calls the “error”.

Although direct sunlight is an important issue, it is one of many design variables that must be considered, particularly during conceptual design phases. Time spent considering this specific issue must be balanced with time investment for other design considerations. For all these reasons, a tool aimed at helping designers evaluate their daylighting strategies during early phases of the design must be able to generate rapid simulations. In addition, the strategies implemented to visualize the result of the calculations performed by such a tool must allow the user to rapidly understand and read the outputs.

4.3/ Early 3D model generation; can we implement computer-based tools during the early phases of the design?

Since daylighting is an issue that occurs in the tri-dimensional space, a computer-based design tool aimed at simulating sunlighting requires the creation of a three-dimensional description of the project’s geometry. Today, architects usually produce Virtual 3D models for the communication purpose only; photo realistic pictures of their project can be rendered when the design is done using computer graphic methods. Because virtual 3D model are typically created during late phases of the design process, daylighting simulations are typically performed using conceptual physical models. However, the development of object oriented CAD program such as “Graphisoft-Archicad” and recent investigations for rapid 3D prototypes generation might make computer-based (early design) tools possible soon.

In “Graphisoft-Archicad,” the description of the three-dimensional geometry of a project is not an extra step because the elements placed on the floor plan view already contain information about their three-dimensional properties. These include their height, thickness, material and more. However, the method requires too much detailed information to use object oriented CAD programs during the very beginning of the design process. The abstract level that designers obtain using the traditional medium of sketching is not supported.

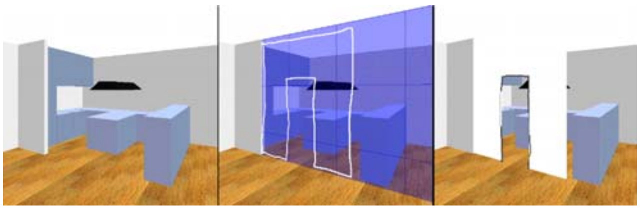


(Fig 4.1): Floor plan sketched using “SketchVR” (Left) translated into a 3D model (Right).

In her investigation, *“The Right Tool at the Right Time,* [4.3]” Ellen Do highlights the importance of sketches as a design tool. In “*VR Sketchpad”* she built a software application for 3D model generation from early sketches. Her paper, “*Create Instant 3D Worlds by Sketching on a Transparent Window*,” describes her project as a “pen-based interface” software that features “3D world creation using freehand drawing [, which] depends on users’ traditional understanding of a floor plan representation.” Sketch-3D interprets the marks drawn on a “cocktail napkin” to indicate the placement of the elements that compose the architectural space. Typically, the program interprets lines as walls, circles as columns, and an editable library of symbols recognizes more complex elements such as furniture and more. Then, a rapid 3D prototype is generated from the interpretation of the sketch.

(Fig. 4.2): "Harold: Objects drawn on a billboard (Trees and "hammock"), and terrain edited from the interpretation of lines sketched on the 3D model.

The article "*Harold: A World Made of Drawings* [4.4]" Cohen J., Hughes J., Zeleznik R, discusses the issues related to a second approach for sketch based 3D model generation. It consists of drawing directly in an interactively navigable virtual 3D world. In Harold, a new scene contains an editable large planar region that simulates the ground and a sphere that encloses the scene to represent the sky; it is possible to draw on any objects contained in the scene. The main problem when sketching in a 3D environment is that it is tedious to draw directly in perspective; indeed, a sketch looks completely different when the point of view is changed. To address this perspective issue, Harold features a "billboard" functionality that consists of a virtual screen used as a support to draw on it. Objects drawn on a billboard can be rotated when the viewer's position is changed; which can be used to represent objects such as trees or characters whose orientation is not relevant.

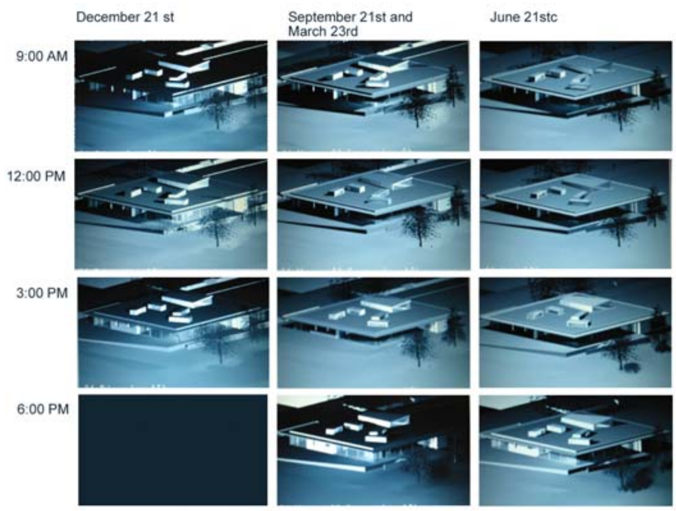


(Fig 4.3): "Space Pen". Sketching on a temporary surface drawing and filling it with a color to represent a wall in the 3D model.

"*Space Pen* [4.5]" by Thomas Jung (2001), a program primarily aimed at annotating 3D models to improve collaborative design investigates the possibilities of editing a model in a 3D environment as well. A 3D model is posted on the internet and the people involved in the project import it into Space Pen so that they can leave or review text or graphic annotations. The interface consists of two windows, one containing a floor plan window generated from the imported model, a second containing the representation of a navigable 3D scene (the user can interactively walk trough the 3D environment using the arrow keys). To leave comments, a clicked point in the 3D environment is marked with a yellow rectangle that represents a post it. At the same time, a window pops up to store textual information associated with this post it. To draw on the model, this program uses an approach that is similar to the one used in “Harold” to edit the 3D model.

Different approaches were envisioned for 3D model generation from early sketches. Since VR Sketch, Harold, and SpacePen are working prototypes, it is too early to predict which method will be implemented in future commercial software. However, it is reasonable to think that in the near future, designers will have tools to generate a 3D description of their ideas at their disposal. The 3D models generated will be available during the beginning of the design process to analyze them for various issues such as daylighting. Because SpacePen enables designers to both import and edit 3D models, it would be interesting to implement a tool for daylighting visualization on top of it. As a result, we could take advantage of the VR Sketch’s method for rapid prototyping and importing of the generated model. Modifications on the project’s geometry could be further performed on the model by drawing on it using SpacePen’s functionalities.

4.4/ Try and estimation of the in between values: a tedious method.

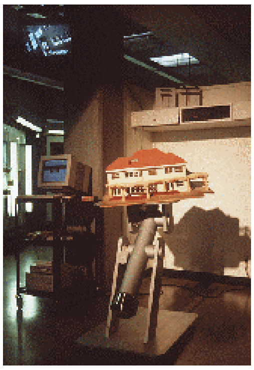


(Fig 4.4): Sunlighting effect on a project simulated at different time and dates, for a single point of view.

We understand that the time spent assessing a project’s daylighting performance is an important issue during the beginning of the design process. Assuming that conceptual 3D models are available for use as early design tools, we can study both physical and digital based strategies implemented to examine the repercussions of the sun on the geometry of a project. Existing tools enable designers to simulate direct sunlighting at a given moment. To get something close to a clear understanding of what is going on over time, a series of simulations must be performed at different moments. Light designers typically simulate sunlighting at different hours during the winter and summer solstice and during the equinox. The in between values can then be estimated. However, the following will demonstrate that existing tools do not meet the rapidity of simulation and visualization requirements for them to be used as early design tools.

Software programs such as Radiance, Lightscape and other commercial sunlight simulation modules are dedicated to architectural solar simulation. These ray tracing computer graphic methods can render photo realistic lighting effects of a space at a given moment in a viewpoint. The ray tracing model is a method that consists of modeling direct light reflection by tracing the path from the light source to the viewer's position. The lighting must be recalculated whenever the viewer or of the object moves. The color of each pixel of the picture that is rendered is determined by testing the rays against the simulated object. To calculate the intensity of each pixel, several parameters are taken into account including the surface, the visibility of the light source, the ambient light and the color of the object surface. Because every pixel contained in the picture is tested and since every ray can intersect many objects this method is computationally intensive.

Ray tracing is a very efficient method for photo realistic image rendering and provides a great deal of information. However, the use of ray tracing based method to simulate sunlighting over time requires large number of images to be rendered. This process requires constant adjustments of the suns position and multiple images need to be rendered. In addition, the process must be performed at various points of view to visualize it in the space.


(Fig 4.5): Physical model placed on a heliodon.

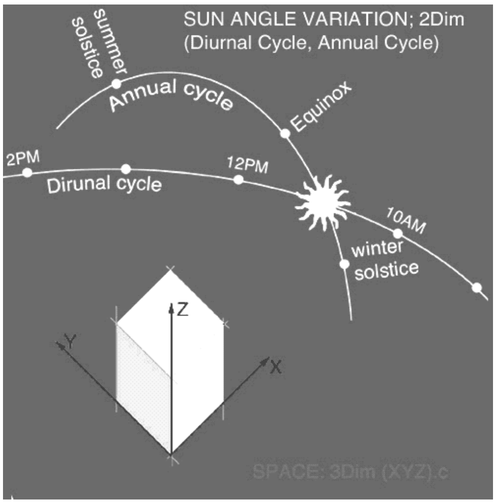
Direct sunlighting can also be analyzed through the creation of physical models placed in a heliodon. The system consists of two components. A projector represents the sun and a support allows precise orientation of the model in relation to the projector.
To simulate the sun's position at any time and for any location, the heliodon allows rotation in the x, y, and z axis. The result is a useful and detailed representation of solar patterns for given moment. However, the method is still limited to the simulation at a given moment. That is why the use of a heliodon requires a great number of manipulations to be performed to examine the implications of the sun path on a project over time.

Ray tracing based methods and simulations that make use of physical models are very efficient methods of simulating the lighting effect in an architectural space at a given moment. However, both representations do not provide designers with quick visualization and assessment of the phenomenon over time. The tools currently available for designers to predict sun penetration in a project do not meet the requirement of rapidity expected to implement them during the beginning of the design process. Recent research investigated different strategies to visualize direct sunlight phenomenon over a period of time.

5 - VISUALIZING A MULTI-DIMENSIONAL PHENOMENON

To rapidly simulate and assess the implication of direct sunlighting, the phenomenon should be automatically simulated for a period of time. Recent research explores this idea. We will see that the main difficulty is not in simulating the phenomenon, but providing architects with legible outputs of the simulation.

5.1/ Direct sunlighting: a 5D phenomenon.



(Fig 5.1): The 5 variables of the direct sunlight phenomenon.

Sunlighting is multi-dimensional issue. The phenomenon must be simulated in three dimensional space (X, Y and Z). In addition, the sun position fluctuates with time, a variable that has two components: the sun angle variation with the diurnal cycle of a day and its fluctuation with the annual cycle of a year. To preview the effect of sunlighting on an architectural space over time, designers must navigate a 5D space that consists of the three dimensions of the space and the two dimensions of time (annual and diurnal).

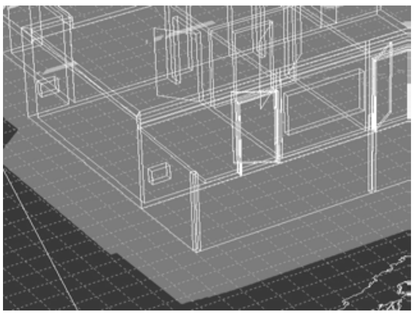
5.2/ Visualizing multiple dimension spaces on a 2D screen.

In the article "*Four-Space Visualization of 4D Objects*,[5.1]" Steven Richard Hollasch explains that the difficulty of visualizing multiple dimensional spaces lies in the fact that "there are no solid paradigms for three-spaces creatures such as ourselves. This difficulty is best understood by imagining the plight of two-space creatures trying to comprehend our three-space world." However, the sun path variation over time is a concrete phenomenon that we can experience and observe. To visualize time dependant phenomenon such as sunlighting, the main difficulty does not consist of understanding an abstract object but the display of a multiple 5D space in a 2D display. The representation used to quickly visualize the phenomenon should enable designers to compare the temporal and spatial variables. Since daylighting is a concrete phenomenon, the visualization method should provide intuitive representation.

