Visualizing Information Associated with Architectural Design Variations and Simulations

David Aurelio

Cambridge, Massachusetts, USA
BostonDave07@hotmail.com

Abstract. As cloud computing becomes more ubiquitous, it produces many advantages for users in the architectural domains, but also some challenges. These advantages include multiple building design variations, energy simulations, structural options, and generations of design options. A literature review was conducted to review information visualizations and the problems that occur with them. The findings from the literature review were comprehensive and provide visualization guidelines for those in architectural domains, as well as for those in numerous other domains.

Keywords: visualization, information, design, variation, simulation, architecture, presentation, navigation, data.

1 Introduction

Architectural information modeling includes the simulation and analysis of building design variations, such as energy simulations, structural analysis options, and multiple generations of building design variations.

With the advent of low-cost, ubiquitous cloud computing, these simulations could generate potentially hundreds of design variations that designers would then need to understand and analyze to make informed design decisions. One can assume that the number of design variations would increase as the cost of computers and computer time decrease. This research was pursued because users are expected to interact relatively soon with a large number of architectural visualizations. The four goals of this research therefore were to better understand:

- Why do users generate design variations and simulations?
- What decisions do they want to make with the variations and simulations?
- What makes a good or useful variation or simulation? and
- What problems emerge when users are faced with a large number of design variations and simulations?

Although the scope of this research focused on architecture, the results of this research could be used by many software developers and user interfaces designers across a wide range of domains and disciplines associated with the visualization of information.

A. Marcus (Ed.): DUXU/HCII 2013, Part IV, LNCS 8015, pp. 469–477, 2013.

2 Method

The research method consisted of a literature review. The literature review of relevant books and articles was conducted to compile visualization guidelines.

3 Results

Overall, the literature review revealed many findings that related to the parameters necessary to consider when designing visualizations. The findings are presented as guidelines stated in the imperative mood and sorted into eight subsections.

3.1 Background

Visualizing data is increasingly important as people generate and access large amounts of data. Data visualizations were originally hand-drawn, but now are typically computer generated and performed for either analysis or presentation.

To access these visualizations, a user interface needs to facilitate discovery and not increase complexity. In Kahn and Hornbaek [1] Shneiderman states, "The purpose of visualization is insight, not pictures." Visualizations can be judged on how well they assist users gaining insight and making decisions.

Visualizations should focus on decision-making and human perceptual abilities, and not on the mechanics of visualizations and color. Some questions for designers are, "How should we present an interface that defeats human perceptual bias? How can we support decision-making that is good?" Designers need tools and algorithms to help people see the differences among data visualizations.

When working with data to visualize, Steele and Iliinsky [2] offer five useful tips:

- Be prepared to spend a lot of time transforming data into a useful format.
- Automate as much as you can to facilitate future work.
- Think carefully about how you will represent time, such as through an animation's pacing and moving.
- Decide when good is good enough (i.e., get feedback and iterate often).
- Approach the problem like a journalist (i.e., untangle complexity and tell an objective story).

To develop a story, one needs patterns and relationships. The goal of visualization would be to understand those patterns and relationships, and facilitate comparisons.

One form of visualization is simulation. Simulation "describes a system in which unspecified emergent behavior can be observed" [3]. Simulations ask "what if?" questions concerning performance. "Using analysis and simulation techniques, it is possible to model and test the performance of a huge range of design options in a very short time, something that only a generation ago would have taken a lifetime's worth of observation and empirical experience to acquire" [3].

Simulations should ideally reveal the effect, contribution and interaction of each simulation parameter and allow discovery. Currently, many architects look at only one variable at a time in a simulation experiment, because they are usually unfamiliar with factorial experiments [4] involving multiple variables and their interactions.

3.2 Generation

Two of the research's goals were concerned with why people generate design variations and simulations, and what decisions they want to make with them. The research found that people generate design variations and simulations to:

- Solve a problem.
- Meet a target.
- Inform themselves about a hypothesis (e.g., does incident sunlight decrease energy costs?).
- Identify problem areas in a design.
- Determine the amount of a variable (e.g., structural deflection) in a design.
- Choose the best design while considering one or more variables.
- Justify a design and convince others that a design variation is the correct one.
- Determine the range of reasonable values for a variable.
- Visualize a database.
- Explore a design space and aesthetic choices.
- Determine if a simulation does or does not meet a project's or client's goals.
- Clarify the relationship among several interacting variables.
- Allow clients and upper management to view work and choose options.

