Integration Methodology For Web Based V.R. Training Simulation Tool To Identify Possible Threats During A Stadium Event

Tulio Sulbaran, Ph.D[1]

Abstract – Stadium security has become an important issue since the tragedy of 9/11. According to researchers' findings, pre-event training was featured as standard procedure for most secure stadiums, but lower level security stadiums showed no emphasis for pre-event training. This is a problem because outsourcing security professionals to merely establish a security presence is no longer a guarantee of safety. An attack on a stadium during an event could kill or injure thousands so it is imperative that security professionals receive timely training using improved training programs so that threats that can lead to disasters can be detected, controlled, and avoided. This paper will focus on the detailed description of the integration methodology for a Web based virtual reality training simulation tool prototype for security personnel in identifying possible threats that can occur during a stadium event. This paper also explains the initial ideas in the planning phase of the project, leading into the details of the final prototype.

Keywords: Training, Simulation, Stadium, Virtual Reality, Threats

INTRODUCTION

Due to the unfortunate terrorist attacks of September 11, 2001, additional multiple security measures are required to protect the life of the people attending large sport events such as football. One of these multiple measures is the proper training of security personnel to detect threats. The preparation and delivery of this training is very challenging due to many factors such as: vast number of possible threats, high range of stadium conditions, and constant security personnel turn-over just to mention a few of the challenges.

In order to develop a Web based virtual reality representation of a stadium to be used for training purposes, the researcher had to perform several tasks. This paper is a detailed description of the integration methodology for the prototype, explaining the initial ideas in the planning phase of the project, leading into the details of the final prototype. The following activities outline each step taken in planning and creating the prototype.

ACQUIRING AND STUDYING THE M. M. ROBERTS STADIUM ARCHITECTURAL PLANS

The first step the researcher took in developing the Web based virtual representation of M.M. Roberts stadium at the University of Southern Mississippi was acquiring the architectural plans or blueprints of the entire structure from the university's physical plant. The plans acquired from the physical plant were very detailed and proved to be very beneficial to the researcher. Although a few decades old, the blueprints were still in good condition and the researcher was able to study the shapes as well as the dimensions of the different structures that makeup the stadium without much difficulty. Once the plans for the stadium were obtained, the researcher studied over them carefully and took notes of dimensions for the West side of the stadium. The researcher then repeated the process for the East side of the stadium and the upper stadium seating areas. The researcher found that the West and East side lower seating areas were different in size and structure. However, it was discovered that the dimensions for the upper seating areas of both sides of the stadium were identical. Once this process was complete, the researcher began the process of taking digital photographs of the stadium itself and the surrounding area.

---

[1] Associate Professor – School of Construction at the University of Southern Mississippi, Box 5138, Hattiesburg, MS, 39406. E-mail: Tulio.Sulbaran@usm.edu.

TAKING DIGITAL PHOTOGRAPHS OF M. M. ROBERTS STADIUM AND THE SURROUNDING AREA

The next task performed was the use of a digital camera to take snapshots of every aspect of the stadium structure as well as the surrounding area. First the researcher focused on the lower seating area of the West stadium seating area. Images were taken of almost every elements of the stadium including the seats for the patrons, steps, concessions areas, restrooms, railings, signs, doors, doorways and walkways, smaller structures beneath the stadium seating area, gates, vomitories, and ramp entrances. Since the West and East stadium lower seating areas are different in size, the same process was repeated for the East side so that the differences between the two seating areas would be captured in images and used for future reference. Next the researcher took digital photographs of the upper seating area for both the West and East stadium sides. Although the upper seating area structure is identical for both sides, signs indicating seating rows are different and had to be photographed for future reference. Once all of the photographs were taken for both West and East stadium areas, the researcher captured images of the area surrounding the stadium including the athletic building, ticket office, scoreboard, and other small structures. All of the images were then exported from the digital camera and saved to the researcher's hard drive for future reference.

BUILDING THE WEB BASED VIRTUAL REALITY REPRESENTATION OF M.M. ROBERTS STADIUM IN VIZX3D

The researcher used the Virtual Reality Modeling Language (VRML) authoring toolkit, VizX3D (now known as FluxStudio), to build a Web based virtual representation of M.M. Roberts stadium at the University of Southern Mississippi. Using the architectural plans and digital photographs for reference, the researcher was able to build the 3D models of the West and East stadium seating areas as well as the upper seating areas, the football field including the goal posts, the structures near the south end zone including the scoreboard and ticket office, the athletic building that is near the north end zone, and other objects such as fences, foliage, and lights (Shown in Figures 1 - 4).
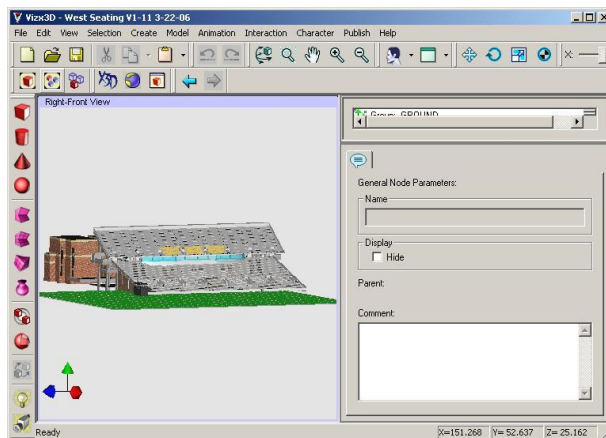

