2006

Web-based 3-D theatre seating charts and ticket reservation application

Patrick Henstebeck

Master's Project Proposal

Web-Based 3-D Theatre Seating Charts
and
Reservation Application

Patrick W. Henstebeck
pwh2198@rit.edu

Chair: Joe Geigel _____

Reader: Sean Strout _____

0 | Abstract

Seating charts for theatres and stadia have always been an important part of buying tickets to an event, despite offering little information about the view from any particular seat. While the internet has increased the potential to provide this information, it is still difficult to find enough information to make an informed decision. This project provides an application that can provide that information by modeling the view of a theatre from every seat in the house. This application can model a wide range of theatres and includes a seat reservation system to provide an easy way to intelligently purchase tickets.

1 | Overview

Despite the rapid advance of 3-D graphics technology on the internet over the last decade, it is only very recently that theatres, arenas, and ticket brokers offered anything beyond a single static two-dimensional image that only showed the different levels of the stadium and the different sections in each level. Most web sites still only offer a marketing-friendly picture from selected sections in addition to the original image.

There are some instances where an arena will offer more than a single instance. The Madison Square Garden website [1] offers an interactive view of Radio City Music Hall, displaying where every seat is in each section, and offering two views per section: one from the front half and one from the rear. However, the website does not offer a similarly detailed seating plan for its other two venues, showing only dual photos per section for the Theater at MSG, and only a single, poorly lit photograph for the main Arena, and then only for certain types of events.

Other sites that host seating charts for multiple arenas are employing newer ideas and technologies in some areas; however, they all still have some shortcomings. For some arenas, Seat Advisor [2] provides real-life pictures from every seat in a theatre, such as the Helen Hayes Performing Arts Center, although this is not implemented for all the theatres in their database. While this is ideal for the consumer, it's not always practical for an arena to provide photographs from every seat or for a server to store those photographs. It would be almost impossible to provide these photos for a large sports arena such as the Rose Bowl, which seat over 100,000 people. For the arenas, such as my hometown's Bardavon Opera House, which don't provide these images, the site has only a view which shows the relative height of each level of seating and standard, two-dimensional charts of a single layer or section. A more practical and general solution would be to use three-dimensional models of an arena, and there are sites, such as Inter@ctive Venues [3], which use these models. However, this site only offers views from each section, not each seat, which still can allow theatres to hide seats that may be blocked by poles or other obstacles. One site that does provide a view from every seat in the house is Blaxxun's Starlight Express ticketing applet. [4] While providing a beautiful, detailed model of a theatre, it was designed with one specific theatre in mind, and the time and memory required to copy every aspect of the theatre make it impractical for widespread use.

(Continued on page 3)

(Continued from page 2)

Many of the sites that employ 3-D technology still use VRML, the Virtual Reality Modeling Language that still uses protocols and technologies from 1997. Unfortunately, VRML does not provide a high level of reliability or performance, and these issues have forced some sites, such as TicketMaster, to stop offing VRML application that allowed customers to view online seating charts. [5] A newer technology used to display three-dimensional virtual worlds on the web is Extensible 3D, better known as X3D, which builds upon VRML but uses XML to represent attributes, and adds stability and efficiency. [8] It also provided support for more advanced rendering features and direct keyboard interaction, a critical shortcoming of the earlier technology. Also, unlike VRML, X3D plug-ins load only the options required instead of the entire feature set, reducing the space requirement from 2.5 MB for VRML to 300 KB in some cases. [6]

2 | Project Statement

For this project, I will use X3D to present three-dimensional models of theatres and stadiums over the internet. These models will display the theatre, stage, and seating area, and be able to present the view of the main staging area from any seat in the house, given a level, section, row, and seat number. To create these models, I will design an Extensible Stylesheet Language Translation (XSLT) specification that will transform XML files that describe the basic characteristics of a theatre or arena into an X3D model. [9] I will also create a full web application that, in addition to displaying the X3D models for each theatre, will allow users to browse through upcoming events and reserve seats that they feel provide a view that is acceptable relative to the price of the ticket.