While there are many reasons to generate design variations and simulations, there are also a few reasons why users do not generate them. One reason would be because the design contract does not include the time needed to generate them, and another reason would be because eventually the design becomes constrained and there are few design options.

After each designer considers whether to generate a design variation or simulation, the next step is to actually generate it. "The challenge for problem-solving interfaces is to support the rapid creation of loose sketches, the ability to modify them, and the ability to discard all or some of them. All this must be done with an interface so simple that it does not intrude on the visual thinking process" [5].

To generate a simulation, one user interface approach might be to provide manual sliders on the simulation variables and request a run. When generating simulations, users may want to change variable levels quickly and then recalculate quickly. If the change can be compartmentalized, then the generation time can be reduced because there may be no need to recalculate the entire model.

Including data concerning the building's occupants and their activities is important because "A simulation which oversimplifies human behavior is unlikely to yield an accurate prediction of a building's energy requirements" [6]. Once occupancy is known, personas associated with different occupancy levels can be used to customize and diversify occupant behavior as it relates to calibrating a model of building energy performance. Architects can use this model to "generate plausible interdependent schedules specifically tailored to their own projects" [7-8].

3.3 Presentation Method

Once a design variation or simulation is generated, the generation software needs to present the results or runs. For visual presentations, “Graphical elegance is often found in simplicity of design and complexity of data” [9].

Visual presentations encompass five dimensions: size, color, location, connection to data, and time.

Data Structure. The key guideline is to use a visualization format that fits the structure of the data. For data sets that are small, non-comparative, and highly labeled, use a table.

Number of Data Dimensions. For one-dimensional data, use one-dimension elements (e.g., height). If the items are arrayed in one dimension, plot all the items along one axis, and when the user hovers or selects an item, allow zooming in on that item’s annotation.

Reorder qualitative categories in a meaningful order such as decreasing frequency of occurrence. If the data are categorical, there are three good visualizations: tree-maps by Shneiderman, mosaic plots by Theus, and parallel sets by Kosara.

For two-dimensional data, use two-dimension elements (e.g., area). If there is a structure among options, make those structural differences obvious. A network diagram can show structure. Provide users with dynamic sliders to remove a network diagram’s weak connections and use a highlight feature that displays the details for one node or instance.

If the data are non-hierarchical, do not display them in an artificial hierarchy. If the qualitative categories are plotted on two axes, order the categories so that the chart trendline is roughly diagonal.

“Diagrams should be used to express structural relationships among program elements, whereas words should be used to express detailed procedural logic” [5]. For example, use a diagram to show an organization chart, and not a series of sentences.

If the data is geographical, display it geographically but feel free to distort or simplify the geography to highlight the meaning of the data. If using visual distortion to convey information, the distortion must take place on a well-known object or projection [10].

For data with three dimensions, possibly use a two-axis chart with the third variable depicted as the circle’s area, not its radius or diameter. For more than three-dimensional data, consider plotting the results in a star chart along with a goal circle. “Given their low data-density and failure to order numbers along a visual dimension, pie charts should never be used” [9].

Static Display of Time. If one of the data elements is time, it can be visualized by a chart, by several charts, or by a motion pathway or arrow. If time is plotted along a dimension, it should be on the horizontal x-axis.

“Time series displays are at their best for big data sets with real variability. Why waste the power of data graphics on simple linear changes” [9]? Take care that the time axis does not mask the true causal variable. "Add spatial dimensions to the design of the graphic, so that the data are moving over space (in two or three dimensions) as well as over time" [9]. For example, the Tufte graphic for the march of Napoleon's army in Russia depicts the number of soldiers in the spatial dimension as well as over time.

To best reveal the overall trend in a time series of graphs, use a data resolution that results in a platykurtic (i.e., flat or lumpy) distribution, not a leptokurtic (i.e., spiky) one.

Animated Display of Time. When there are many data visualizations or one wants to speed up a variable, present the visualizations in one user-controllable animation to highlight patterns. For learning purposes, animations are about the same as rich static diagrams, and they both are more effective than text. However, text is better for details and exploration.

A shift in color (e.g., from blue to red) is a typical parameter of a visualization to modify that indicates a continuum of values, but it might be more true to the data's structure to use one color and other visual techniques such as intensity, isocontour shapes, or photorealistic renderings on top of the designs.

3.4 Detail Level

Once the software presents the visualization, there is a question of how much detail to show in the default visualization. The general rule is, Don't just present data; provide meaningful focus, comparisons and context [5].

Let each graphic have multiple layers of detail and meaning. "Graphics can be designed to have at least three viewing depths: 1) an overall structure usually aggregated from an underlying microstructure, 2) the fine structure of the data, and 3) that which is behind the graphic" [9].