Figure 1. Screenshot of West stadium in VizX3D
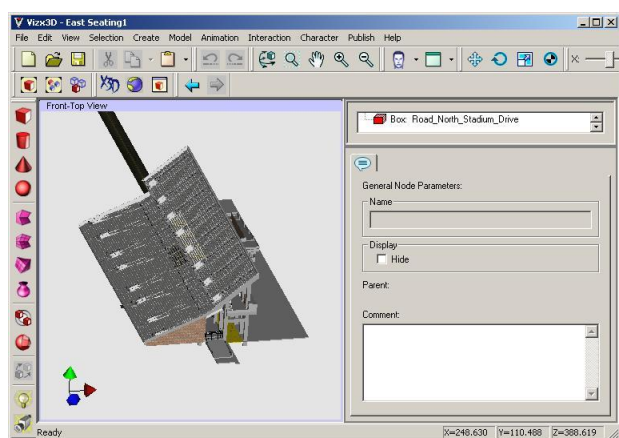

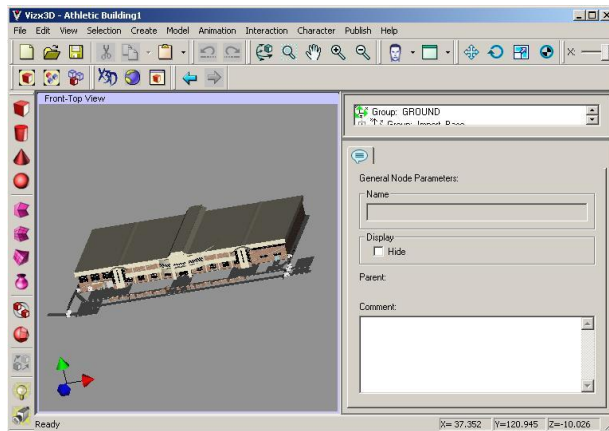Figure 2. Screenshot of East stadium in VizX3D

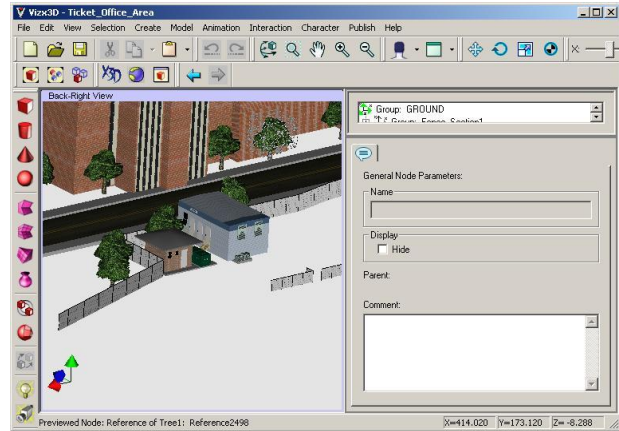Figure 3. Screenshot of athletic building in VizX3D   Figure 4. Screenshot of ticket office area in VizX3D

The researcher used the four primitive shapes (box, cone, sphere, and cylinder) provided by VizX3D to build most of the stadium structures and other objects. Editable meshes, which allow the user to shape an object by moving the vertices, were also used to create some of the more complex shaped models such as seating brackets and vomitories. Each area was saved as different VRML files and then the researcher created a VRML world file that uses inline nodes to call the other VRML files into it thus assembling the entire M.M. Roberts stadium environment (Shown in Figure 5).
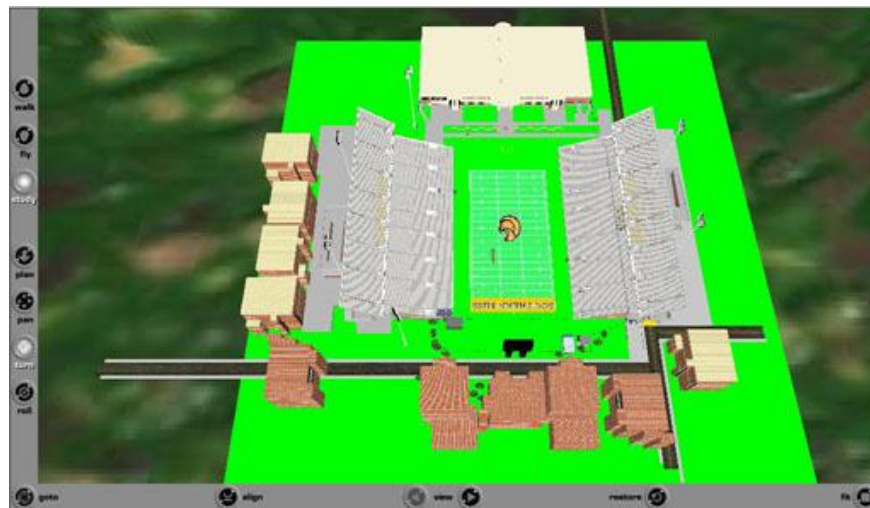