3 | Functional Specification

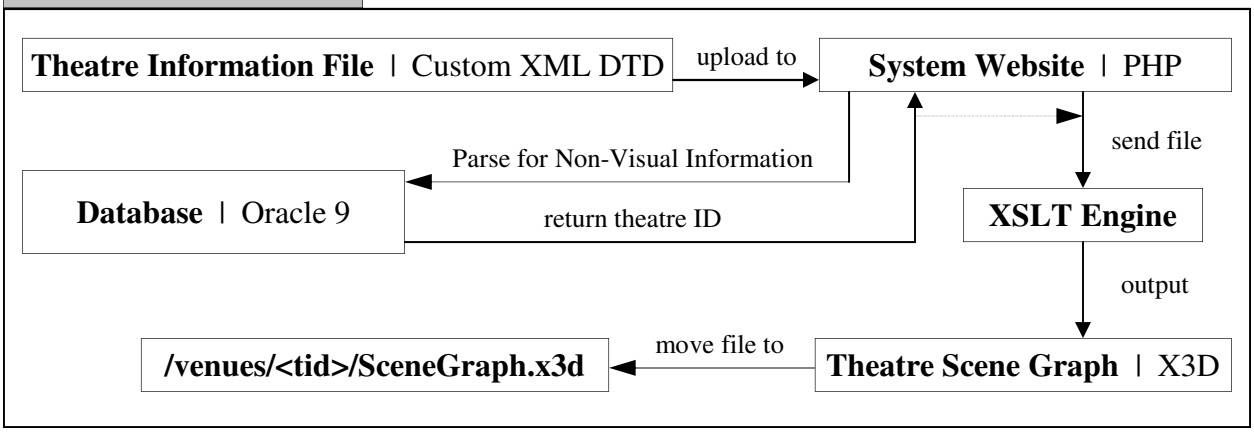
There are three types of users in this system: a database administrator, theatre administrators, and patrons. The database administrator can do anything in the system, and is the only person who can add or remove theatres and users from the system. Theatre administrators can alter information about their theatre and add or remove events. Patrons are limited to viewing theatre information and reserving tickets.

3.1 Creating the theatre model

The first step in making this application functional is creating the X3D models for the theatres. However, before an X3D model can be built, an XML file that contains all the important information about the dimensions of the theatre's staging area and must first be created. A simple program to assist in the creation of this XML file will be included in this project; however, it will only provide forms to enter in attributes for an element and that ensure all required attributes are included when adding the element. For many legitimate theatres, the data needed to build an XML file can be found in StageSpecs, a compilation of stage diagrams

(Continued on page 4)

Figure 1 | System Overview



(Continued from page 3)

and important theatre information. [7] Once the XML file is complete, the theatre can be added to the system. To do this, the database administrator can login to the system and access a web form to create a new theatre object. The form will upload the theatre information file, which will be used to create both the database entry and the X3D scene graph. First, PHP will parse the file and extract the name, location, and seating data, and use that information to create the necessary entries in the database. The database query will return the unique theatre ID to the PHP program, which will then run the XSLT transformation engine, creating the X3D model from the original XML input file. Finally, the PHP program will move the X3D model to a standardized location. After this process is completed, a link will be provided for the theatre administrator to download the X3D file, should they wish to do so. At this point, the theatre will have been successfully added to the system, and its information and seating chart will be available for viewing using any recent browser.

Another web form available to theatre administrators is an add performance form, which takes the name of the event, the original date and time, and how many days for which the event will run at that time. Events can be modified or removed through other web forms at a later date. Theatre administrators can choose whether to have past events removed from the database or have them preserved for historical or data mining purposes.

3.2 Looking at a theatre's information

A patron will be able to register with the system or browse without registering or logging in up until the point where they choose to reserve a ticket. They will be able to look through a list of registered theatres, or search for a certain field. Once a theatre is selected, a page will be loaded with a list of upcoming events at that theatre, which can be searched to find a particular event. A link is also provided to view that particular theatre's information, which will include the X3D seating chart. Alternatively, a patron can search for an event from the main page, and then get a list of theatres that are hosting that event.

(Continued on page 5)

(Continued from page 4)

Once a particular instance of a performance is selected, a page loads that has the X3D seating chart for the theatre at the top of the page and a form to look for tickets at the bottom of the page. The patron can search for a particular seat whose view they've examined and found acceptable, or search for any available seats in a given section, price range, or in the entire theatre. The database will be queried to see if there are any remaining unsold tickets that fit the user's request, and if so, the patron will be given the option to reserve those seats.

The X3D seating charts default to a view that shows all available seats. The patron should be able to zoom in on a section or a seat, and then be able to switch between the seating chart and the view of the stage from the selected seat.

4 | Architectural Overview

4.1 Elements and their associated attributes in an XML theatre information file

- **Theatre** - A single theatre in the system.
  - Theatre Name
  - Location
  - Other information that could be displayed in the seating chart's header.
- **Stage -** A staging area in the theatre.
  - Stage depth & width.
- **Stage Extension** - A protrusion of the stage from the main rectangular area. This element would include the orchestra pit in legitimate theatres.
  - Extension depth & width.
  - Difference in height from the main stage.
  - Location of the extension in relation to the center of the stage it extends.
- **Section** - A group of rows determined to be connected in some fashion.
  - Section name or number
  - Direction towards which the first column of seats are facing.
- **Row** - A group of seats that are all next to each other.
  - Row number
  - Lowest (x, y) coordinate that the row covers
  - Number of seats in the row
  - Number of seats on each end of the row that are unaffected by a curvature in the row
  - Radius of curvature in the row
  - Angle of curvature
  - Number of the first seat in the row
  - Change in the seat numbers as one progresses through a row.
- **Seat** - A single chair in the arena.
  - Unique seat number.