5.3/ Related work: different approaches to display the variables of the phenomenon.

Many methods have been developed for this purpose of displaying multiple dimensions phenomenon on a 2D screen. I will provide examples approaches that are of particular interest to architects because they are graphic-based. We will analyze the potential of these main methods to visualize sun lighting.

a) Ecotect: projecting all the variables on a single diagram.



(Fig 5.2): "Ecotect": Layering of the shadows cast using "the Shadow Range" function. Location: Seattle - July 9th - Time range: 11:00-14:00

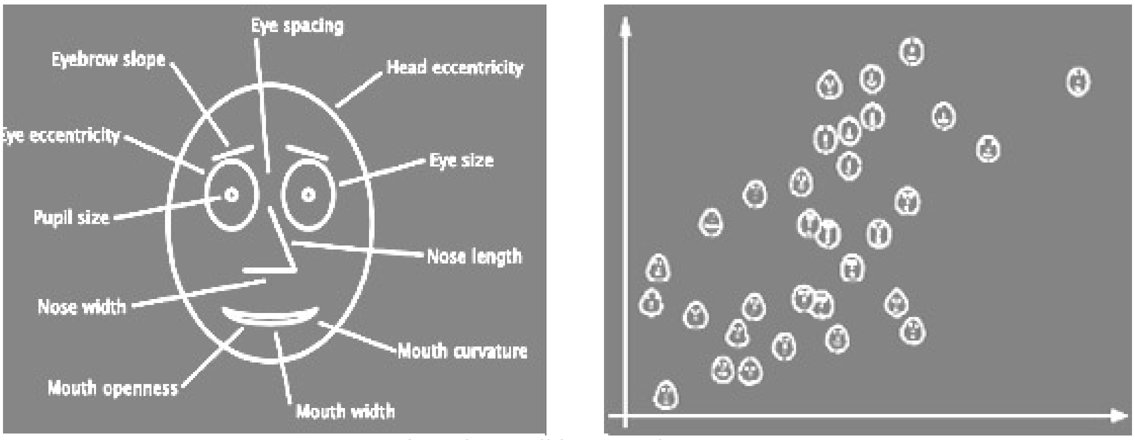
The building analysis software "ECOTECT v5 [5.2]" integrates a 3D modeling interface and features many advanced functions aimed at helping architects assess their environmental strategies. ECOTECT is intended for use during the beginning of the design process.

To represent multiple shadows on one

diagram, ECOTECT features a “Shadow Range” function that produces a shadow profile. This function generates a representation where the shadows are drawn at regular intervals over a day (e.g. every half hour). This results in a pattern of shadows running from west to east. The same simulation can be generated for a range of days throughout the year.

This representation helps designers understand the geometrical relationship between a project and the sun’s path. However due to layering, multiple shadow casts from different times make it impossible to quantify the amount of light received. ECOTECT only displays the boundary of the area that can be shaded at least once during the selected time period (fig 5.2). Projecting all of the variables of the sunlighting phenomenon in a single view results in a useless representation when one’s goal is to understand the spatial and temporal distribution of sunlight. A range of methods were developed to visualize the relations between multi-dimensional data.

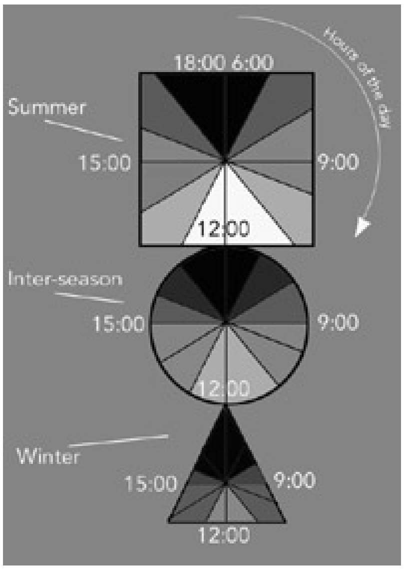
B-Glyphs/icons : rearranging the coordinate system to be non-orthogonal.



(Fig 5.3): The eleven characteristics of the Chernoff faces (Left) and their display in an orthogonal coordinate graph.

The variables of a multi-dimensional space can be displayed on a single diagram by rearranging the coordinate system to be non-orthogonal. The graphic properties of a glyph or icon can convey lots of information. A famous

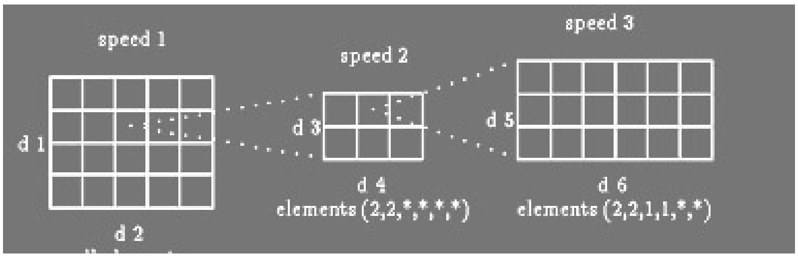
example of this approach is the “Chernoff faces,” a method developed by the statistician Hermann Chernoff.
The representation consists of visualizing the value of the variables through the traits of a face icon. The faces vary in eleven characteristics: eyebrow slop, eye spacing, head eccentricity, eye size, nose length, mouth curvature, mouth width, mouth openness, nose width pupil size, and eye eccentricity represent a value of one of the dimensions involved (Fig 5.3). The icons can be placed in an orthogonal coordinate graph; in this case, it is possible to compare thirteen variables.



(Fig 5.4): An icon composed of 3 shapes to represent the variables of the direct sunlighting.

We analyzed the potential of this approach for the display of both the temporal and spatial variables of the sunlight phenomenon. We created an icon composed of three shapes that would represent the seasons; a triangle would stand for the winter, the inter-season could be represented by a circle and the summer could be a square. Each shape would be divided in segments representing the hours of the day. The segments would be colored with a color gradient that would display the amount of light received for each segment of time and for each season. The shapes would vary in size, depending on the amount of light received over each season. Those “tri-partite” icons would be placed in a floor plan.
To assess its relevance, this icon-based approach should be further tested. So far, we haven’t implemented and developed this method.

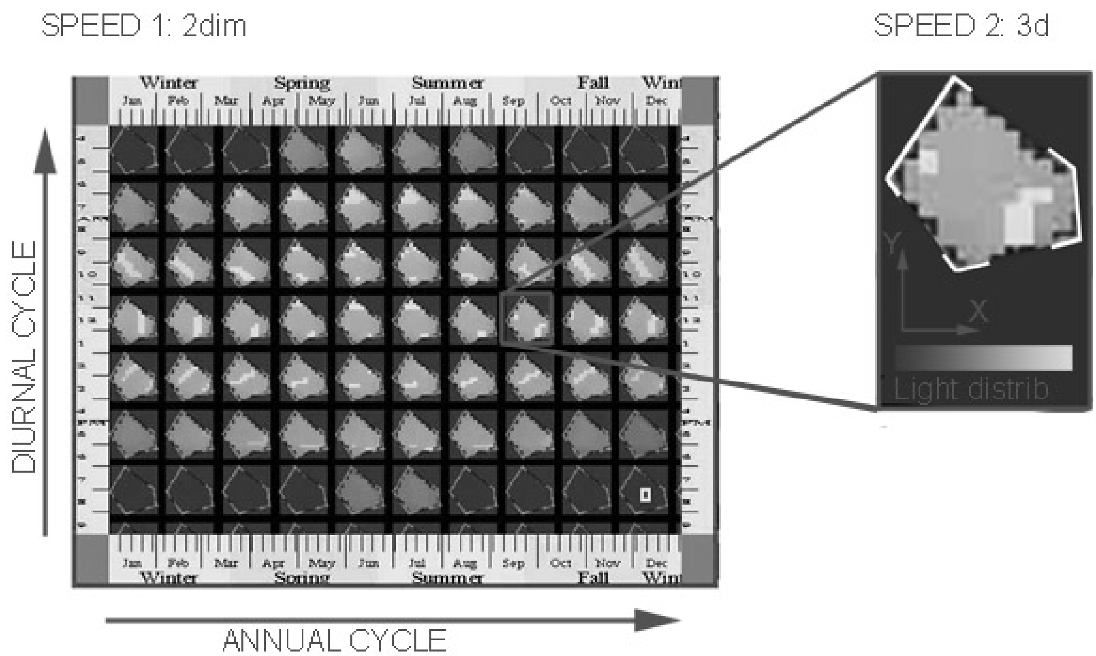
C-Embedding data dimensions to form composite spatial dimensions: the dimensional stacking



(Fig 5.5): The display of a six-dimension space in a dimensional stacking. The variables are organized in three speeds (pair of variables) embedded each other.

To form a composite spatial dimension display, the "Dimensional Stacking" is a recursive projection method that consists of embedding dimensions within other dimensions. Organized by pair, the dimensions (displayed on the X-axis and Y-axis) form a grid called "speed". The next two dimensions create the next speed that is embedded in the cells of the previous speed (e.g., in fig 5.5, D3 and D4 form the speed 2 that is embedded in the speed 1 which is formed with D1 and D2). This embedding process continues until all the variables are displayed. The position of a speed K in a speed K-1 determines its relation to variables of the speed K-1. Effectively, this approach supports the display and the comparison of all the variables on a single diagram.

To display the sunlight distribution over a year, Dan Glaser, a researcher from the University of California-Berkeley, implemented a variant of the dimensional stacking. His software program displays the illumination levels calculated by the program Radiance.
The distribution of the illumination levels at series of given moments is displayed on floor plans colored with varied gradient shades. Those images convey three variables- the X and Y spatial variables (orthogonally projected in a plan) as well as the light levels. In the classical form of the dimensional stacking, the floor plan pictures correspond to speed number 2. However, they communicate three dimensions instead of just two. A number of these floor plans are generated to simulate the lighting effect at different times of the day and different days of the year. Accordingly, the images are then displayed and organized in a calendar that conveys the temporal dimensions. This particular calendar will correspond to speed number one in the classic dimensional stacking model.



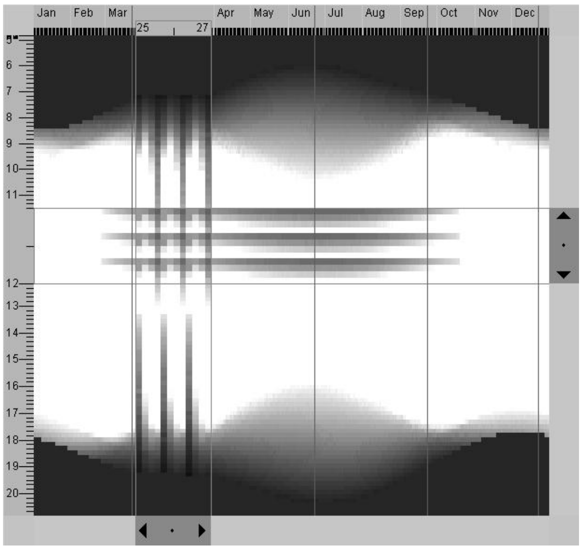
(Fig 5.6): Dan Glaser's software. Floor plans illuminated at different dates and time and displayed in a chart that communicates the temporal variables.

This representation enables to show an overview of all the data in a single display. However, to optimize the shape of a building to make it sunlight efficient, designers need to navigate the three dimensions of the space. In Dan Glaser's project, the illumination levels are displayed in a work plan, which means that one of the three spatial dimensions is lost. The floor plans are displayed in the cell of a calendar chart. Another limitation of this representation is that those cells are too small to display an entire building. Designers can assess the sunlight penetration only in a single space.

D- Panning or zooming to maintain context: the focus plus context

An inherent difficulty for displaying multiple dimension datasets on a single diagram is keeping an overview of all the data and providing detailed visualization of critical areas at the same time. The focus plus context method graphically addresses this problem; the focus displays specific dimensions in the context of the whole. Various variants of the focus plus context explored different ways of magnifying the graphic in the focal area; these include panning or zooming (with diverse types of lenses) into particular dimensions.