Figure 5. Screenshot of VRML stadium environment

CREATING TEXTURES TO ADD REALISM TO THE VIRTUAL REALITY STADIUM MODELS

The researcher used Adobe Photoshop software to create the textures from the photographs taken of the stadium and surrounding area. The researcher transferred the digital photographs of the stadium area and structures from the camera to the computer. Once the photographs were transferred, the researcher opened the images using the Adobe Photoshop software and began editing the images to create textures (Shown in Figure 6).
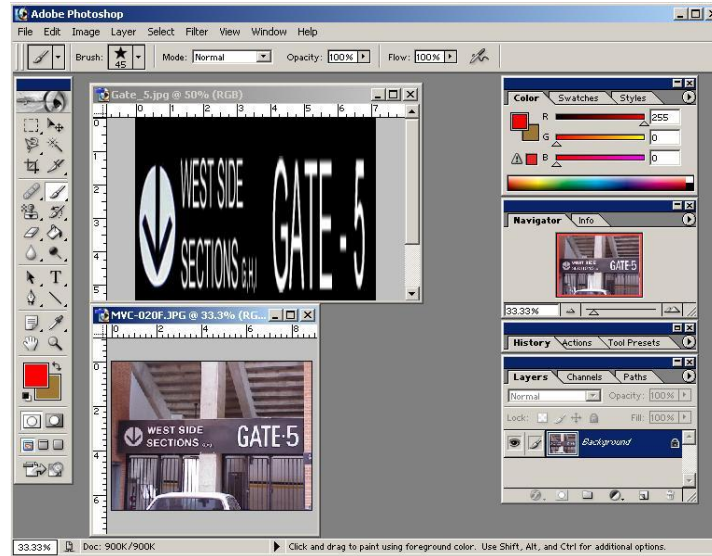
Figure 6. Screenshot of creating textures in Adobe Photoshop

By editing the photographs using the Adobe Photoshop software, wall textures, signs, seating textures, gates, door textures, window textures, and others were created, saved in JPEG format, and applied to the 3D models inside the VizX3D software (Shown in Figure 7).
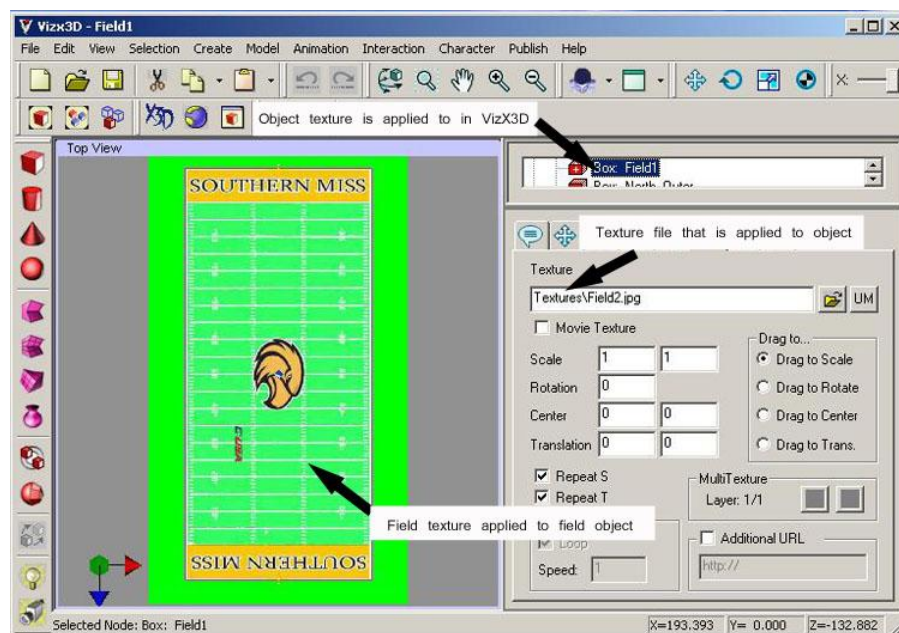

Figure 7. Screenshot of texture application in VizX3D

MEETING WITH A PANEL OF EXPERTS

During the beginning of the Spring 2006 semester, the researcher put together a meeting with a panel of experts from the university to identify common threats and appropriate responses. The researcher met with a panel of experts from the Human Performance and Recreation Center and stadium security professionals at the University of Southern