Ben Shneiderman's three-step mantra for interacting with information visualizations: "Overview first, zoom and filter, then details on demand." As a corollary, it might be acceptable to offer all three steps at the same time if the user interface is clear. Otherwise, just present an overview at first.

For amount of detail, don't be afraid to have extremely high data densities in a graphic; the mind can handle it. One simple way to increase density is to drastically reduce the overall size of the graphic so that many of them can be seen at once. This process results in small multiples, such as the sparklines feature used in Microsoft Excel.

Simulations of building performance may contain years of data. "Future building energy models may become so detailed that real time compression is required for simulation output data as well as input data" [11].

3.5 Navigation

The previous two subsections discussed how to present a visualization and how much detail to provide. But if there were multiple visualizations, the user would need to move or navigate through them. Users can navigate among visualizations spatially, structurally, or temporally.

Based on users' tasks, some users would prefer to look at one simulation at a time. Some of those users may have the individual difference of spatial ability to compare the visualizations in their minds. Other users may need to quickly flip between the simulations, or advance through them using a slider or autoadvance.

Visualizations should reveal the data aspects that people are not good at seeing. Differences among the simulations may be small and some users will have trouble being aware of the differences. This trouble could be a perceptual issue called "change blindness" due to a lack of attention. The users' awareness can be increased by the software's navigating through the simulations, highlighting the differences, or producing a heat map for the change between two simulations.

Some users, such as senior partners in architectural firms, are likely not familiar with the rotational viewing tools associated with simulations. Those users are usually provided with PDF files of rotated views so that they can navigate linearly in the PDF file through the views.

For comparison, a side-by-side arrangement of views by itself may not be enough. Users would benefit from software that compares simulations statistically. Although the detail level of multiple thumbnail images can increase the density of the visualization and are useful for design exploration, the thumbnail images are not particularly useful for analysis.

3.6 Analysis

Once users have navigated through multiple visualizations, they need to make sense of them all and select a subset for further analysis, refinement or iteration.

One analysis workflow would be to pick one variable and then verify if it meets the criterion and if all the other variables are acceptable. However with multiple visualizations, it would be easy for users to miss the best run or the most important variable affecting the simulations.

Instead of the user investigating one variable at a time, the software could use statistics to analyze the simulation results and show the few variables that drive the most variability of the results. The user can then have more freedom to vary the unimportant variables.

Factorial design of experiment is an experimental technique where multiple variables are analyzed at the same time [4]. Statistical analysis determines if each variable or their interactions have a significant effect on the results. However, factorial designs and statistics are generally unknown in the architectural fields, though some larger architectural firms might have simulation analysts or others familiar with statistical analyses.

Architects and designers are more likely to trust their eyes, so visualization is key for presenting statistical and optimization results. Optimization techniques do produce a single best answer, but it is difficult to include the paramount parameter of aesthetics in an architectural optimization model. Few clients would be willing to trade off aesthetics for the best performing (i.e., lowest energy cost) building.

With these optimization visualizations and "simulation results in hand, an architect is better able to predict the energy demand associated with various designs, and choose from among the more sustainable options" [7-8].

3.7 Annotation and Tagging

Annotation and tagging should be provided automatically by the software to provide meaningful information and facilitate decision-making by the user.

Annotation can begin with the names of the simulation files themselves. For their names, there should be an automatic, hierarchical naming method that indicates the file content, structure or ancestry. Design variations are usually generated in a hierarchy of genus (i.e., category) and species (i.e., subcategory). Users would benefit if the software kept track of and displayed this hierarchy. The users could use this hierarchy to help identify the variations for further refinement and selection.

Annotation and tagging can continue with the file's content. Beginning with the visualization's or chart's title, an effective title includes a short heading followed by a number of title phrases that match the number of independent variables in the chart. "Titling a drawing speeds the acquisition of…knowledge and dispels potential ambiguity" [12].

Visualization Annotation. For all visualizations, annotate or tag the numbers with labels, graphical context, data point focus and "normal" range. If a number is out of bounds, provide a visual indication.

If the visualization is displaying a range of values for one variable, then use only one coding variable or symbol that varies along one dimension. For example if showing five levels of building heat loss, use one symbol that varies along the size dimension, not five different symbols. If the visualization is a mapped picture, combine its "representational images with scales, diagrams, overlays, numbers, words, and images" [13].

Time Log Annotation. Many users need to store the generation date of the design variations and simulations to track their progress and create a time log. They can create this time log perhaps by using gbXML for script writing or annotation.