(Continued on page 6)

(Continued from page 5)

4.2 Creation of X3D world

The XML files will be converted into the viewable X3D files using XSLT. Each element in the XML file will be transformed into a basic box element, with its location attributes also determined by the XML file. Other attributes, such as color, will most likely be added by the translator, and used mostly to distinguish between blocks of seats. Also, all animation and keyboard interaction features will be added at this time. This may require the use of a more general-use language with XSLT libraries. Finally, an X3D profile will either be created or an acceptable profile found that will limit the feature set loaded by X3D browsers to those options required by the interactive seating chart.

The process of creating an X3D file from another XML DTD has been used before in other areas. Chemists at Virginia Tech have used XSLT to create X3D and VRML files from their own XML-related Chemical Markup Language, and the Contigra Architecture uses XSLT-to-X3D as the last step in building component-based web applications. [10, 11] Further showing the flexibility and potential of XSLT and X3D, other researchers have used the transform in the other direction, using X3D as a starting point to build a custom modeling language of their own. [12]

Verification of the models' accuracy will have to be obtained outside of the world of computing. The models must be visually compared to the real-life theatre, either through personal visits to a theatre or through existing photographs or videotapes. Madison Square Garden will likely be one venue used for this process, as my personal collection includes many hours of taped events and numerous printed features involving that location.

4.3 Reservation Database

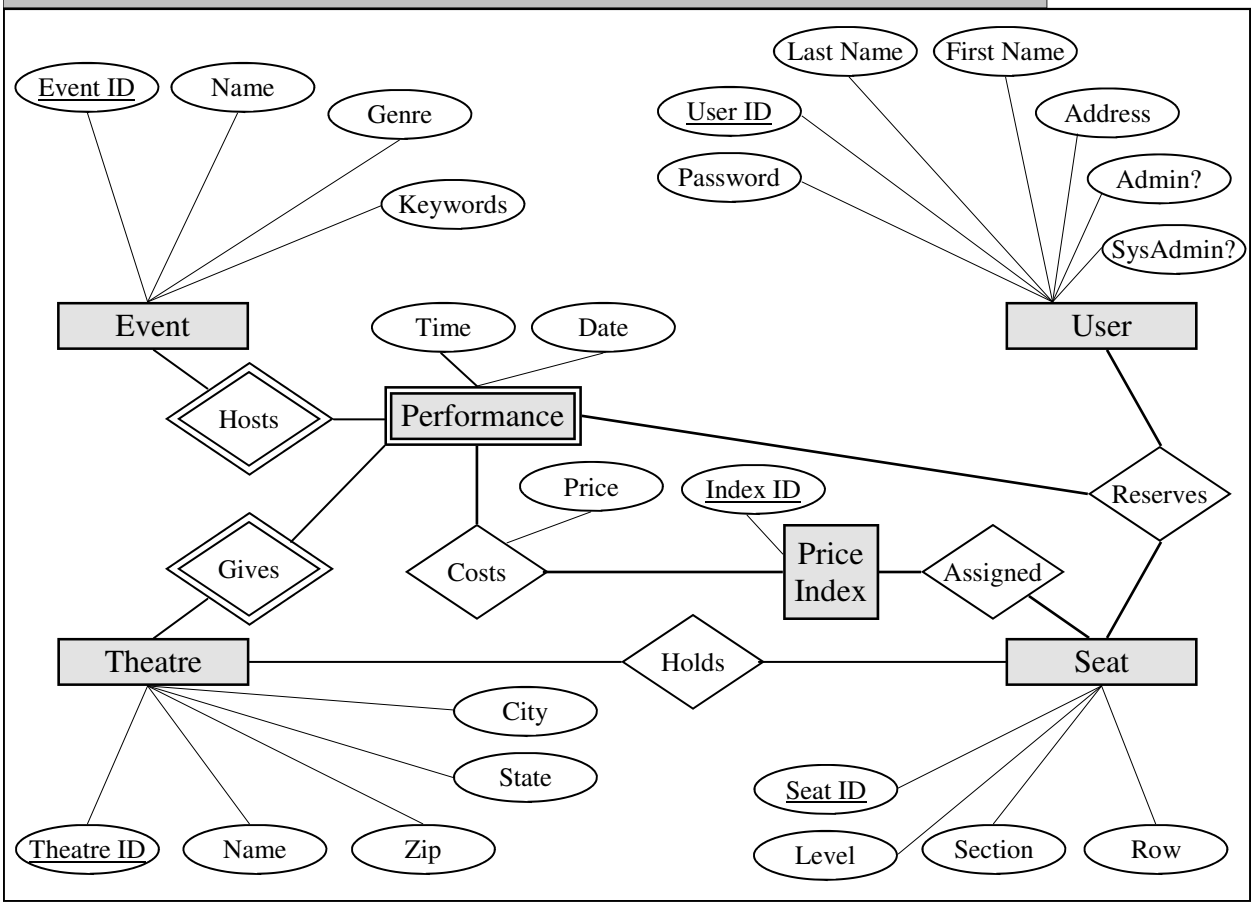
The reservation database will be created using Oracle 9. It will contain five tables for entities representing theatres, seats, events, individual performances and users. There will be a sixth table representing the relationship between performances, seats, and users, specifically which user has reserved each seat for each performance. The table representing the theatres will use the theatre ID (tid) attribute to both identify theatres and to provide a link to the location of the seating chart on the server running the system website. When a theatre is entered into the database, the seating chart must be uploaded at the same time, and the file will be renamed and moved to `/venues/<tid>/seatingchart.x3d`. The seating chart must be uploaded to the server running the associated website at the time the theatre is entered into the database. The tables and their attributes are displayed in the E/R diagram in Figure 2 on the next page.