Mississippi to discuss some of the threats they encounter and what actions should be taken in response to certain threats. The experts from the Human Performance and Recreation Center included Dr. Jerry K. Phillips, Dr. Walter R. Cooper, and graduate student, Ms. Lauren Beckman. The stadium security experts included Mr. Terry Steed, Mr. Ed Grimsley, and Chief Bob Hopkins of the University of Southern Mississippi's Police Department. The researcher prepared a questionnaire for each individual to answer and submit after the meeting. During the meeting, the researcher, using the virtual reality stadium, performed a walkthrough and asked each individual for input relating to specific threats and which areas of the stadium are considered high risk if a threat were to occur. All of the experts on the panel agreed that common areas of the stadium such as restrooms, traffic areas, electrical rooms, and non-secured utility rooms were targets for a threat. All experts agreed that if someone were to put X threat at Y location, then Z disaster could happen. Since some of the areas of the stadium such as the electrical rooms and utility rooms are not open to the public, the researcher was unable to include interior models of them inside the overall virtual reality stadium. Much of the information obtained by the researcher from the panel of experts is deemed law enforcement sensitive material and cannot be disclosed here. The information will only be used as reference during the development of the virtual reality stadium security training tool for the University of Southern Mississippi.

CREATING THE THREATS AND NON-THREATS IN VIRTUAL REALITY

The researcher used VizX3D to create virtual representations of possible threats and non-threats that security personnel could encounter at the stadium during an event. The threats include models of patrons wearing heavy clothing or attempting to bring prohibited items into the stadium (Shown in Figure 8). Other threats include models of unidentified packages and misplaced bags placed or hidden around different areas of the stadium. Models of unauthorized vehicles are also used (Shown in Figure 9). The non-threat models include innocent patrons (Shown in Figure 10) and large stationary garbage cans that can be found in different locations of the stadium.
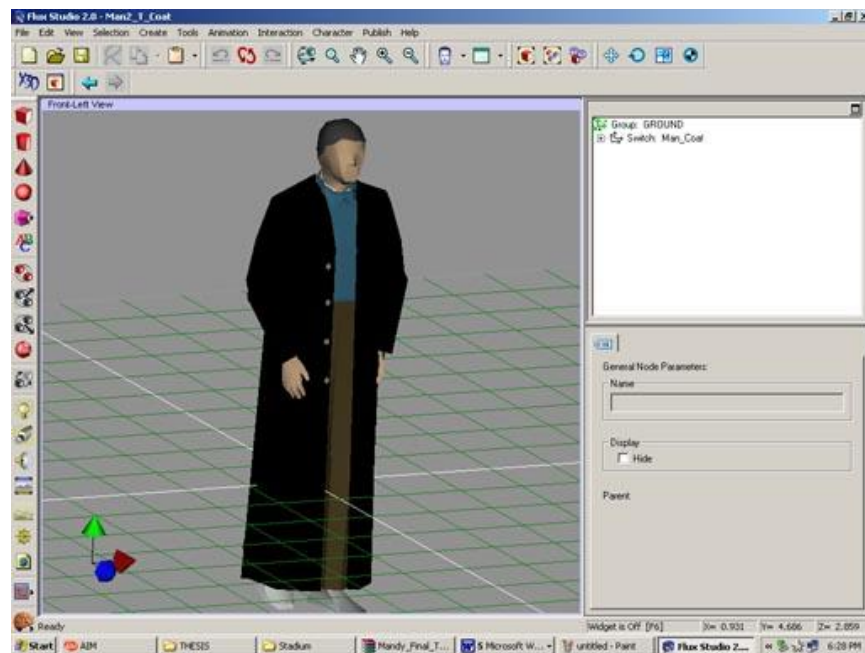

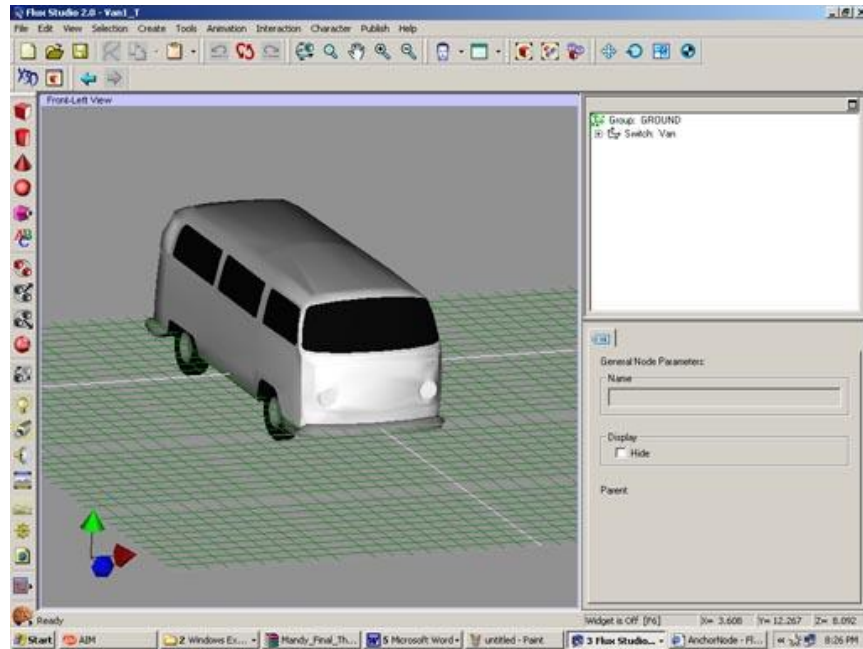Figure 8. Screenshot of potentially threatening patron in VizX3D

