Visualizing Building Occupancy Pattern on Campus

Beng-Kiang Tan
Harvard University, USA
National University of Singapore, Singapore

Abstract
This paper addresses the problem of information opacity that planners and university administrators have when they have multiple sets of data that are not interconnected and how these data can be visualized. The visualization of building occupancy pattern on campus is used as an example to illustrate how this general problem can be addressed through a database driven effective visualization that supports decision-making. This paper proposes a solution using web-based 3D Virtual Reality Modeling Language (VRML) animation dynamically generated from a database and describes a prototype in progress. The prototype displays a broad overview of building occupancy patterns across campus through 3D animation of occupancy over time. From the overview, users can navigate further to find out the details of occupancy throughout the day for specific buildings on campus.

Keywords
Visualization, VRML, Animation, Campus Population, Information Visualization
1 Introduction
In a university environment, various departments are each in charge of different sets of data. Users are often presented with many sets of data whose interconnectivity is not obvious. Typical data visualization techniques are 1D and 2D visual structures with information positioned on orthogonal axes. 1D visual structures are typically used for timelines while 2D visual structures are typically used for chart and geographic data (Card, MacKinlay and Shneiderman, 1999). Examples are tables, graphs, bar charts and scatterplots which are all 2D static representations. Tables are the best way to show exact numerical values but are preferable to graphics only for many small data sets (Tufte, 1983). The most common way of representing building occupancy data, course schedules are by tables (rows and columns) of numeric and text data or by 2D static graphs and charts. The shortcomings are that rows and columns of numbers and text do not draw upon human perceptual and cognitive strengths and it cannot show all the building occupancies over say twelve hours all in one screen display. For eighteen buildings, there would need to be eighteen tables.

With 3D graphical representations, users can see more information at one glance and comprehend them better. Animation can be used to enhance the user’s ability to keep track of changes of view or visualization and mapping time data into space allows comparison between two points in time (Card, MacKinlay and Shneiderman, 1999). Good visualization also reveals patterns to the users which are not quickly discernable from a table of numbers and text, for example, animations are used for applications in capital markets (Wright, 1995) and logistic data (Chuah et al., 1995).

Visual information can be processed by our perceptual system in two different ways, controlled processing and automatic processing. Controlled processing, like reading, is detailed, serial, low capacity, slow, able to be inhibited and conscious. Automatic processing is parallel, can be processed nonfoveally, has high capacity, is fast, cannot be inhibited, is independent of load, unconscious, and characterized by targets “popping out” during search (Card, MacKinlay and Shneiderman, 1999). Perception literature tells us that graphical properties such as width, size, color, length, intensity, motion can support automatic visual processing (Healey, Booth and Enns 1995). In coding techniques, using features such as color and size that can be automatically processed will aid search and pattern detection. There are many examples of effective coding techniques in visual representations from graphic design to maps (Tufte, 1990). In this paper, coding techniques are employed in 3D animation and 2D web-based graphical display to present building occupancy information accessed from a database.

The paper is organized as follows. Section 2 outlines the objectives of the project. Section 3 describes scope of the project. Section 4 describes the solution for the prototype, and Section 5 offers conclusions of lessons learned and future work to be done.

2 Objectives
The audience of this visualization are campus planners, administrators, faculty, staff and university students. The objective is to allow interested parties to have a good overview of building occupancy patterns on campus through 3D animation of occupancy over time. Users will be able to find out the occupancy patterns of students in campus buildings throughout the day. Through the 3D visualization, one can see where the concentrations and fluctuations of student population are at one glance instead of referring to tables or a series of 2D charts. This will be useful for many of the following purposes:

- For a planner to decide where to locate new facilities; to understand the pattern of occupancy for deciding landuse
- For an administrator to know which buildings are under utilized comparing occupancy with maximum capacity
- For an activity organizer to decide in which building to put up notices or booths for maximum exposure (i.e. if one has only one day to distribute flyers, one could plan where one should be at what time to capture maximum audience)
- For emergency situations like a fire, one could immediately tell how many people were suppose to be in the building

The 3D visualization provides an overview and an organizing structure from which users can navigate further to retrieve details.
3 Scope
For purpose of the prototype, the number of buildings was restricted to a central part of the campus. Only buildings occupied by undergraduates for lessons and for residence were included. As the purpose is to develop a prototype as proof of concept, the accuracy of building occupancy data was not crucial at this stage. Occupancy of each building by time was taken from course schedule and the enrolment of students in each course. The maximum capacity of each building and number of students living in the dormitories were obtained from the relevant university departments. As all students in dormitories have meal plans, it was easy to estimate the number of students returning to the dormitories during meal times. These data were used as a basis to design the database fields of the prototype. Most of the major features of the prototype have been completed at the writing of this paper.

4 Solution
The solution is a web-based 3D Virtual Reality Modeling Language (VRML) model generated dynamically from a database, supplemented by 2D graphical representations. The model is viewed through a web browser with Cosmos Player plugin. It can be viewed anyway, anytime through the world wide web (WWW). The database contains information on building number, room number, course number, course name, start time and end time of course and number of students taking the course, for example, Literature L103 takes place on Mondays from 8am to 10 am at Building 2, room 110 with 150 students enrolled (Figure 1).

