Theatrical Storytelling in a Virtual Space

Joe Geigel
Department of Computer Science
Rochester Institute of Technology
Rochester, NY 14623
585-475-2051

jmg@cs.rit.edu

Marla Schweppe
School of Design
Rochester Institute of Technology
Rochester, NY 14623
585-475-2754

Marla.Schweppe@rit.edu

ABSTRACT

In this paper, we explore the uniqueness of storytelling in a theatrical context. From this exploration, we present a theoretical context for classifying theatrical performance and describe a technical framework for creating such performances in a virtual space. The framework allows for interplay between human actors, virtual performers and live audience members cohabitating both physical and virtual spaces. We describe a prototype implementation of the framework and provide the details of a distributed performance given in the spring of 2004 as a proof of concept for the system.

Categories and Subject Descriptors

I 3.7 [**Computer Graphics**]: Three Dimensional Graphics and Realism – *Virtual reality*

General Terms

Algorithms, Design, Human Factors

Keywords

Virtual theatre, distributed performance, storytelling.

1. INTRODUCTION

What is theatre? One definition can be obtained by considering the derivation of the words *theatre* and *drama* [1]

> *The word comes from the Greek theatron, or "seeing place." It is a place where something is seen. And the companion term drama comes from the Greek dran, "to do." It is something done. An action. Theatre: something is seen, something is done. An action witnessed.*

When we talk about theatrical storytelling, the story is not merely told, instead, it is performed, acted out. Just as important is the fact that the performance is witnessed. "Theatre does not exist as a physical reality unless it is an event including both performer and spectator."[2]

Theatre is one of the oldest forms of storytelling and virtual worlds present one of the newest venues for presenting theatre.

Our goal for this project is twofold. First, we set out to explore theatrical storytelling and categorize the means by which it

can, and has, been brought into a virtual setting. Secondly, we set out to define and create a basic technical framework for theatrical storytelling in a virtual, distributed environment while keeping within a traditional theatrical paradigm.

We begin by presenting a survey of works that combine theatre with virtual reality. Next we consider the mechanisms of theatrical storytelling and present our classification for performance. This is followed by the definition of an architecture for a virtual theatre system and a description of the distributed virtual theatrical piece, *What's the Buzz?*, which was performed in the spring of 2004 as a proof of concept for our system. Finally, we present conclusions and discuss future work.

2. BACKGROUND

For purposes of this paper, we define "virtual theatre" as the integration of virtual reality technologies into the theatrical storytelling process. Considering the use of the virtual space, an obvious application is the use of a virtual world as a stage. One such example, which is one of our primary inspirations, is *VRML Dream*, a distributed version of Shakespeare's *A Midsummer Night's Dream*, which was streamed over the Internet in April of 1998 [3]. Virtual spaces have also been used to assist in theatrical productions on *physical* stages in both the areas of lighting and set design [4] as well as actor rehearsal [5].

The merging of physical and virtual spaces as part of the theatrical performance is another active area of exploration. Experimental theatrical performances on physical stages such as works performed by the Troika Ranch Dance Company [6] and those that employ the Interactive Virtual Environment (IVE) developed at MIT [7], have incorporated computer-generated effects that become an integral part of a performance. In these works, the action of the performers on a physical stage controls theatrical elements resident on a computer, which are then projected in the physical space.

Research in virtual theatre has focused not only on the use of the virtual space, but also on defining and guiding the behaviors of performers that inhabit the stage. Advances in software agent technology and artificial intelligence has allowed for the creation of virtual autonomous actors that can react and actively participate in a theatrical performance. Such performances have been created for the physical stage, virtual stage and a combination of the two. In Claudio Pinhanez's *It/I* [8], as well as productions performed by the i.e VR Project at the University of Kansas [9] human and virtual characters share a physical stage with the virtual actors appearing on stage via image projections. In works like AlphaWolves [10], human participants interact with the virtual characters that exist entirely within a virtual space. Using New York

University's IMPROV System [11], one can create performances that take place on a mixed reality stage where computer actors in a virtual space can react to human actors in a physical space.

One unique aspect of theatrical storytelling is that the audience is an essential participant in the performance. Without becoming part of the story, audience interaction can greatly affect the performance of the actors on stage. Experiments in combining *collaborative* virtual environments with broadcast television have been undertaken in the Inhabited Television project [12]. This project not only illustrates the viability of an interactive, distributed, and scripted entertainment in a virtual space, but also does so in the archetypal framework of an existing storytelling medium, namely television.