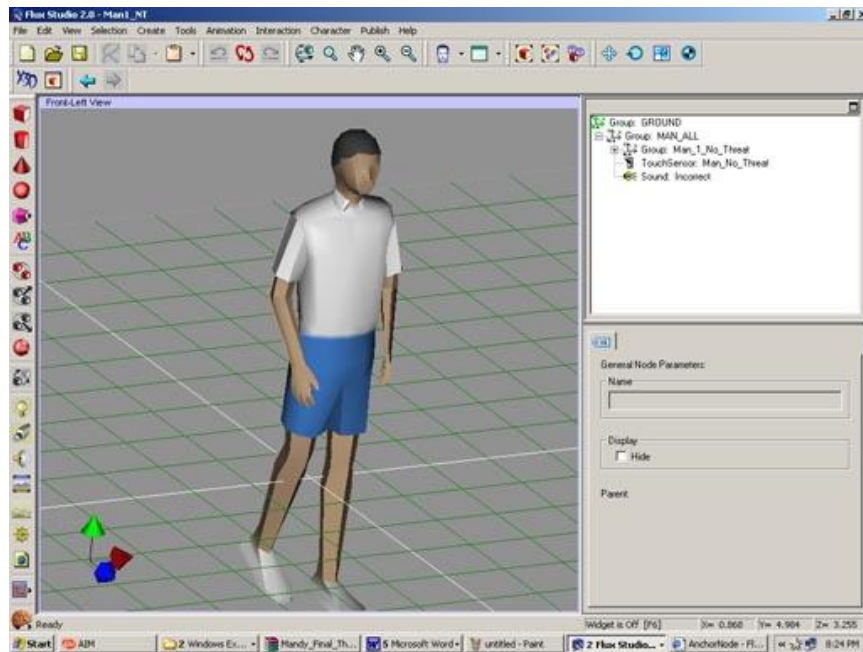Figure 9. Screenshot of unauthorized vehicle in VizX3D

Figure 10. Screenshot of non-threatening patron in VizX3D

The researcher was able to download most of the models needed, such as models of people and vehicles, from free 3D modeling Web sites. Like the virtual reality stadium, the researcher used VizX3D to create other items needed and to alter existing models. Adobe Photoshop software was used to create or edit any textures needed for the threat models and then applied in VizX3D. Once the researcher completed modifying the threats, interactivity was added to each one using special VRML nodes inside VizX3D. A switch node was used in combination with a touch sensor so that when a user clicks the threat inside the training session, the model disappears and a textured box appears in the

same place. The touch sensor also triggers a sound node of a harp indicating the correct identification of a threat by the user. The box that appears uses both a billboard node so that it always faces the user and an anchor node that opens a Web page in a separate browser window. The non-threat models include a touch sensor that triggers a sound node of a buzzer indicating that the user incorrectly identified a potential threat. Upon completion, each threat and non-threat model was then saved in VizX3D. To ensure that the locations of each threat would be different each time the training tool begins, the researcher identified five areas of the stadium a particular threat might be found. The researcher then placed that particular threat in five different locations in the virtual reality stadium world, wrote a short description of the location, and recorded the X, Y, and Z coordinates. The researcher repeated this process for each threat. Using the coordinates, the researcher then exported five different VRML files of each threat using the naming convention "Threat_XX.wrl". The first "X" in the filename is equal to a number that represents which threat. In this case, the number can be from 1 to 15 meaning there are fifteen different threats. The second "X" is equal to a number that represents the location of the threat. Since each model is exported five different times with five different coordinate values or locations, the second "X" can be a number from 1 to 5. For example, if the second threat in the third location is placed in the VR stadium environment when generated, then the threat model chosen is "Threat_23.wrl". This process is repeated for the non-threat models, but they are only exported three times as VRML files using the naming convention "NThreat_XX.wrl".

CREATING THE LEVEL-OF-DETAIL (LOD) STADIUM MODELS

In VRML, there is a special node known as Level-of-Detail or LOD. Using this node, the author of a VRML world can place models ranging from lowest detail to highest detail of the same object in the same location. The LOD node allows the author to set a distance or range that will switch from one level to another depending on the distance from the user's viewpoint to the object. In simpler terms, the closer the user gets to the object, the more detailed version of the model the user will see and the farther away the user is, the more crude version the user will see. The LOD node helps to speed up the rendering time of the VRML environment because the highly detailed models only appear if the user is near it.