Glaser designed “*Space Series* [5.3]”, a new focus-plus-context technique for displaying both spatial and temporal data in a 2D plot. This project is intended to be used for visualizing time-dependant phenomenons. In his article “Space Series: simultaneous display of spatial and temporal data,” Glaser illustrates this visualization technique with a simulation of architectural daylighting.



(Fig 5.7): Space Series. The focus is set on March 25-27, from 11 am to 12 pm.

The example in Glaser’s paper is the display of the illuminations levels calculated for several dates of the year and time of the year by the software Radiance. These calculations are performed for sixteen subdivisions of the simulated space that is divided into 4x4 areas. To present the outputs over a whole year, Space Series consists of a diagram that has two major axes representing the dimensional dimensions (day - time).

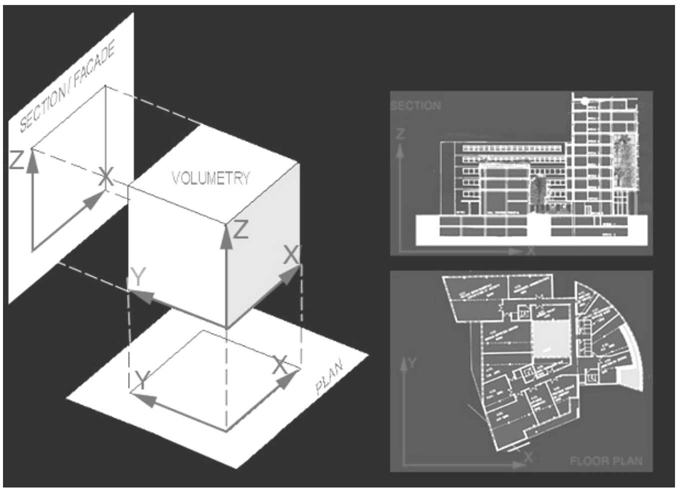
Initially, the graphic displays an overview of the data that is the average light levels into the whole space over time. Using the vertical focal bar, the user can focus on a given day. The horizontal bar enables him to set the focus on a time interval. The program will then reorganize the plot creating four types of regions: the fractional hour focus is a slide that shows the light levels in the space at different times, the day focus shows is another slide that displays the same information for a given day, and the context is still visible in the background. In addition, the focus intersection which is the intersection between a horizontal focus and a vertical one corresponds to a given moment.

This program was intended to display the data in a single plot, so that the users would not be forced to mentally link different plots together to understand the phenomenon. However, even though the context is displayed in the background, Space Series requires focusing on different dates and times to get an understanding of the daylighting over a year. For this reason, architects would still have to read several sub-plots.

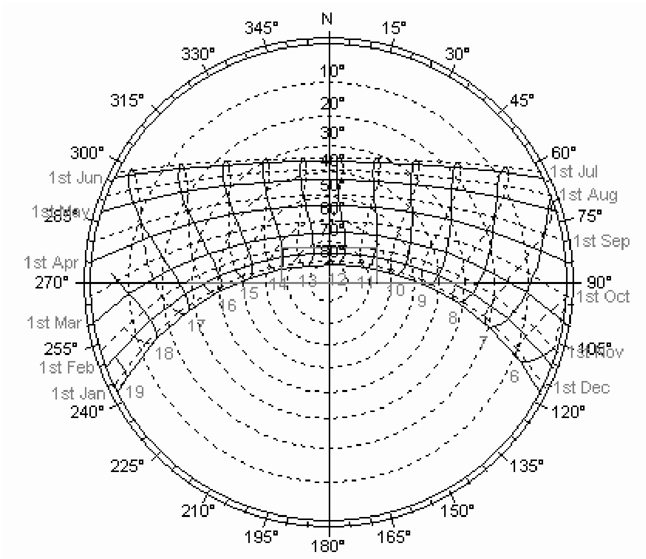
Another limitation lies in the fact that the specificity of this representation requires architects to have specialized knowledge of the focus plus context to use it. Since Space Series does not allow intuitive visualization of time dependant phenomenon, it is intended for use by light designers.

e-The Projection: using multiple views of the data.

Architects have traditionally used the orthogonal projection to represent the three spatial dimensions of a project on 2D media (paper or screen). The projection method consists of transforming an N dimension space into an N-1 dimensions space. The loss of a dimension requires additional views to be drawn; the plan sections convey the X and Y dimensions, and the Z dimension is displayed in vertical sections (Fig 5.6). To link two different views of the data, a dimension is represented on both plots. In the example of the figure "5.6," the X dimension is represented on the floor plan and on the section.



(Fig 5.8): The orthogonal projection, the X dimension links a plan and a section.



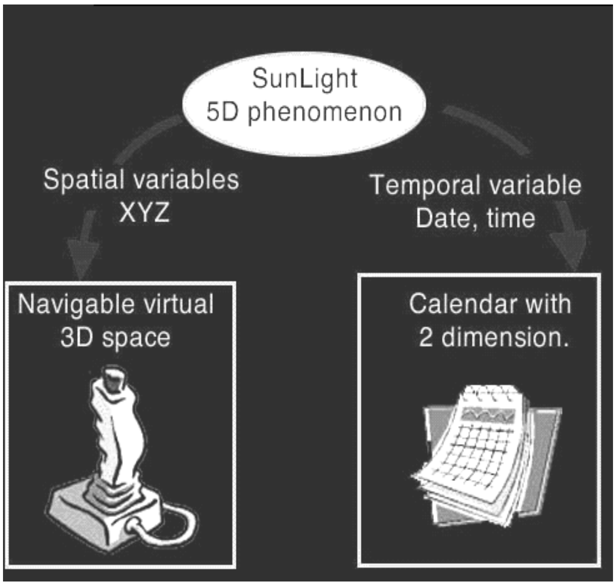
(Fig 5.9): Stereographic projection of the sun path

Different types of projections enable the display of multi-dimensional spaces. For example, the variations of the solar path over a year can be represented using stereographic projections. The sun angle is represented by an altitude that describes how high the sun is in the sky and an azimuth that informs us about its direction. Variations of the sun path over time are drawn on a diagram where the concentric lines represent the altitude and the radial lines represent the azimuth (fig.5.9).
The angles are considered in respect to a given point, which means that the cardinality of the X, Y and Z spatial variables is one. Therefore, this representation does not allows users to navigate the space.

For a long time, this representation was the only method available for architects to consider the sun path in a project. The part of the sky masked of a particular point can be projected to overlay the sun path diagram on it. Even though this method is tedious, it enables architects to determine when a point would be lit by the sun over a year.

There are no efficient methods that support quick visualization of the direct sunlight over time. In additions, architects have traditionally used the projection method for multi-dimensional space visualization. We therefore believe that they have the faculty to visualize phenomenon using various views of the data and mentally rearrange them together. For all these reasons we propose SPOT, a software program that would display the phenomenon in different media. Recent techniques such as VRML, Java3D or 3D video games engines support interactive navigation into virtual 3D spaces. These methods that enable the user to virtually walk through the space in real time represent three spatial variables on a 2D screen. The temporal variables (date and time) would be displayed in additional views that would consist of a 2D graphic calendar.

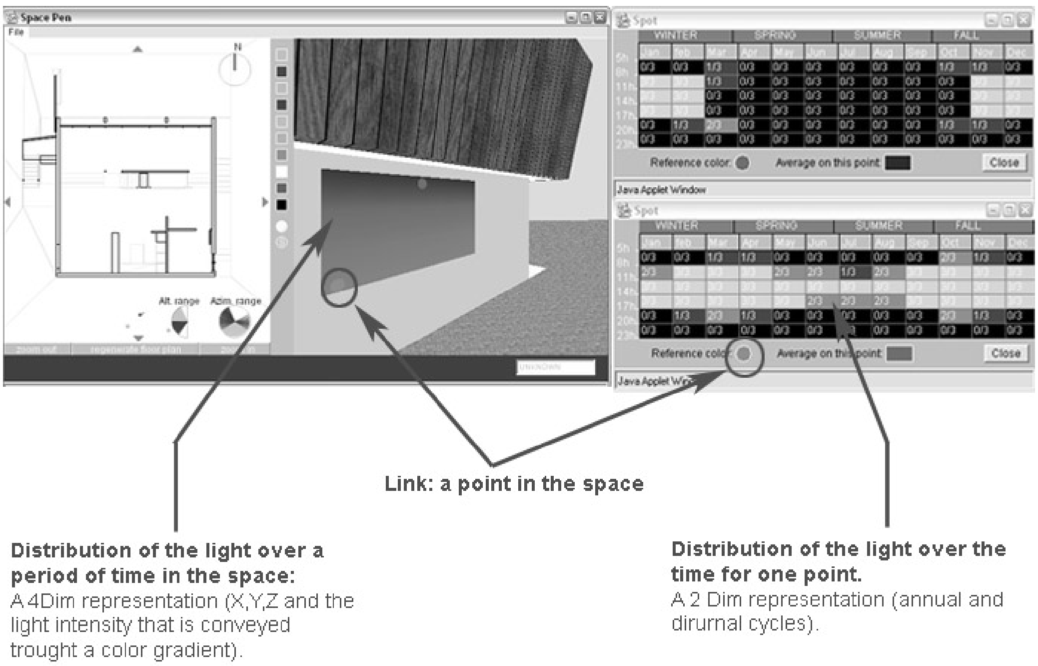
5.4/ SPOT: The Time Projection.



(Fig 4.10): The representation of the spatial variables into navigable 3D spaces and the projection of the temporal variables into 2D graphic calendars.

To help architects visualize direct sunlight penetration in a project, the main ideas behind SPOT is the display of the spatial distribution of light in a navigable virtual 3D space and the projection of the temporal variables in additional views. The use of navigable 3D spaces enables the display of the spatial dimensions in a same view; spatial data are not projected. To use it, the user sketches an area for simulation on the 3D model. The program then renders the surface with colors of varied gradients where the color of each pixel of the surface is the result of the calculation of the percentage of illumination over a selected period of time.

The distribution of light over time of a particular point is displayed in a 2D calendar chart where the X axis represents the months of the year and the Y axis the hours of the day. Each cell of the calendar is colored according to the percentage of illumination the point it is associated receives. Several calendars could be generated for a series of points in the project. To inquire this function, the user clicks a point on the 3D model. SPOT mark it with a sphere whose reference color would link the point in the 3D space to the calendar it is associated with.



(fig. 5.11): Spot: the time projection functionality.

5.5/ SPOT: The Navigable Animation.

Existing CAD programs such as “Autodesk VIZ” enable the user to animate the position of sunlight between two given dates. This feature makes it possible to render animation movies that simulate the direct sunlight effects on a project over time in a view point. The limitations of the representation are that the spatial variables are static and the linearity of the animation does not support the navigation in both temporal variables (day and time). Indeed, it is possible to simulate the variations of the shadow cast effect over a day at different times, or at a given time over a year. Another inconvenience of the method is that it is very time consuming.

The Navigable Animation functionality of SPOT enables the user to interactively visualize the shadow cast effect and animate it over time. A simple ray-tracing method implemented in SPOT rapidly renders the shadow cast effect on a selected area. Once an area is rendered, the date and time appears on the screen. By clicking a forward and backward button, designers navigate through time by changing the date or time. SPOT then renders the shadow cast effect on the selected area in real time.

6- SPOT IMPLEMENTED IN SPACE PEN

We presented Space Pen, a Java3D applet aimed at easing design collaboration by annotating a 3D model that can be posted on the internet. Space Pen provides a 3D navigable world. To ease navigation in the space, a cursor is displayed on a floor plan generated from the 3D model imported in the program.
Space Pen supports simple 3D model edition functions by sketching in the 3D model. This would enable one to make some changes on the 3D geometry and assess the consequence of these modifications on the project's lighting. SPOT is implemented on top of Space Pen because design decisions occur in collaborative work. Implementing SPOT in Space Pen results in an environment that supports sunlight simulation on the web and eases communication. The outputs generated by SPOT are also available online for remote collaborators.