3. THEATRICAL STORYTELLING

3.1 Theatre as Collaboration

Theatre, by its very nature, is a collaborative art. Theatre can be defined as a *place of action; field of operation* [13]. The phrase "theatre of war" is often used when describing historic battles. In this sense, success is achieved by the coordination of groups of participants moving toward a common goal in a shared space. When applied to art and performance, the telling of a theatrical story takes place in a common field of operation and involves a number of artists who create the storytelling space and action that is viewed and perceived by an audience. The artists involved include.

- Playwright – describes space, sound, movement and lays down blueprint for the characters and story

- Director – interprets playwrights plans and applies them to the space

- Designers / Technicians – design and create and control the actual physical and aural elements placed space.

- Actors – execute the actions within the space, create the motion and action of the characters

- Audience – reacts and interacts with the action.

3.2 Role of the Audience

Other forms of media, like film and television, also involve the telling of stories via acting and share the set of participants mentioned above. What distinguishes the theatrical experience from these other media is the role of the audience. In theatre, the audience takes an active rather than passive role in the entertainment. The performance of a story within a theatre is not just live, but it is also interactive. The audience is present to witness the performance and audience reaction drives the actors. Every performance is different, unlike film and TV where the entertainment is fixed, captured in time.

In both film and theatre, the goal of the storyteller is to grab the attention of the audience. In film, this is done irrevocably; the audience cannot view any part of the scene that the filmmaker does not include. In theatre, the job is a bit more challenging as the storyteller must earn this attention since the audience has more control over where he or she looks. Many devices must be employed in the theatre to control the focus of the audience's attention. Motion, actor focus, and lighting all play an important role in this effort. [14]

Audience expectations are certainly different. Theatre is a melding of reality and imagination. In many films, the perception of reality is portrayed in detail, in theatre it is suggested by use of set elements, lighting, and action. In this way theatre is more similar to animation than to live action film. But different from animation, since theatre is live, the chance of mistakes and variations and happy accidents exist.

The field of interactive drama [15] moves the role of the audience in the story from active observer to collaborative participant. This project does not seek to redefine theatre, merely the method of presentation. In Virtual Theatre, as we define it, the audience is still an observer, and although it may influence the performance of the story, it doesn't create the story as it would in interactive drama.

3.3 Classifying Theatrical Performance

Based on the discussions above, we present a classification of theatrical performance. The classification is adapted from a similar classification for Collaborative Virtual Environments presented in [16], modified to focus specifically on theatrical performance. Performances are classified using three dimensions. Each dimension is defined using a sliding scale between a set of extremes. A particular performance can be classified by a value on the scale for each dimension. The dimensions are as follows:

- Stage (Physical vs. virtual) -- Defines where the stage exists on which the performance takes place.

- Participants (human vs. computer) – Each of the participants involved in the performance is classified by how much of the their actions are controlled by a human and how much is controlled by a computer agent.

- Audience Presence (passive vs. active) – Describes the effect that the response of the audience has on the performance.

- Performance (scripted vs. intereactive /improvisational) – Describes the specificity of the path the performance with take with each iteration.

Figure 1 shows the classification of various performances, both traditional and virtual along the dimensions of Audience and Stage. Considering the stage axis, traditional forms of performed entertainment are classified on the left since they are performed in a physical space (stage, sound stage, set, etc). Moving up the Audience axis, the level of audience participation becomes more interactive. Special effects in films, where elements created by animators in a virtual space are composited with live action footage, can be considered performance in a mixed-reality setting. NYU's IMPROV represents active participation within a mixed-reality setting where virtual actors in their virtual setting interact with human participants in a physical space. Finally, computer animation represents performances that exist solely in a virtual space with minimal audience participation, whereas Inhabited Television implies full audience immersion on the virtual performance stage.