The west and east stadium structure models created by the researcher are very detailed. To save on rendering time, the researcher decided to test the LOD node in the VRML stadium world. After testing to make sure it functioned correctly, the researcher then took the model representing the west stadium structure and removed a great amount of detail including seats, railings, beams, textures, and other components. The researcher then repeated the process for the east stadium model and the end result was two very crude models of the west and east stadium structure. The researcher then added these models to the main VR stadium world, initialized two LOD nodes (east and west), and saved the main stadium world file. Now when the user viewing the VR stadium world is a certain distance from either stadium structure, the LOD node switches from high level to low level (Shown in Figures 11 and 12).
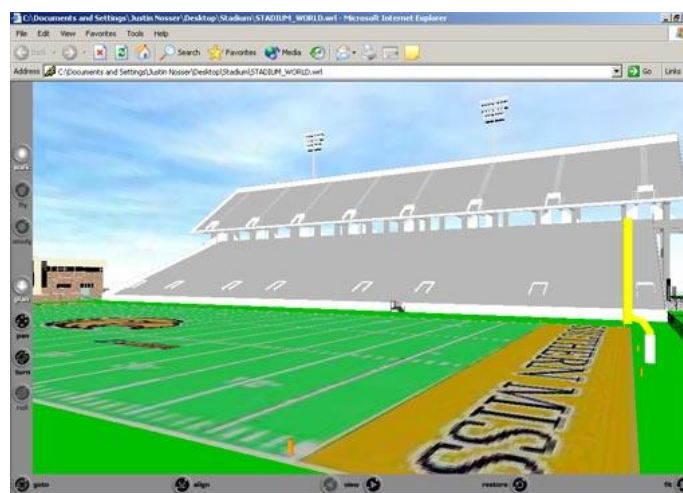

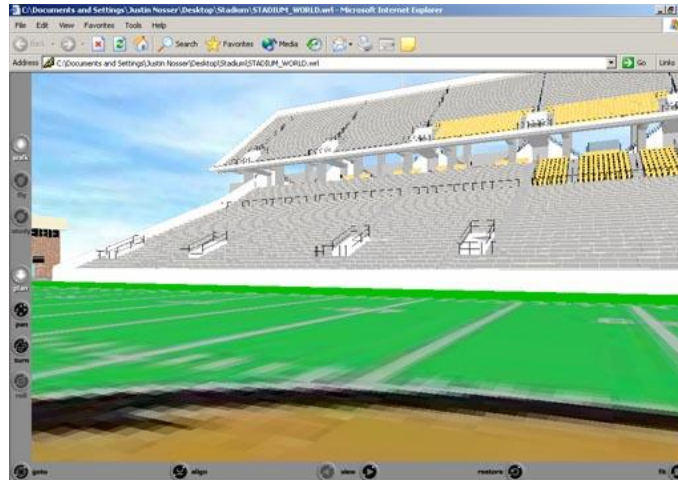Figure 11. Screenshot of low level of detail east stadium structure

Figure 12. Screenshot of high level of detail east stadium structure

DEVELOPING THE OUTPUT WORLD

To ensure that the main VRML file to be generated for the training exercise would be relatively small in size, the researcher used what is known in VRML as prototypes and external prototypes. Defining something as a prototype or PROTO is similar to creating your own VRML object that can be used over and over. External prototypes or EXTERNPROTO are prototypes defined in external files and referenced by a URL. The researcher defined the main stadium world file, threats files, and non-threats files as prototypes and then created a simple VRML output file that referenced them as external prototypes. The following table shows an example of defining an object as a PROTO and then referencing it as an EXTERNPROTO in another file:

Table 1: Defining PROTOs and EXTERNPROTOs

#VRML V2.0 utf8
#Threat file
PROTO Threat_1 [
] {
Group{
     children[
#*****************
 	
#****************
    ]#children
} #group
 	
}#proto end
	#VRML V2.0 utf8
	#Output file
	EXTERNPROTO Threat_1[]"Threat_1.wrl"
	DEF t1 Threat_1{}

This allows the output file, once generated by the user, to remain small in size even though several external prototypes can be called into it at once.

IMPLEMENTING THE APPROPRIATE RESPONSE WEB PAGES

Using Microsoft FrontPage software, the researcher created simple Web pages containing appropriate response information relating to one of the possible threats. Each page contains a question, a drop-down list of three possible answers, a Submit button for submitting the answer and the pertaining threat to the database, and a Reset button for deselecting a chosen answer in the drop-down list. Once the Web pages were created, the researcher placed them in a directory on the server and modified the URL of the anchor nodes in each threat model to link the pages to the corresponding threat. Another page thanking the user opens when the Submit button is pressed and the user is instructed to click the Close Window button on the new page and continue with the training session until completed.