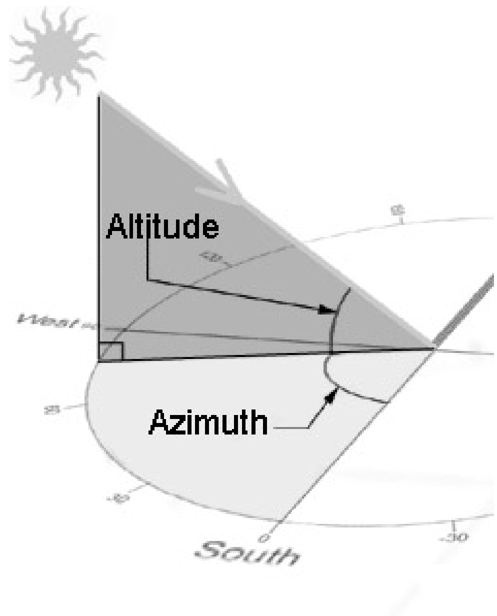
6.1/ Method implemented to test a point for direct sunlight illumination.

To determine if a point in the 3D model is lit by direct sunlight, SPOT calculates the sun angle for the specific location, date and time, and searches for intersections between a vector rotated according to the sun angle and one that originates at the tested point in the virtual 3D environment. If an intersection is found, it means that the point is shaded; otherwise it is lit.

a-Calculating the sun angles for the specified location, date, and time.

The sun angles consist of two values: the altitude angle and the azimuth angle.

>Definitions: Solar azimuth and solar altitude.



(Fig 6.1): The solar azimuth and altitude angles.

The altitude angle describes how high the sun is in the sky. It is measured between an imaginary line that represents the sunray that reaches the observer and the horizon. Its value is positive between the sunrise and the sunset.

The azimuth or zenith angle, informs us about the direction of the sun. It is the angle formed by the projection of a sunray on the ground and from the South direction. Its value is positive when the sun's position is East of South, and a negative azimuth angle indicates West of South.

>Method of calculation:

The first step is to determine the declination, the angular distance of the sun north or south of the earth's equator.

To calculate the declination, SPOT uses the "sunPosition" function from the "SunPosition.class" provided by OpenMap™ which precisely determines the solar position. According to the OpenMap™ documentation, this class "calculates the latitude/longitude on the Earth that is closest to the sun, i.e. the point on the Earth that the sun is over". All of these calculations are based on an epoch, or starting point from where the Sun's position is known.

The declination angle (D) is equal to the latitude of the point of the earth such that the sun is directly overhead. The hour angle (HA) expresses the difference between local solar time and solar noon. Although it is related to time measurement, its value is expressed in degrees. SPOT also determines the hour angle from the value returned by the “sunPosition” function. The hour angle is equal to the longitude of the point of the earth such that sun is directly overhead minus the longitude of the observer. From these values, the altitude (Alt) angle is calculated using the following formula:

sin Alt = (cos lat x cos D x cos HA)+(sin lat x sin decl)

where “*lat*” and “*long*” are the latitude and longitude of the observer; “D” is the declination; “HA” is the hour angle, and all the values are expressed in degrees.

Then, the azimuth angle (*azim*) is determined from the following formula:

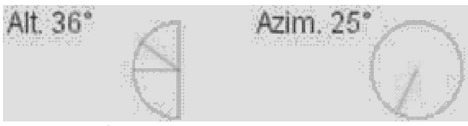
cos azim = (sin alt x sin lat – sin D) / (cos alt x cos lat)

if azim x HA>0 then azim = -azim

where “*lat*” and “*long*” are the latitude and longitude of the observer; “D” is the declination; “HA” is the hour angle, and all the values are expressed in degrees.

>Outputs.

Once “SPOT at a given moment” is launched, the program graphically informs the user of the solar angle at the date and time selected in the bottom panel. For a positive altitude angle, the lines representing the angles are yellow. If the selected moment appears to be at night, implying a negative solar angle, the program generates blue lines.



(Fig 6.2): Output: the solar azimuth and altitude angles.

b- Search intersections between any object in the scene and a virtual sun ray whose origin is a clicked point.

The idea implemented to test a point for direct illumination is straightforward. We called the function that performs this operation “fetchSunlight”. In our representation, a “sunray vector” is a vector rotated according to the solar angle and originates at the point that is being tested. The *“fetchSunlight”* function returns the value “true” if no intersection between any opaque object in the scene and a “sunray vector” exists.

To reduce the computation intensity, programmers usually implement a visibility algorithm for pre-calculation of the shaded areas at a given moment in the scene. SPOT tests each pixel’s value rather than pre-analyzing the geometry of the shadow cast for two reasons. First, SPOT is a tool intended for simulating the effect of direct sunlight over a period of time. As a result, the same point must be tested at different moments. It would be more time consuming to predetermine the shadow’s pattern at different moments than testing a series of points at different moments. The second reason is that SPOT is a tool aimed at visualizing a particular phenomenon that might have consequences on specific parts of the project; therefore, it analyzes particular areas rather than generating a full scene render.

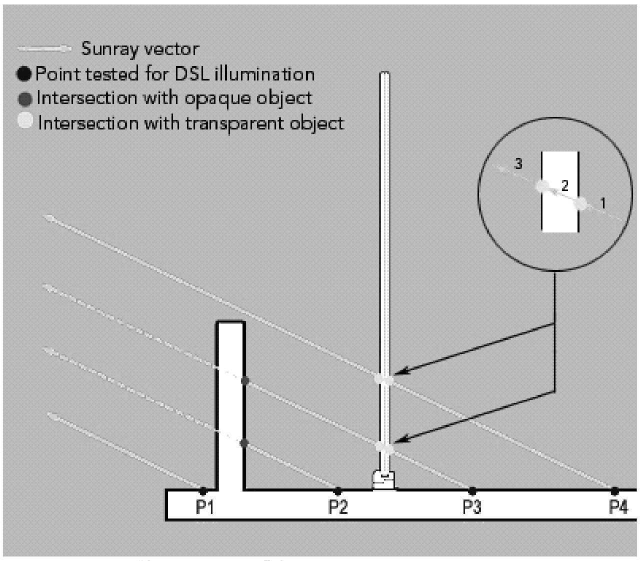
The transparency of the objects that compose the model is a factor that must be considered when simulating optical phenomenon such as direct lighting. For this purpose, we assume that direct light goes through transparent objects. To implement this in SPOT, we envisioned three different approaches.

The first approach consists of disabling transparent objects to be taken into account in the calculation. Interaction with Java objects requires their capabilities to be read and “picked” to be set. After a 3D model is imported in SpacePen, the function “setPickable” prepares each node to be accessed. While the “setPickable” function is processing, the appearance properties of the objects can be accessed and it is possible to specify that transparent objects should not be set as “pickable”. We did not choose to implement this strategy because objects must be picked in order to draw on them in SpacePen. This strategy would have resulted in the inability to draw on transparent objects.

The second approach is implemented while the 3D model is imported in SpacePen. The scene graph structure of Java3D features some grouping facilities. The nodes that compose a scene can be stored into different “BranchGroup” (group of nodes) whose capabilities can be changed at any time. We could store the objects of the model that is being imported in two different “BranchGroups”; one containing the opaque objects, the second the transparent ones. As a result, all the transparent objects capability to be picked could be turned off when “*fetchSunlight*” is processing. Once the calculations are performed, all the capabilities could be restored to draw on transparent objects again. The advantage of the two first strategies is that the transparency properties of the objects do not have to be considered while “*fetchSunlight*” is processing.

However, we might ultimately need to consider the objects penetrated by a “sunray vector”, whatever their transparency attributes. Future work would consist of heat gains assessment over time functions. For this purpose, the incidence angle of a “*sunray Vector*” on a transparent surface must be considered. For these reasons, we decided to consider the transparency of the objects while the “*fetchSunlight*” function is processing. This added more complexity to the algorithm. To access the transparency attributes of an object imported in a Java3D environment, it must be copied and converted into Java3D language. However, we tested two variants of the “*fetchSunlight*” function, one considering the transparency, the other one neglecting it. The first one appeared not to affect the speed of processing “*fetchSunlight*”.

The “*fetchSunlight*” function is recursive. Its algorithm is: if a “*sunray vector*” does not intersect any object (the point is lit), the value “true” is retuned. If an intersection exists, the transparency of the intersected object must be considered. If the intersected object is opaque (the point is shaded), the function returns the value “false”. If it is transparent, the “*fetchSunlight*” function calls itself recursively and takes the intersection as a parameter.



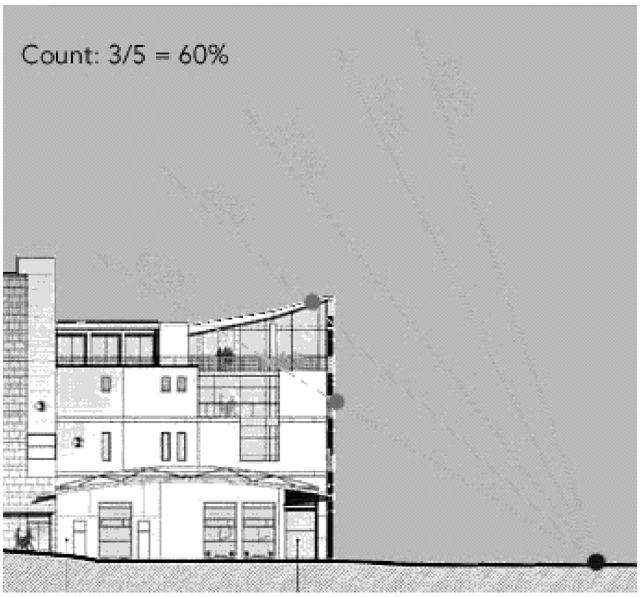
(Fig 6.3): The "fetchsunlight" function.

The following diagram illustrates the behavior of the *"fetchSunlight"* function in four different situations. In the first case, a point P1 is considered to be illuminated because the sunray vector that is associated with it does not intersect any object in the scene.

In the second situation, a point P2 is shaded because the function detected an intersection with an opaque object.

In the case of the point P3, the "*sunray vector*" penetrates two transparent surfaces that represent the pane of a window in the 3D model. The first intersected element is transparent. For this reason, the *"fetchSunlight"* calls itself and takes the intersection as a parameter. In this situation, this process is repeated because the "sunray vector" intersects a transparent object again. Finally, SPOT detects an intersection with an opaque surface. Therefore, the point P3 is shaded.

The case of the point P4 is similar. The difference is that the "sunray vector" never intersects an opaque object. P4 is also lit.



(Fig 6.4): Percentage of illumination incident on a point over time.

6.2 "Light_count( )"- Assessment of the amount of light incidence on a point over time

To statistically evaluate the percentage of illumination over time of a point, the “light_count” function of SPOT performs a series of tests for direct sunlight illumination at different times on regular intervals. A counter computes the number of times sunlight hits the tested point. A ratio is made comparing the counter to the number of tests performed at different times. Fig.6.4 illustrates this idea; the percentage of direct sunlight illumination is assessed for a point during a day, from the hours of 9:00 to 12:00. A test is performed every 30 minutes and the point is lit three times out of five tests (the ratio is 3/5 = 60%). Therefore, SPOT estimates the percentage of illumination over the selected period of time is sixty percent.

The default parameter of SPOT is to perform a test twice an hour and twice a month. However, users can decrease the test’s frequency. Each test for sunlight illumination consumes memory. Indeed, to maximize the speed of the computations but lose result precision, the amount of tests performed can be decreased. The frequency can also be increased, increasing the accuracy of the results. The duration of the diurnal cycle of the earth’s rotation is 24 hours; the hour angle is approximately 15 degrees. Performing tests twice an hour results in a precision level of 88%.

Imprecision level = (15 degrees / 2 tests) x 100 percents /360 degrees) = 2.08%

The earth declination variation is 46.9 degrees (from -23.45 degrees to +23.45 degrees compared to the equatorial plane) between the summer and winter solstices, which are spaced out by six months. When performing tests for sunlight illumination twice a month the average precision level is 99%.