The annotations could serve as a tool for decision tracking. With a time log, the user can keep track of how the variables changed and their effect on the results. The annotations could answer, "Why did we pick these three variations? Who decided on them?" More specifically, the annotations could answer "What happened in the design that downgraded the amount of interior light by 50%?"

3.8 Searching and Filtering

Besides annotation and tagging, users might want to search and filter the visualizations to facilitate their decision-making. Both searching and filtering select only certain visualizations. For this research, searching was defined as writing a search query, while filtering was defined as directly manipulating the graphical user interface associated with the visualizations.

"Visualization supports exploration and discovery" [2]. Therefore, present information so that users can use hierarchical filtering to explore data. Filtering would be important to see if each criterion is met. The filters should be additive so the users can see the effect of each filter. The filters should be controlled by data range sliders and updated dynamically, such that the user does not experience a delay in the updating of the results as the user moves the sliders.

Another example of users' filtering visualizations takes place in GPS navigation systems where users can filter the travel route by shortest time, shortest distance, or avoidance of tolls.

4 Discussion

This research's method of a literature review of visualizations seemed to be reasonable. When developing visualizations for design variations and simulations, a designer or developer can consider the following visualization design guidelines [14]:

- Determine if the simulations should reside in the cloud or on the desktop.
- Provide a preview of the time needed to generate requested simulations.
- Rank the simulations automatically on one parameter.
- Provide a visual method to review simulations; avoid textual tables.
- Provide a statistical method to determine the driving factors in the simulations.
- Provide a method to view the simulations quickly, such as using autoadvance or a manual slider control.
- Filter and rank the results on one or more criteria.
- Allow for a method of selecting and comparing two or three visualizations.
- Transport designs and data correctly into and out of the simulation software.

Following the above guidelines should provide users with useful experiences concerning design variations and simulations. The number of these visualizations is likely to only increase as cloud computing becomes more ubiquitous.

References

1. Khan, A., Hornbaek, K.: Big Data from the Built Environment. In: Proceedings of UbiCOMP 2011: 13th International Conference on Ubiquitous Computing, ACM Press, New York (2011)
2. Iliinsky, N.: Preface. In: Steele, J., Iliinsky, N. (eds.) Beautiful Visualization: Looking at Data through the Eyes of Experts, O'Reilly, Sebastopol (2010)
3. Khan, A., Marsh, A.: Simulation and the Future of Design Tools for Ecological Research. Architectural Design 81, 82–91 (2010)
4. Box, G., Hunter, W., Hunter, J.: Statistics for Experimenters: Design, Innovation and Discovery, 2nd edn. Wiley, New York (2005)
5. Ware, C.: Information Visualization: Perception for Design. Morgan Kaufmann, New York (2004)

6. Goldstein, R., Tessier, A., Khan, A.: Space Layout in Occupant Behavior Simulation. In: Proceedings of Building Simulation 2011: 12th International Conference of the International Building Performance Simulation Association, IBPSA, New York (2011)
7. Goldstein, R., Tessier, A., Khan, A.: Customizing the Behavior of Interacting Occupants Using Personas. In: Proceedings of SimBuild 2010: 11th International Conference of the International Building Performance Simulation Association, IBPSA, New York (2010)
8. Goldstein, R., Tessier, A., Khan, A.: Schedule-Calibrated Occupant Behavior Simulation. In: Proceedings of SimAUD 2010: Symposium on Simulation for Architecture and Urban Design. The Society for Modeling and Simulation International, San Diego (2010)
9. Tufte, E.: The Visual Display of Quantitative Information. Graphics Press, Cheshire (1983)
10. Rooke, M., Grossman, T., Fitzmaurice, G.: AppMap: Exploring User Interface Visualizations. In: Proceedings of Graphics Interface 2011: Graphics Interface Conference. Canadian Human-Computer Communications Society, Waterloo, Canada (2011)
11. Goldstein, R., Glueck, M., Khan, A.: Real-Time Compressions of Time Series Building Performance Data. In: Proceedings of Building Simulation 2011: 12th International Conference of the International Building Performance Simulation Association, IBPSA, New York (2011)
12. Bertin, J.: Semiology of Graphics: Diagram, Networks, Maps (W.J. Berg, Trans.). Esri Press, Redlands (2011) (Original Work Published 1967)
13. Tufte, E.: Beautiful Evidence. Graphics Press, Cheshire (2006)
14. Lipton, R.: Information Graphics and Visual Clues: Communicating Information through Graphic Design. Rockport, Gloucester (2002)