4.4 System Website

The website will be implemented using HTML and CSS for formatting and PHP for database queries. Viewing the seating charts will require an X3D Viewer; it is expected that the blaxxun3D viewer will be used, as it can be run on any machine or browser that has java installed without an additional download.

(Continued on page 7)

Figure 2 | Anticipated Entity/Relationship (E/R) Diagram for the Reservation Database



(Continued from page 6)

5 | Tentative Schedule

November 19	Project approved.
December 3	Database tables and relations created.
December 10	All necessary queries implemented.
December 16	Web site templates designed.
January 7	Database and Web Site completed.
January 14	Theatre Information File Generation Helper App. completed.
January 21	XSLT to create static objects completed.
February 4	XSLT that adds animation and keyboard interaction completed.
February 11	X3D Generator completed.
February 15	Project Writeup and Component Documentation submitted.
February 25	Project Defense

(Continued on page 8)

(Continued from page 7)

6 | Deliverables

The deliverables for this project will include:

1. Source Code
    a. X3D Generator
    b. Web Site (PHP)
    c. Database (Oracle 9 SQL)

2. User's Guide or Technical Documentation for each component

3. Formal Project Writeup
    a. Introduction and Background
    b. Description of my implementation
    c. Results and Analysis
    d. Future Work
    e. Conclusion

7 | References

[1] Madison Square Garden: Seating. http://www.madisonsquaregarden.com/seating.html .

[2] Seat Advisor. http://www.seatadvisor.com/.

[3] Inter@ctive Venues. http://www.interactivevenues.com/.

[4] Blaxxun Technologies. "Stella Starlight Express Ticketing." http://developer.blaxxun.com/samples/index.html#blaxxun3d/application/stella/.

[5] Hurwicz, Mike. "Web Virtual Reality and 3D – in VRML or XML?". http://www.webdevelopersjournal.com/articles/virtual_reality.html . June 21, 2000.

[6] "Virtual Reality Modeling Language is back from the dead--again." http://news.zdnet.com/2102-3513_22-845018.html .

[7] Gould, Robert B. and Lee, Susan E. Stage Specs: A Guide to Legit Theatres. New York: The League of American Theatres and Producers, Inc, 1990.

[8] Web 3D Consortium. "Extensible 3D (X3D) Documentation." http://www.web3d.org/x3d/.

(Continued on page 9)

(Continued from page 8)

[9] World Wide Web Consortium. "Extensible Stylesheet Transformations (XSLT)." http://www.w3.org/TR/xslt .

[10] Polys, Nicholas F. "Stylesheet Transformations for Interactive Visualization: Towards a Web3D Chemistry Curricula." Proceeding of the eighth international conference on 3D Web technology, March 2003. Pages 85 - 90, 205.

[11] Dachelt, Raimund; Hinz, Michael; and Meißner, Klaus. "Contigra: an XML-based architecture for component-oriented 3D applications." Proceeding of the seventh international conference on 3D Web technology, February 2002. Pages 155-163.

[12] Kim, Taewoo; Lee, Jinho; and Fishwick, Paul. "A two-stage modeling and simulation process for web-based modeling and simulation." ACM Transactions on Modeling and Computer Simulation (TOMACS), July 2002. Pages 230 - 248.