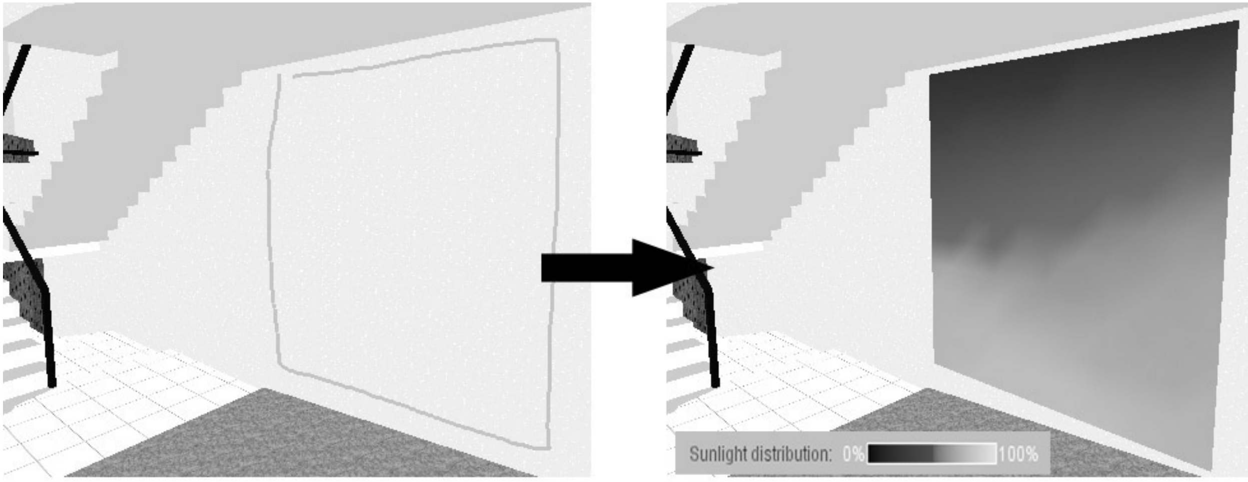
Imprecision level = (49.9 degrees / (6months x 2 tests) x 100 percents / 360 degrees) = 1.09%

Considering the diurnal and annual variations of the sun’s path, the average precision of the default settings is very precise.

A point can be tested for direct sunlight illumination for a discrete moment and its percentage of illumination over time can be statistically and precisely assessed. The following will explain how the result of these calculations is displayed in the 3D scene.

6.3 The display of the sunlight distribution in the space.

To specify a surface for simulation in the 3D space, the user draws a boundary area on the 3D model. SPOT then colors the selected surface in the 3D environment with colors of varied gradients. The colors of the surface's pixels are the result of the calculation of their percentage of illumination over time. The following will explain the implementation of this.



(Fig 6.5): Selecting an area for simulation by drawing a boundary shape on the 3D model (left). The display of the percentage of illumination incident on the selected surface (right).

A-Retrieving the coordinates of a stroke's boundary area.

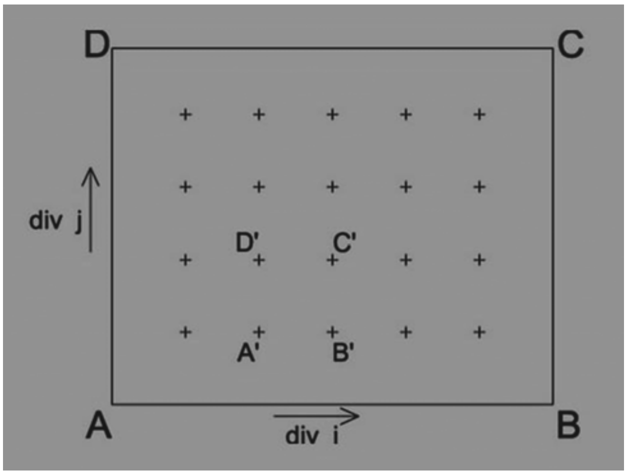
SPOT retrieves the coordinates of the bounding box of the stroke drawn using "Space Pen" functions. To ease freehand sketching on a model, SpacePen features simple shape recognition. Each time a line is sketched on the 3D model, SpacePen analyzes its geometry to determine if the user intended to draw a rectangle, circle, spiral or triangle. The program can then rectify the shape. Those shape recognition functions were also implemented to support gesture inputs. For example, when the user draws a straight line on the model and does not do anything after that, SpacePen generates a temporary drawing surface in the 3D model. This enables the user to draw not only on existing surfaces, but also in the space.

The geometric characteristics of each stroke drawn in "Space Pen", such as their bounding box, are available for use by SPOT.

* This part of the program returns four 3D points “ABDC” that are the vertices of the bounding area of the stroke drawn by the user.

B-Defining the cells of the surface’s characteristics (spatial coordinates and colors).

Once the vertices of the stroke’s bounding box are known, a loop implemented in SPOT divides the simulated area into cells. The approximate size of those cells is determined in the advanced inputs window of SPOT, which is labeled “space between the tested points.” Since SPOT computes the color for each cell’s vertices, this advanced parameter has an influence on the processing speed. The quantity of rows and columns that divide the surface into cells is calculated so that their size is as close as possible to the value of the “distance between the tested points.”



(Fig 6.6): An ABCD area divided into cells.

>Determine the dimension of the cells contained in the ABCD surface

*Declare the variables div_ I and div_ J (integers) that store the number of columns and cells.

*div_ i= distance AB/“distance between the tested points”

*div_ j= distance AD/“distance between the tested points”

SPOT then determines the color for each vertex of the cells which is filled by Quads surfaces. Java 3D interpolates the color of each pixel of the surface when the vertices of a mesh are different. As a result, the surface drawn by SPOT is a perfect color gradient.

Because the cells of adjacent rows have common vertices, SPOT maximizes the processing speed by recording the value of their color. Those values can then be accessed when the cells of the next row are created.

>Prepare the loop that creates the quad shapes that fill the ABDC area.

*declare and initialize the arrays of colors that store the value of the vertices color.

[ ] "last_color_c"

[ ] "last_color_d"

*Declare the integers I and J that are the index of columns and rows.

i=0

j=0

>Start the loop

*when j< div_ j |

| * when i < div_i | Determine the coordinates of the vertex a' of the current cell

| * x.a'= x.A + (x.B – x.A) X i + (x.D – x.A) X j

| * y.a'= y.A + (y.B – x.A) X i + (y.D – y.A) X j

| * z.a'= z.A + (z.B – z.A) X i + (z.D – z.A) X j

| Determine the coordinates of the vertex b' of the current cell

| * x.b'= x.A + (x.B – x.A) X (i+1) + (x.D – x.A) X j

| * y.b'= y.A + (y.B – x.A) X (i+1) + (y.D – y.A) X j

| * z.b'= z.A + (z.B – z.A) X (i+1) + (z.D – z.A) X j

| Determine the coordinates of the vertex c' of the current cell

| * x.c'= x.A + (x.B – x.A) X (i+1) + (x.D – x.A) X (j+1)

| * y.c'= y.A + (y.B – x.A) X (i+1) + (y.D – y.A) X (j+1)

| * z.c'= z.B + (z.B – z.A) X (i+1) + (z.D – z.A) X (j+1)

| Determine the coordinates of the vertex d' of the current cell

| * x.d'= x.A + (x.B – x.A) X i + (x.D – x.A) X (j+1)

| * y.d'= y.A + (y.B – x.A) X i + (y.D – y.A) X (j+1)

| * z.d'= z.B + (z.B – z.A) X i + (z.D – z.A) X (j+1)

|if the first row is being created:

| * if i=0 | Determine the color of the 4 vertices using "Light_count"

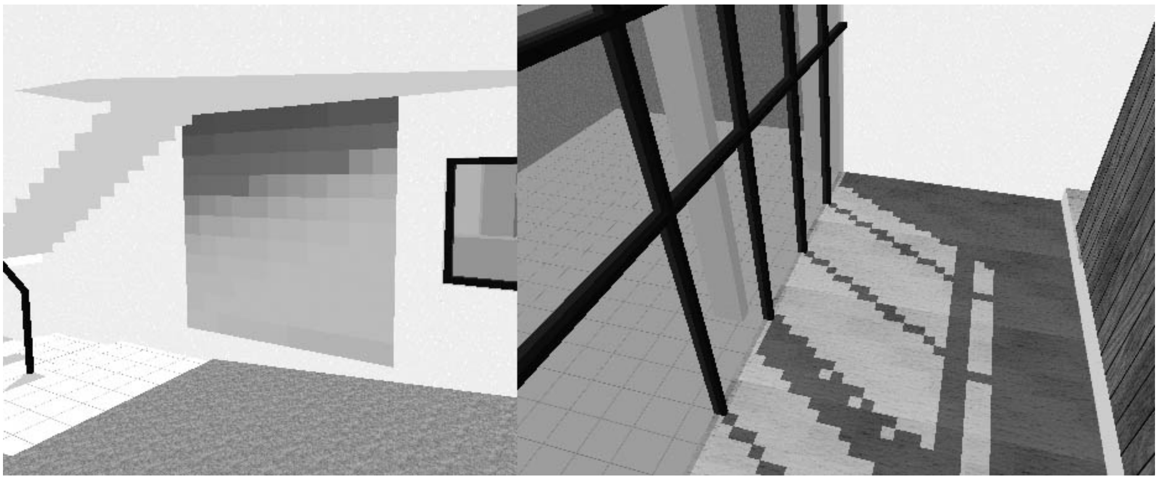
| * color_a' = light_count (a')

| * color_b' = light_count (b')

| * color_c' = light_count (c')
| * color_d' = light_count (d')
| Store the color of c' and d' at the index # i of the array
| * last_color_c'[i] = color_d'
| * last_color_d'[i] = color_d'
| * Create the a'b'c'd' quad array at the raw j and the column i.
|* i++
|* i++

| * else (i>0) | The color of c' and d' must be determined
| * color_a' = last_color_d'[i-1]
| * color_b' = last_color_c'[i-1]
| * color_c' = light_count (c')
| * color_d' = light_count (d')
| Store the color of c' and d' at the index # i of the array
| * last_color_c'[i] = color_d'
| * last_color_d'[i] = color_d'
| * Create the a'b'c'd' quad array at the raw j and the column i.
|* i++
|* i++

A variant of this loop fills each cell with a single color. Each cell becomes a type of pixel drawn in the 3D environment. This is used in the following two cases.

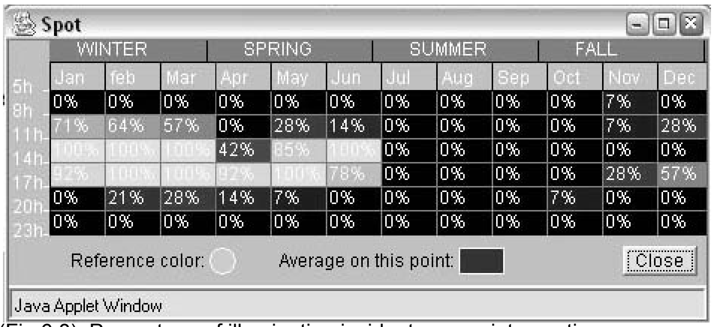


(Fig 6.7): Surface rendered by the "*Time interval*" module of SPOT with the parameter "*no interpolation between the tested points*" (Left). Surface rendered by the "*Spot static*" module (right).

The advanced parameter of SPOT's "*no interpolation between the tested points*" enables one to render the area with cells that have a uniform color. SPOT determines the value of cell's color using the "*light_count*" function at the center of the cell. This makes it possible to see the cell's limits and understand where the tests were performed. This parameter also maximizes the processing speed.

The variant of this loop is also used when "*Spot static*" is launched. This module of SPOT renders the shading effect of a selected area at a given moment. The cells are also filled with a 3Dshape that has a single color. This color is determined with the "fetchSunlight" function. A cell is black if the function returns "false" (the point is shaded) whereas it is yellow if "true" is returned.