In a fully implemented system, the database can be linked to relevant departments that have different parts of the information which can then be automatically uploaded and updated. For example, the various departments are the source for course schedules, the Registrar Office is the source of student enrolment numbers and the campus police department is the source for maximum capacity of each building. PHP, a server-side HTML embedded scripting language, is used in the system. A PHP script queries the database, totals up the number of students in each building at each hour and generates VRML code that draws the model and animation to be displayed in a web browser (Figure 2).

The VRML animation shows the number of students for each hour in each building throughout the day. This replaces the conventional approach of showing rows and columns of time and numbers in many tables or bar charts. The same data is now displayed in just one screen. The base plane of the model is a map with campus buildings and their names. This provides an underlying structure on which to display the abstract information. The map is a clear and widely understood way to show physical distance between buildings.

Some of the coding techniques used here are position, color, hue, shape, transparency and motion. A number of key elements used in the animation are:

- Objects – Graphical geometrical objects that represent data and transforms as the data changes.
- Marker – Graphical geometrical object that is permanent in position and size to be used a reference to make comparisons. Important textual information is displayed next to the object.

Figure 1. Database Fields and Records. Figure 2. Components of the Program.
- Brushing - Users can place a cursor over any object and retrieve numerical and textual information behind it.
- Viewports – Users can zoom and manipulate the model in all axes using the browser. There are preset fixed points of views such as top, left, right, etc to present the display such that heights can be compared in 2D views too.

The peak time and peak number of students for each building are displayed at the tip of the 3D model. The mapping of the data and visual variables are as follows:

- Location of building – map of campus (x,y)
- Building Type (Dormitory or Teaching) – color of cone. Green indicates teaching and yellow indicates dormitory.
- Peak number of students – cone changes to red
- Number of students in building – height of cone (z)
- Maximum capacity of building – light green rectangular box with transparency. This is a marker.
- Time of the day – changing hues of blue color in the background

Each building is represented by a cone. Color differentiates the usage of buildings (green for teaching and yellow for a dormitory). The cone animates vertically according to the number of students in the building by each hour. At the hour when the number of students is peak for that day, the cone changes to red momentarily. A transparent rectangular box around the cone represents the maximum capacity of that building. With this as a marker, it is easy to read how much a building is under utilized throughout the day. The user can choose which day of the week to view from a control panel. When fully implemented, the user can also choose “freeze” frames of the desired time of the day to look at. The user can also select preset viewports for looking at the 3D models e.g. front, side, angle views (Figure 3).

It is very obvious from the animation where the lulls and peaks are in campus. For example, the dormitories have a sharp surge in the morning and evening when students leave for class and return for dinner respectively.

When fully implemented, the passage of time will be represented by the changing hues of blue color in the background (Figures 4 to 6). Light blue represents early morning and dark blue represents morning.
night time. The changes in background draw on human ability of peripheral vision to detect subtle changes.

Users can place a cursor over any object in the VRML animation and retrieve numerical and textual information behind it. A separate window will pop up showing charts of the detail occupancy by the hour within each building (Figure 7).

When the cursor rolls over one of the cone on the chart, the cone changes color and details of course time will pop up next to the chart (Figure 8).

The PHP script uses external prototype definition for the cone, the box representing maximum occupancy, the text for the peak time and occupancy. This strategy is used to speed up the generation of graphics, for ease of debugging and minimize usage of resources. The contents in the PHP script are:

- ExternProto for Cone (with animation)
- ExternProto for Box
- Extern Proto for Text
- Query database and pass variables needed into VRML code.
- Generate VRML code.

Each ExternProto of the script was tested separately before assembling into the main script.

5 Conclusion and Future Work
5.1 Lessons Learned
This prototype was showed to many students, a few faculty and two campus planners. Most of them were captivated by the VRML animation and felt that it is a more effective way of representation. One planner, however, was concern
whether a layman will misread the cone as the actual form of the building. It is a valid concern and more reactions to the shapes will be sorted before deciding on the most appropriate choice of shape. Structuring the script into modular parts in which objects were defined as ExternProto was very useful in debugging and making changes to the shapes.

5.2 Future Work
The next stage is to complete the interactive links from the VRML model to detail information and charts of each building, background transition, improve on the navigation interface design and include filters that allow viewing of selected objects, for example, users can select to view dormitories only.

5.3 Conclusion
This prototype was able to demonstrate an effective way to unify data from different sources and present it with more clarity. The display is able to present more data in a small space, to give a broad overview for comparing occupancy, and then allows users to navigate further to obtain details. Animation is a powerful means to display patterns to the users. The prototype meets its objective of providing a good overview of building occupancy patterns on campus. It can be a basis to build applications for other types of information associated with buildings where comparison across buildings is important.

Acknowledgements
I would like to thank Laurence Chiang and Hou Jun-Hao for their invaluable technical assistance and Professors Jeffrey Huang, Urs Hirschberg and David Rose for their constructive comments of the project.

References