ESTABLISHING THE DATABASE

For the VR stadium training tool, the researcher created one database using Microsoft SQL Server 2005 Management Studio software. The database is named stadium and it is comprised of three tables: users, new user, and user activity (Shown in Figure 13).
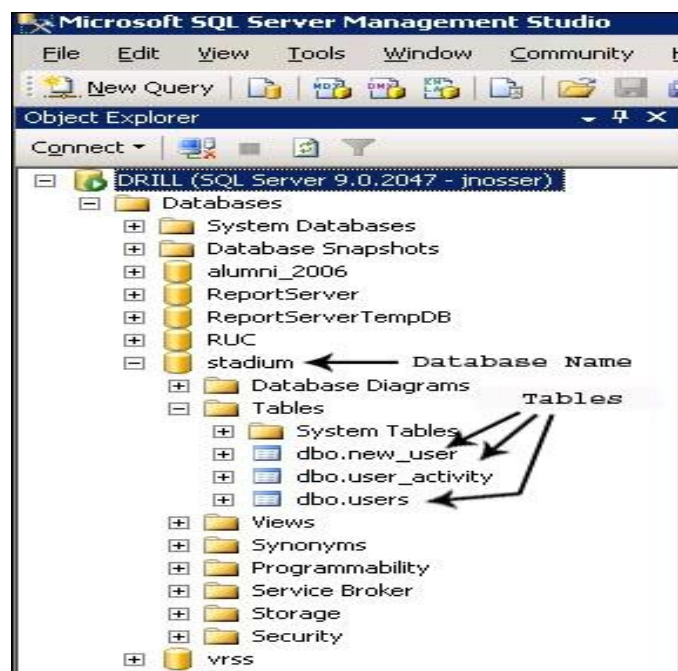

Figure 13. Screenshot of stadium database and tables in Microsoft SQL Server 2005

The new_user table receives information provided by a user when they register to use the stadium training tool. The users table receives the user registration information upon approval from the administrator. The user_activity table includes columns that are filled once the user has completed the exercise. The information from this table is retrieved by the administrator after the training session has ended and is used for evaluation. This database is needed to hold the registration information of registered users, new users awaiting approval, and the performances of the registered users on the VR stadium training tool.

MODIFYING THE LOGIN AND WELCOME WEB PAGES

Since the login process for the VR scheduling simulation is exactly what the researcher required for the VR stadium training tool, the login and welcome Web pages from the past project were copied and placed into the main stadium directory on the server. Next, the researcher edited the text on the Web pages using Macromedia Dreamweaver to adapt them to the VR stadium project. The welcome page was completely redone to include a brief overview of the project, detailed instructions on how to use the VR training tool, and a link to install a VRML viewer plug in located on the server (Shown in Figure 14).

Figure 14. Screenshot of welcome page for VR stadium project

The buttons on the login page were also edited to link to files stored in the project's directory on the server. For example, the admin button code was edited so that when pressed, the user account java server page is displayed so that the administrator can approve or deny new users.

MODIFYING THE JAVA APPLICATIONS

A past project completed by graduate students for the Innovation of Construction and Engineering Enhancement (ICEE) in the School of Construction at the University of Southern Mississippi using VRML in combination with Java servlets for a virtual reality scheduling simulation proved to be very useful for the researcher's virtual reality stadium training tool. Since the VR stadium project is a Web application, it was imperative that Java servlets be used. The researcher received assistance from the graduate students and modified the already existing Java applications to work with the VR stadium project files.

The first step was deciding which Java source files from the VR scheduling simulation project would be vital for the VR stadium project. These Java source files were then copied and placed into a directory within the project's main directory on the server. A Java server page was also copied and placed in its own directory on the server. Next the Java source files were examined, modified, and compiled with JBuilder 2005 software, and then tested to ensure that project components were working properly with the Java classes such as connecting to the database, storing the registration information, and allowing approved users to login. The Java server page was tested as well to ensure that the newly registered user's email addresses were being placed on the JSP so that the administrator can view them and then approve or deny their request to use the VR stadium training tool. Approving or denying their request sends an email to the user notifying them of their status and if approved, the registration information is placed into the database.

The next step was testing the Java source file that generates the stadium VR world when the button is clicked on the project welcome page. The idea behind the code was that when the button is pressed, a Java class is called that creates a VRML file with extension .wrl and then writes text to this file that externally calls the VRML files needed to create the VR environment. After some modification, the code worked perfectly.

The researcher began the next step to bring threats into the VR stadium world by adding unique random number generator code to the Java class file that writes the output VRML file. After careful research, a unique random number generator code was discovered and edited to fit the requirements of the stadium project. In simple terms, fifteen unique numbers are generated and the first five are placed into an array. Next, five unique numbers are generated by a second unique random number generator and they are placed into a second array. These numbers,

beginning with the numbers from the first array, are then concatenated to form one number. For example, 15 and 5 become 155. This new number was then appended to the section of code in the output file that externally calls the threat model prototypes.

The next step was ensuring that the non-threat models would also be externally brought into the generated VR environment. The researcher added a simple for loop to the Java source code that would generate consecutive numbers and append them to the section of code in the output file that externally calls the non-threat models. The code was tested and it worked.