6.4 The graphic calendar outputs.



WINTER			SPRING			SUMMER			FALL		
Jan	feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	7%	0%
71%	64%	57%	0%	28%	14%	0%	0%	0%	0%	7%	28%
100%	100%	100%	42%	85%	100%	0%	0%	0%	0%	0%	0%
92%	100%	100%	92%	100%	78%	0%	0%	0%	0%	28%	57%
0%	21%	28%	14%	7%	0%	0%	0%	0%	7%	0%	0%
0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

(Fig 6.8): Percentage of illumination incident on a point over time.

SPOT enables designers to visualize the temporal information of light distribution over time for a given point. For each point clicked on the 3D model, SPOT generates a calendar diagram of a chart where the X and Y axis represent the months of the year and the time of the day. The color of each cell of the calendar is the result of the calculation of the light amount reaching this specific point.

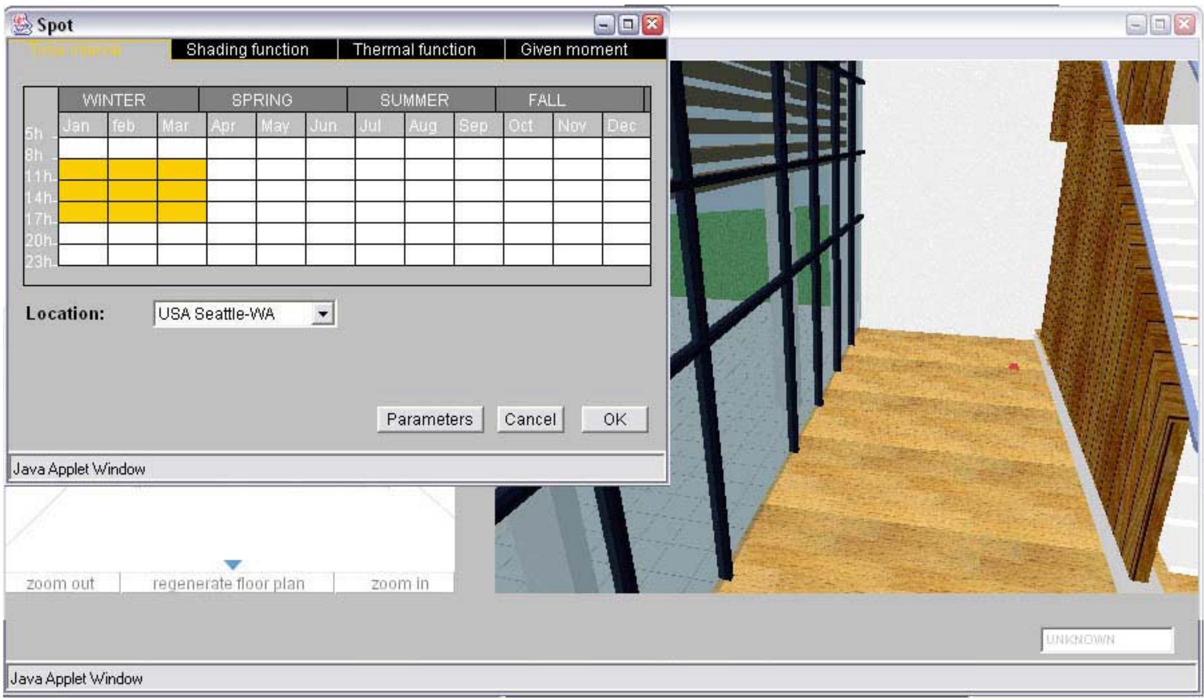
The display in 2D graphics is drawn by a loop that is similar to the one that shows the sunlight distribution in the 3D space. A date object that stores a date and a time is recursively incremented. SPOT counts how many hours the selected point in the 3D space will be lit during a period of time that corresponds to a cell. Once the loop has gone through the time interval of a cell, it can be drawn with the color that corresponds to the value of the counter.

7- SCENARIO

SPOT is based on the premise that enabling designers to visualize direct sunlight penetration in their early sketches will improve their design process. Since the system is now built, user testing would verify the validity of this hypothesis and would be a future step for this research project. However, the following scenario demonstrates how SPOT would be useful in the early stages of the design process.

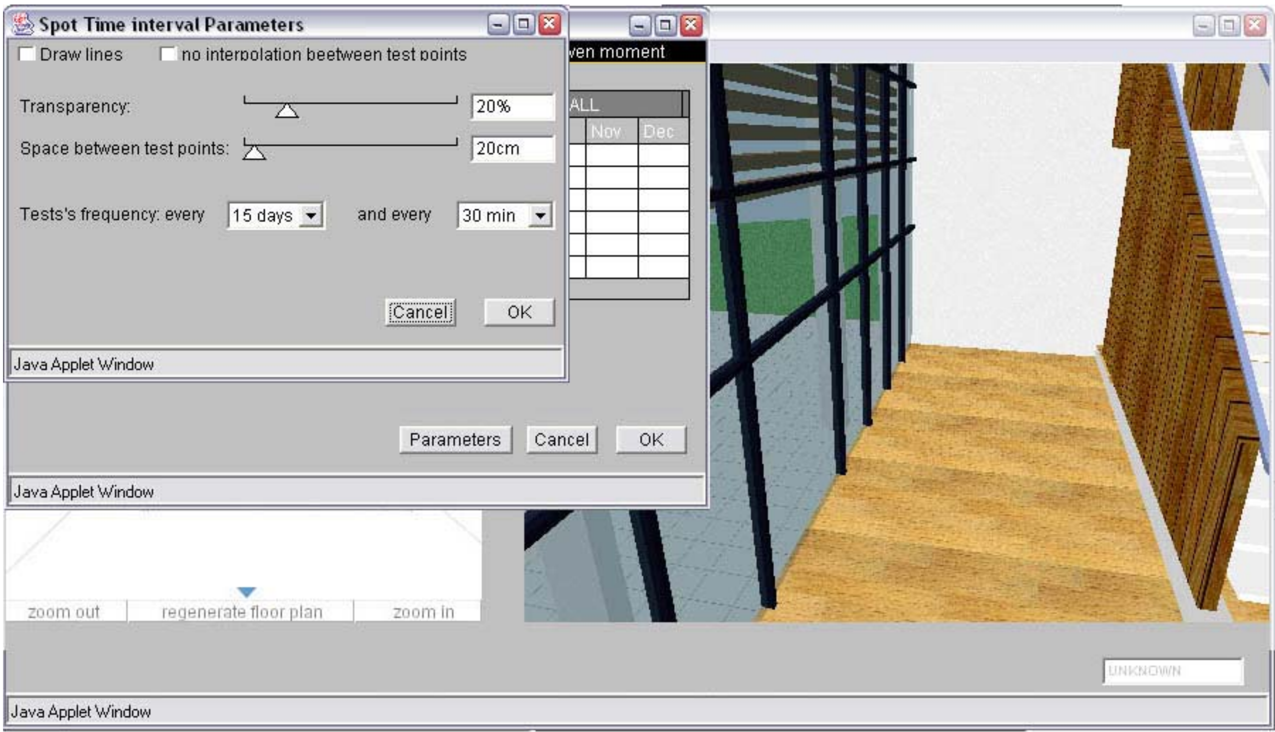
Jane, an architect, is designing a building with south facing windows. She would like to assess the sunlighting performances of her current design. SPOT enables a quick preview of the amount of direct sunlight projection over time in a navigable 3D web environment. Using early 3D model generation techniques, Jane has quick 3D models available at the beginning of the research. She exports the generated models into the VRML format so they can be read by SPOT.

Jane uses two distinct but complementary modules of SPOT: (1) “Time Projection” and (2) “Navigable Animation.”



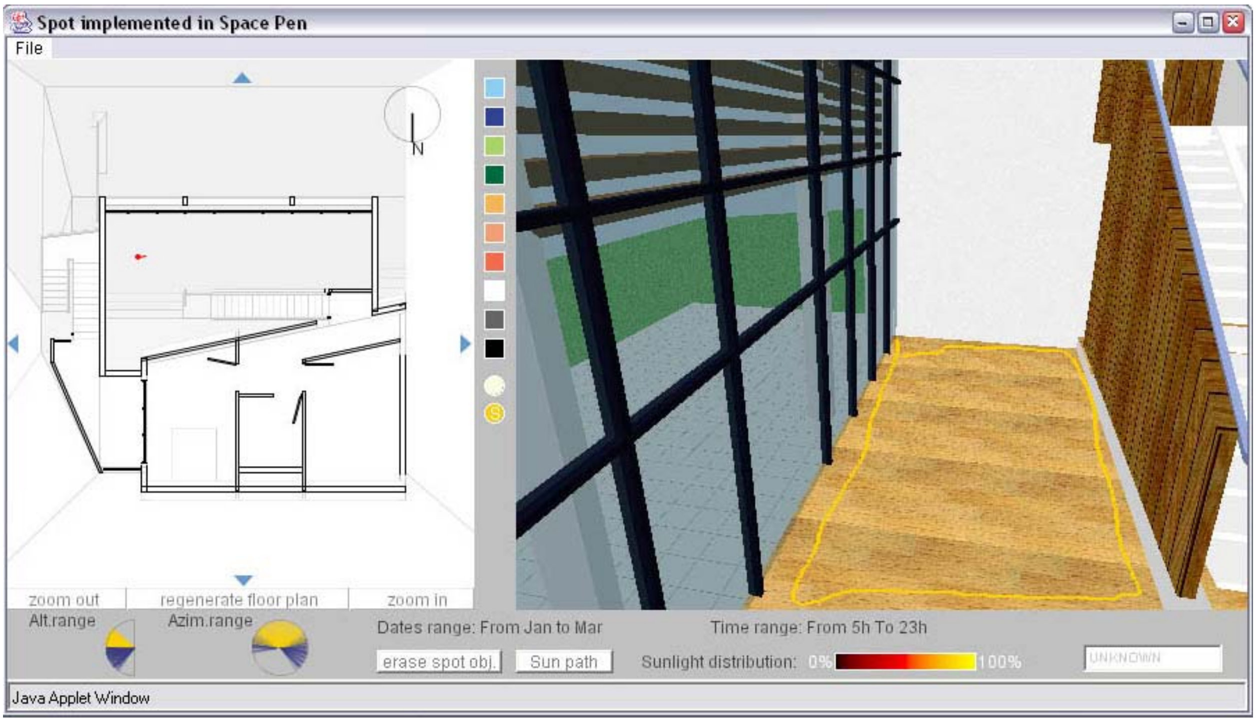
(Fig 7.1): The selection of a period of time on a graphic calendar chart.

To use the "Time Projection" module of SPOT, Jane selects a period of time (e.g. 8AM-5PM during January-March) by dragging the mouse over a graphic calendar chart (fig.7.1). She chooses geographic locations from those pre-listed or the coordinates can be entered manually for non-listed locations. To orient the model, a north direction input window pops up when she clicks on the north direction compass on the floor plan.



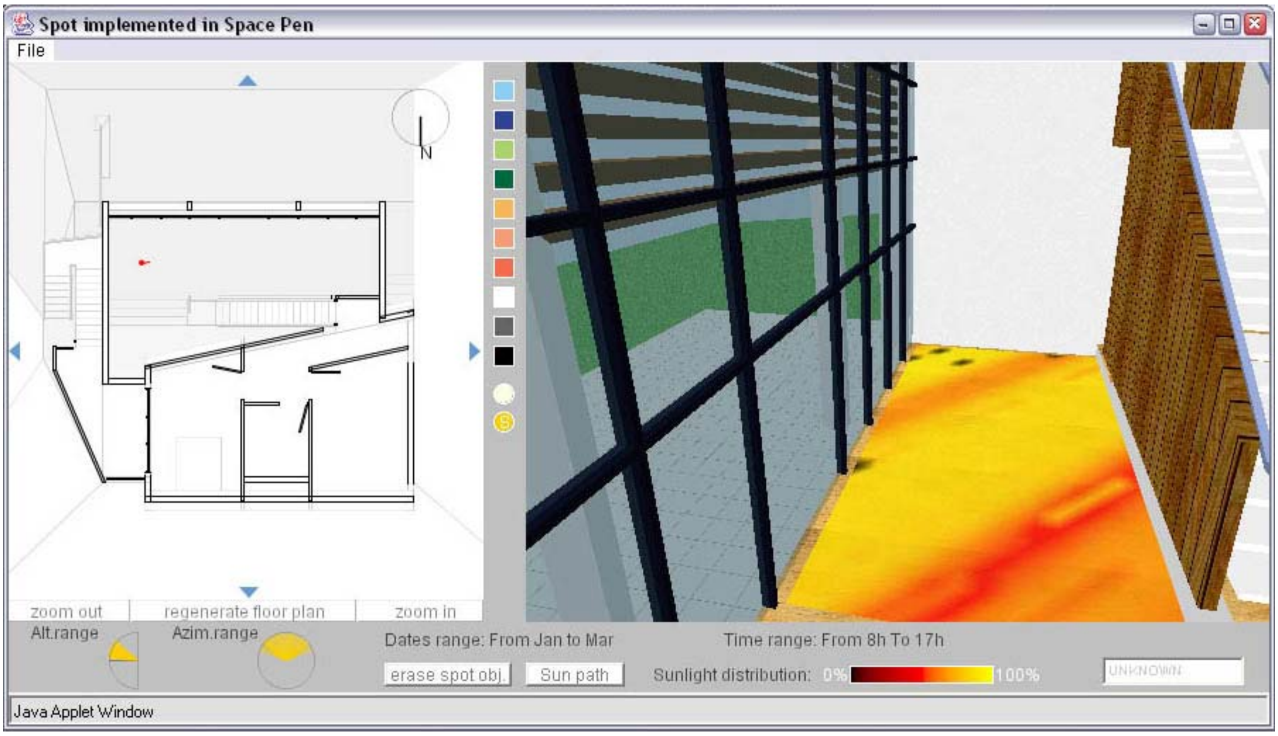
(Fig 7.2): The advanced input window of the “time interval” module of SPOT.

Jane can adjust several parameters in the "*advanced inputs*" window. These include the test's frequency and the distance between the test points. Other parameters such as the transparency of the annotations generated by SPOT can be changed as well (fig.7.2).



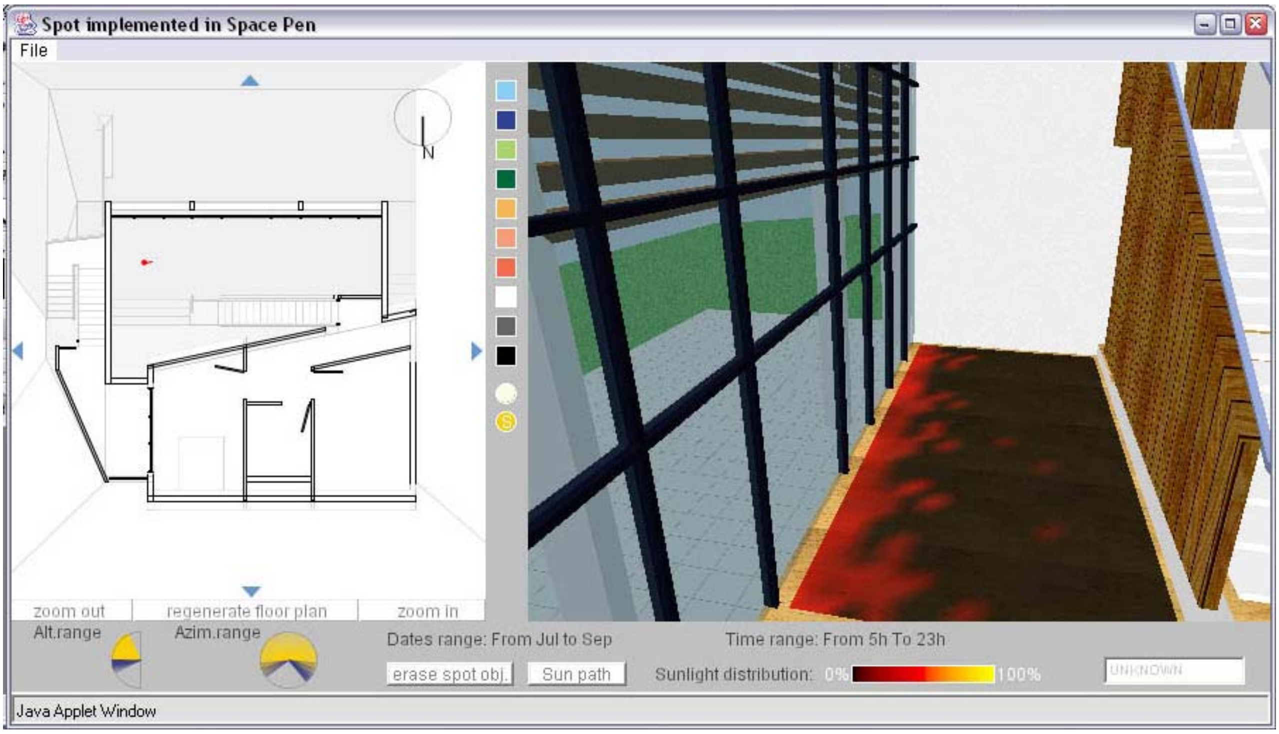
(Fig 7.3): Selecting an area for simulation by drawing on the 3D model.

By pressing the “OK” button in the input window, Jane launches the "*Time interval*". The solar azimuth and altitude range over the selected period of time and a color code are displayed in the bottom panel (fig.7.3). The “Time Projection” module consists of two views of the data: (a) the temporal lighting distribution displayed on a selected surface in the navigable 3D world; (b) and the temporal distribution of light represented in graphic calendars. To indicate the area for visualization of sunlight distribution, Jane sketches a boundary shape on the 3D model (fig.7.3).



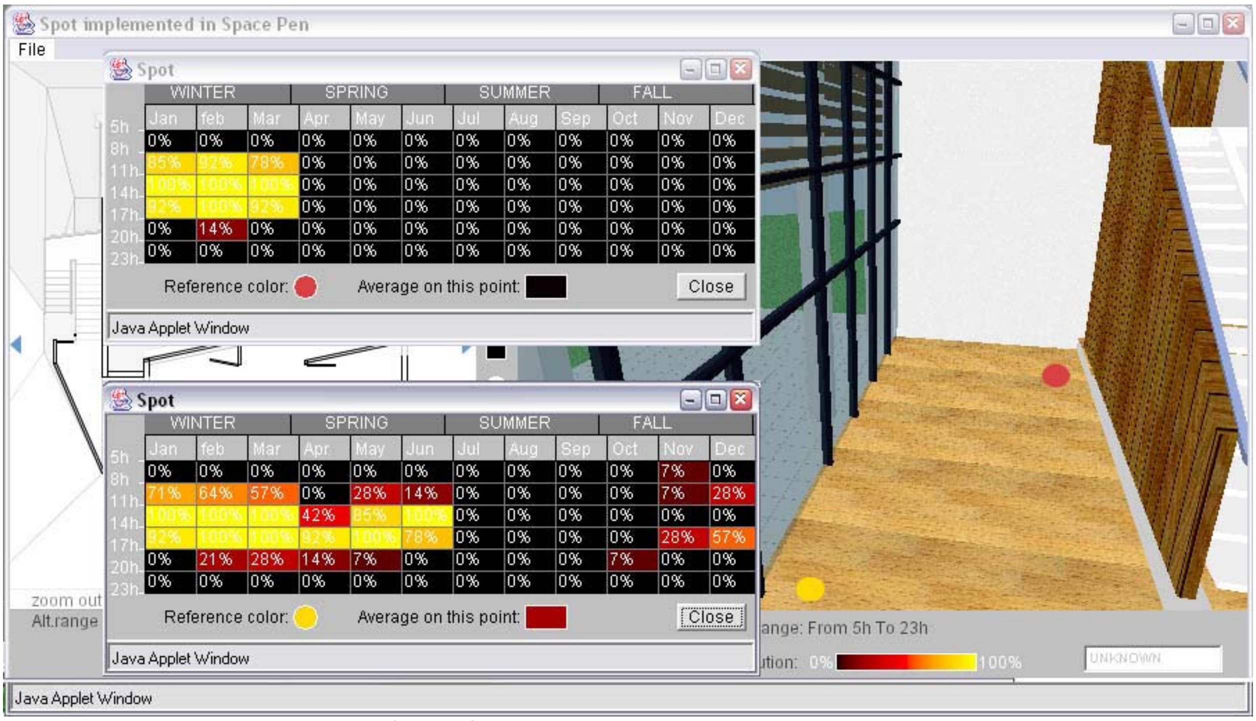
(Fig 7.4): The rendering.

SPOT then renders the selected area with colors of varied gradients. The color of each pixel of the surface corresponds to its percentage of illumination during the selected period of time (fig.7.4).



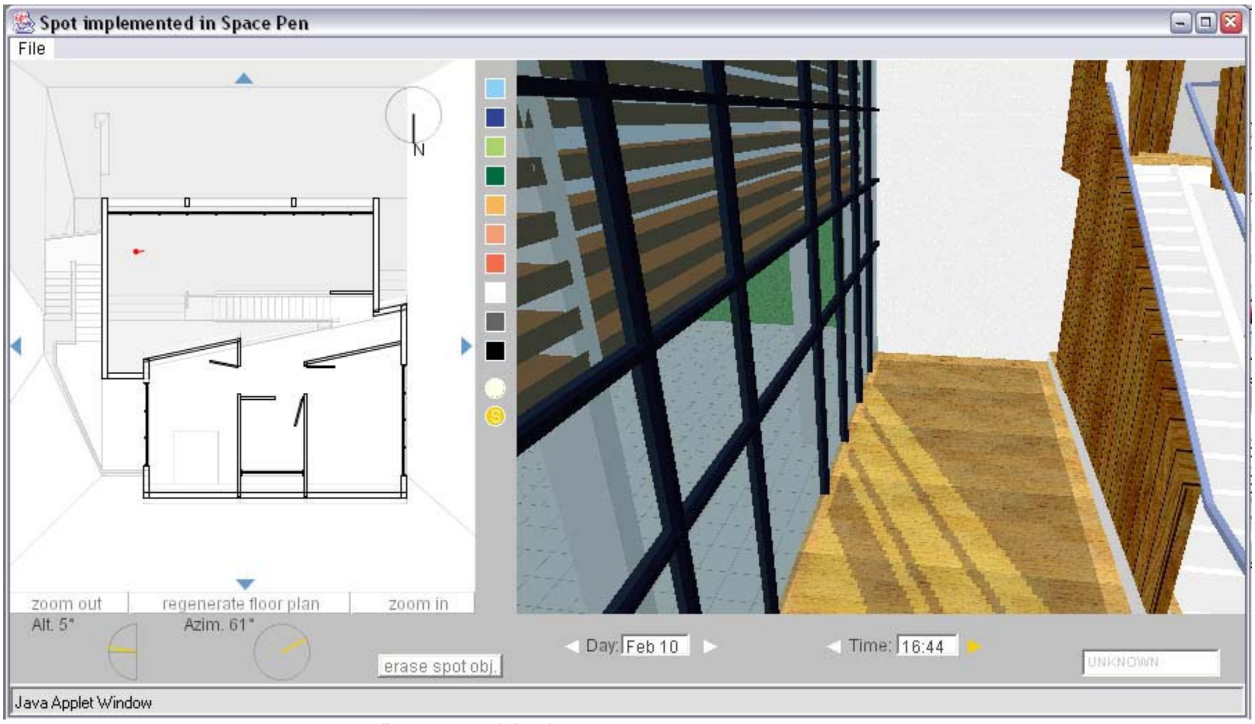
(Fig 7.5): The spatial distribution of the direct sunlight during the summer.

Jane now performs the same test for another period of time; all day during the summer. In our example, except from the areas located near the window, the living room will be protected from the sun during the summer (fig.7.5).



(Fig 7.6): The temporal distribution of light of a given point.

Jane clicks on the floor to inquire the temporal distribution of light in a year of that particular point. For each point clicked on the model, SPOT generates a calendar chart to show the sunlight amount against the months of the year (X axis) and the time of the day (Y axis) (fig.7.6).