The final step was modifying the Java source code so that when a user registers to use the VR stadium training tool, a folder is created using their username and placed in the directory on the server where the VRML files are stored. If approved, the user can login and when the Generate VR World button is pressed, it creates the VRML file and places it in their folder. By doing this, multiple users can use the VR training tool and have their own unique world created for them. Once this final step was completed, the code was tested by generating a new VR stadium environment and everything worked precisely as planned.

ORGANIZING THE DIRECTORY STRUCTURE

The directory structure of the VR stadium project is very important to the functionality of it as a whole. Inside the main directory are three other directories: Questions, Stadium_VR_Files, and WEB-INF. A representation of the directory structure is shown below in Figure 15.
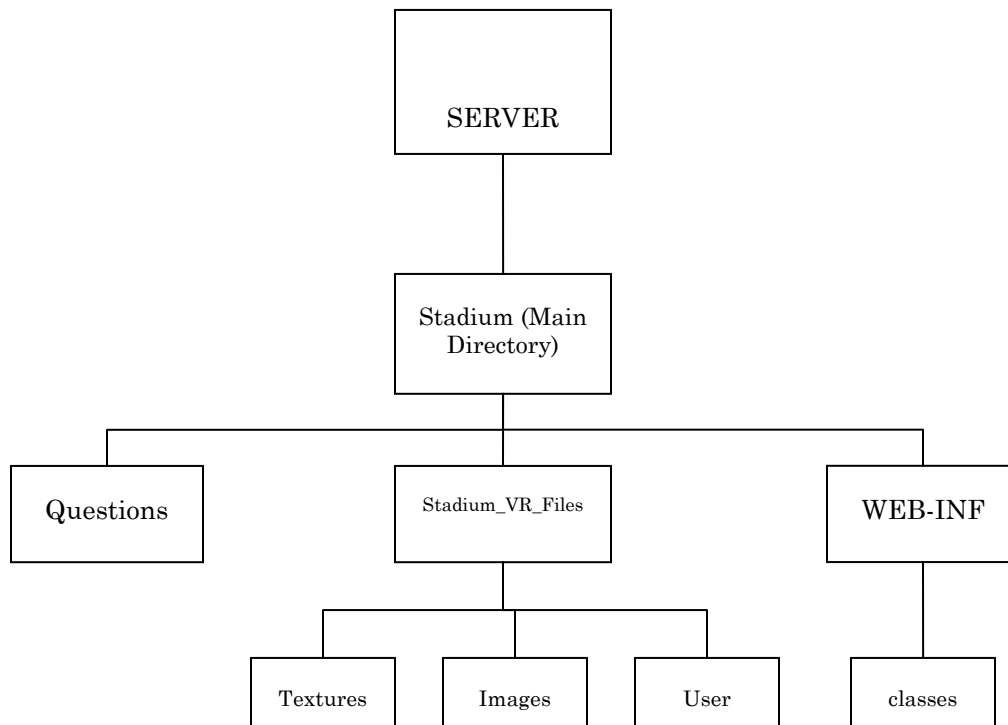
Figure 15. Directory structure of the VR stadium project

The Web pages relating to appropriate responses to threats reside in the Questions directory. These pages are referenced by the anchor nodes that each threat model possesses. If moved, each threat model's anchor node would have to be edited to reflect the change of the Web page's location in the main directory. All of the VRML files are located in the Stadium_VR_Files directory. This directory includes other directories relating to the VRML models: Textures and Images. These two directories contain image files that are applied to the models inside of VizX3D. The Stadium_VR_Files directory also houses the directories that are created whenever a person registers to use the VR training tool. The directory is always named after the user name specified by the person at registration. The third

directory inside the main directory is WEB-INF and it holds the classes directory that contains all of the Java class files needed to run the VR stadium project. It is the researcher's recommendation that the administrator keep the directories and files located in the stadium project's main directory as they are now. If the administrator does decide to make any changes, it is imperative that the changes be recorded so that the proper adjustments can be made to the files or errors could occur when a person tries to register, login, or run the VR training tool.

TESTING THE APPLICATION

The final activity performed by the researcher was testing the application. Many test runs of the project were performed personally by the researcher. The test runs included registering three anonymous users, approving or denying them, logging in using one of the approved user's usernames, and then generating a VR world. After generating the VR world, the researcher navigated through the world and checked the appearance and functionality of the threat models. Once completed, the researcher then checked the databases to ensure all of the information regarding user performance was saved into the appropriate columns in the user performance table. The other tables were also checked to make certain user information was saved accurately.

SUMMARY

In the first phase of this project, the researcher had to imagine how the VR stadium training tool would function and write a methodology. As time progressed and the project became more than just an idea, the researcher was forced to make some changes to the preliminary methodology. Extensive research into VRML, Java, and database administration and the aid provided by fellow graduate students were contributing factors to the modifications made to the methodology. Although the researcher's limited experience with Java programming and database administration were reasons for some of the changes, the initial ideas were not discarded entirely and the VR training tool works as originally planned.

ACKNOWLEDGEMENT