(Fig 7.6): The “Navigable Animation” module of SPOT.

Using the “Navigable Animation” module of SPOT, Jane navigates through time by clicking the forward and backward buttons to change the time and date. SPOT rapidly renders the shadow cast effect on the selected area for the given date and time. (fig.7.6) Jane moves her viewpoints in the 3D environment to examine the lighting effect from different locations. The “Time Projection” analysis and the “Navigable Animation” both illustrate that a large amount of sunlight would enter the space in the winter. To protect some areas, she changes her design and uses SPOT again to assess her modifications.

8-CONCLUSION

The purpose of SPOT is to help users gain a deeper understanding of the sun's impact on their project. This research was based on the premise that considering light issues from the beginning of the design process has a significant impact on the project's final shape and its sunlight performances. SPOT is a software program that is now available for designers. So far, no user testing has been performed to determine if it is a tool that improves the design process. To analyze the significance of this prototype, the first step would consist of formulating test criteria to see if SPOT is effective.

We would like to assess the relevance of the visualization method provided by SPOT. Does it improve the design process? What impact such program could have on a project? What is the sunlight performance of projects designed using SPOT?

To answer these questions, SPOT should be used to design a real project. We based this research on the premise that 3D models of early conceptual projects could be generated using recent techniques. However, those technologies are still under development. It would be therefore difficult to utilize SPOT today. However, my final project for my architectural studies includes assessing the relevance of SPOT. For this study, I will both design and analyze how SPOT influences my design decisions. This should illustrate how SPOT could be improved. However, we are currently working on making some improvements.

SPOT is implemented on top of the 3D geometry-editing engine SpacePen that has many functions and features to support collaboration over the web. As a result, designers can make simple modifications on the model's geometry using SpacePen's 3D editing functions. Daylight performances on these modifications can then be assessed in SPOT. Future work would include functions for generating elaborated design elements such as shading devices. SpacePen also provides annotation functions to document design rationale. Other Future work includes improving SPOT to facilitate special annotation of the lighting simulation result for remote collaboration over the web.

9- ANNOTATED BIBLIOGRAPHY

[2.1] Heschong Mahone Group
"SKY LIGHTING AND RETAIL SALES-An Investigation into the Relation Between Daylighting and Human Performance." (Condensed report) Submitted to George Loisos – Pacific Gas and Electric Company, August 1999.
This study conducted by Pacific Gas and Electric examines the correlation between occupant productivity and exposure to daylight within retail and school buildings. The study found that the sales were increased (average 40%) and bettered the student's performances at tests (typically 10-20%) when working in spaces exposed to daylight.
Efficient daylight design also results of energy saving (by 30-60% in commercial and institutional buildings).
Retail sales analysis included data from 108 stores. The study concludes that a skylit store would be likely to have an average of 40% higher sales than similar stores without skylights, with an expected range of increase between 31-49%.

[2.2]
Gregg D.Ander, AIA
"DAYLIGHTING-PERFORMANCE AND DESIGN-ANDER."
Publisher: John Wiley & Sons; 2nd edition, December 1997.
The purpose of this book is to show architects how to incorporate natural light with their design. It provides the reader with complete information about the fundamentals of dayliging. This authoritative resource is also a catalogue of daylighting strategies.

[2.3] by: Warren E. Hathaway, John A. Hargreaves
Gordon W. Thompson and Dennis Novitsky
"A Study into the Effects of Types of Light on Children – A Case of Daylight
Robbery"
Distributed by Policy and Planning, Branch Planning and Information Services Division, February 1992.

The Department of Education of Alberta, Canada, carried out a two years study in an elementary school to analyze the influence of full spectrum light on the students. The physicians, educators, social workers, nutritionists and dentists that conducted the study found that students under full spectrum light with trace ultraviolet: learned faster, tested higher, grew faster, had 1/3 fewer absences due to illness, had 2/3rds fewer cavities than expected.

[2.4]
Gila Lindsley, Ph.D., A.C.P.
"Seasonal Affective Disorder (SAD): About light, depression & melatonin"

New Technology Publishing, Inc -
http://world.std.com/~halberst/contrib/sad.html
This article describes the symptoms, cause and treatments of the Seasonal Affective Disorder (SAD); the blues of wintertime. Some people have light sensitivity that is severe enough to provoke sadness, anxiety, irritability, and violence. Those symptoms are typically observed during the winter under our latitudes, when the daylight hours decrease.

[3.1]
HOUPERT Sylvain, SIRET Daniel, MARENNE Christian.
“Les ambiguïtés d'un dispositif solaire de référence : la loggia de la « maison radieuse » de Le Corbusier”
In CISBAT'99 Energie solaire et bâtiment, Lausanne, 22-23 septembre 1999. Lausanne : Ecole Polytechnique Fédérale de Lausanne, 1999, pp. 63- 68
The sun path is higher during the summer. Therefore, Le Corbusier explained facades that contain openings can be protected from beam light with overhangs such as balconies or loggias. His solar principle became a reference. However, this paper reveals Le Corbusier failed to implement it in the “Unites d’Habitation de Marseilles .” His principle works on South facades only but the “Unites d’Habitation ” was oriented East-West.

[3.2]
S. Robert Hastings
Lecture: “THE EVOLUTION OF SOLAR ARCHITECTURE”
Lecture given at the "6th Summer School for Solar Energy 2000" at Klagenfurt, Austria, 2000.
This essay travels through the history of solar architecture from over two millennia ago to the present with a focus on residential construction. It explains the concept of using sun in architecture can be found far back in the history. According to S. Robert Hastings, making use of solar energy within buildings was already a know art. Egyptians perfectly mastered the sun path to dramatize the atmosphere of the temples. Socrates had the idea to use the seasonal variation of the sun path to maximize sun penetration during the winter only.

[3.3]
Marc Schiler, Assoc. Prof.
“TOWARD A DEFINITION OF GLARE: Can Qualitative Issues Be Quantified?”
In 2° CONFERENCE ARCC-AEEA - 2d EAAE-ARCC CONFERENCE PARIS, Ecole des Beaux-Arts, July 4-7 2000

A method for predicting the factors that creates glare consisted of scanning a series of images and analyzing their pixel's variations of intensity. The data are displayed in a histogram that enlightens the number of pixels that have a given intensity. Different intensities are represented on the X axis, the distributions which could be associated with discomfort glare corresponds to a strong contrast.
This paper also assesses the current technologies implemented to design sunlight efficient buildings.

[3.4]
Rebecca Ellison
"THE EFFECTS OF DAYLIGHT"
In the "Building Conservation Directory 2000"
Publisher: Cathedral Communications Limited, 2000.
Rebecca Ellison, conservator at International Fine Art Conservation Studios in Bristol explains that daylight might damage vulnerable objects, especially those made of organic materials. Solar radiations excite electrons, causing material to be damaged and colors to fade. Heat fluctuation that engenders expansion and contraction of objects is responsible for their deterioration too. She underlines the necessity of controlling exposure levels to daylight, expressly to direct sunlight that is the most intense.

[3.5]
Andrew M
THE BIBLIOTHEQUE NATIONNAL
In Bonjour Paris, December 1998.

This articles divulges the boobs the architect Dominique Perreault committed when designing the France's Bibliothèque National (National Library). *"Books are shy, retiring types who aren't that keen on sunlight*;" nevertheless, the books were to be exposed to daylight. Perreault's concept was to store them in four transparent towers. He did not achieved his goal, "*because protective wooden shutters have had to be installed on them to protect the sun-shy books*".

[4.1]
Design Protocol Data and Novel Design Decisions
By Omer Akin and Chengtah Lin
In Analysing Design Activity, The Delft Protocols Workshop, conference proceedings, 1994.
Research in AI is based on the premise that we could artificially reproduce the human design thinking. The purpose of recent scientific investigation is to have a better understanding of the human design reasoning.
The design protocol analysis consists of studying the cognitive strategies designers employ. Each subject is instructed to design something while an

experimenter analyzes the produced outputs. Designers employ a variety of visual or graphical mediums of work to treat different kinds of information that is sequentially processed. In this research, the activities that are observed while a designer is working are the following: thinking, writing, examining, and listening. The subject was instructed to think aloud to enable the experimenter to interpret his activities.

[4.2]
Charles M. Eastman
"On the Analysis of Intuitive Design Processes"
In Emergining Methods in Environmental Design and Planning, Gary T Moore (ed), MIT Press, 1968.

The purpose of this research is also a protocol analysis based investigation into the designer's reasoning. However, this article points out the limitations of the use of the think aloud method when one wants to access the designer's mind. Nevertheless, even if we do not have accurate tools today, the results of those experiences are remarkable.
Thanks to the protocol analysis, quite a few patterns of the strategies designers employ while designing were documented.

[4.3]
Ellen Y. Do
"VR Sketchpad - Create Instant 3D Worlds by Sketching on a Transparent Window"
In CAAD Futures 2001 Eindhoven, pp161-172, 2001.

VR SketchPad"[3] is a software application for 3D models generation from early sketches. The Author Ellen Do describes her project as a "pen-based interface" software that features "3D world creation using freehand drawing [, which] depends on users' traditional understanding of a floor plan representation."

[4.4]
Cohen J., Hughes J., Zeleznik R
"Harold: a world made of drawings"
In ACM Press SIGGRAPH, pp 83-90, 2000.

Harold" [4] investigates another approach that consists of drawing directly in an interactively navigable virtual 3Dworld. It is tedious to draw directly in perspective. To address this issue, Harold features a "billboard" functionality that consists of a virtual screen used as a support to draw on it.

[4.5]
Thomas Jung., Mark D. Gross M., Ellen Y. Do
"Space Pen: Annotation and Sketching on 3D Models on the Internet"

In CAAD Futures 2001 Eindhoven, pp 257-270, 2001.

Space Pen is a program that supports annotating on 3D models to improve collaborative design. The interface consists of two windows, one containing a floor plan window generated from the imported model, a second containing the representation of a navigable 3D scene (the user can interactively walk trough the 3D environment using the arrow keys). To leave comments, a clicked point in the 3D environment is marked with a virtual post it, at the same time, a window pops up to store textual information contained in the Post it. The user also can draw lines and polygons in the 3D universe.

[5.1]
Richard Hollasch
"Four-Space Visualization of 4D Objects"
August 1991

The difficulty of visualizing multiple dimensional spaces lies in the fact that "there are no solid paradigms for three-spaces creatures such as ourselves. This difficulty is best understood imagining the plight of two-space creatures who try to comprehend our three-space world."

[5.2]
Roberts A., and Marsh A.
"Ecotect: Environmental Prediction in Architectural Education"
In eCAADe 2001, Helsinki, pp 342-347, August 2001.

This paper assesses the use of the software "Ecotect" into teaching. "Ecotect" is a program aimed primarily at architects to help them evaluate their ecological strategies. Intended for use during the earliest design phases, it integrates a simple 3D modeler that enables the user intuitively describe the shape and materials of their project. The software features several functions aimed at predicting various environmental questions; these include solar reflection; sun penetration; overshadowing; and other acoustic and thermal issues.
Used during an environmental course at the Welsh School of architecture, "Ecotect" was proved to enable students to evaluate they design and take them to a deeper understanding of environmental related issues.

[5.3]
Glaser D., and Hearst M.
"Space Series: A focus+context technique for displaying spatial and temporal data"
In IEEE Symposium on Information Visualization '99, Late Breaking Hot Topics, San Francisco, Oct. 1999.

To display multiple dimensional data on a single 2D display, "Space Series" uses a focus-plus-context technique. The illumination levels in a "workplane" are generated for different times or dates in a 2D display where the X and Y axis respectively represent the annual and diurnal cycle of the sun path. Initially, the graphic displays an overview of the data that is the average light levels into the whole space over time. The user can then focus on a given day and a time interval. The program will then reorganize the plot creating four types of regions: the fractional hour focus is a slide that shows the light levels in the space at different times, the day focus shows is another slide that displays the same information for a given day, and the context is still visible in the background. In addition, the focus intersection that is the intersection between a horizontal focus and a vertical one correspond to a given moment.