In Figure 2, performances are classified along the axes on Participant and Stage. Traditional theatre, where works are performed on a physical stage with human actors falls in the lower left of the graph. Remaining in the physical realm and increasing the role of the computer generated or controlled participant, we classify Troika Ranch performances where

computer elements under human control are an essential part of the performance mid range, and It/I, where a completely autonomous agent is an actor in the performance, at the top. Considering performances on a mixed stage, special effects fall in the middle being that the effects or digital characters do not exist in the physical world, however, are created and controlled by human animators. The IMPROV System once again, exemplifies totally autonomous characters interacting in a mixed space. Moving to the completely virtual space, computer animations portray animator controlled virtual characters, whereas systems like AlphaWolves presents totally autonomous characters in a virtual world.
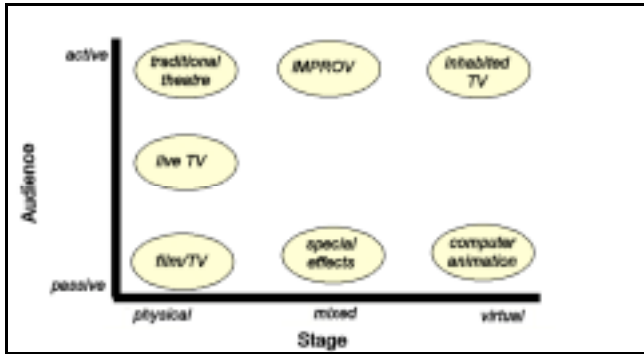

Figure 1 -- Performance classification along Audience vs. Stage dimensions
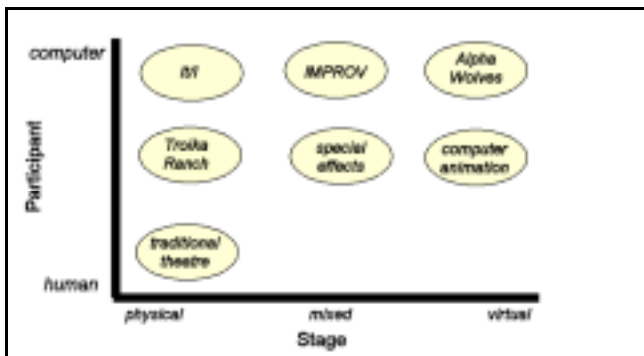

Figure 2 -- Performance classification along Participant vs. Stage dimensions

4. A VIRTUAL THEATRE SYSTEM

In this section, we describe the technical framework for a Virtual Theatre System. The framework is designed to facilitate the building of hardware and software components necessary for the execution of a virtual, distributed theatrical performance.

Much like traditional theater, the philosophy behind the virtual theatre system is motivated by teamwork and collaboration. The framework is designed with the expectation that the performance will be a cooperative effort of a number of teams, each team responsible for programming the intelligence of, or interface to, the entities participating in the performance (e.g. actors, stage personnel, audience members). The framework is designed to be general, making no assumptions as to where the performance falls in the classification

presented above except that it is expected that some virtual space is employed in the performance.

It is also expected that visual elements will be designed separately by artists using a modeling / animation software package such as Maya or 3D Studio Max.

Similar to traditional theatre, integration and communication between individual participants is used to test and debugged the production during a rehearsal process.

4.1 System Goals

When designing our virtual theatre system, we had the following goals in mind:

1) The system is designed using an object-oriented framework. This choice facilitates the division of labor and parallel development allowing each team to focus solely on objects that they are tasked to develop.

2) The general framework is defined in the context of the traditional theatrical paradigm described above. Programmers developing objects for a performance are able to interface with the system using this paradigm.

3) It is assumed that visual elements and pre-scripted animations will be created using a modeling /animation tool and should be easily importable into the system.

4.2 System Architecture

The overall architecture of the system is illustrated in Figure 3 below. The general programmer interface to the system is the Java programming language, however, many of the components of the system are written using C and C++.
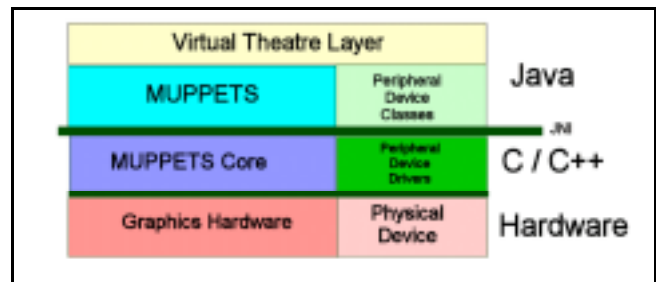

Figure 3 -- System Architecture and Components

4.2.1 The Virtual Theatre Layer

The Virtual Theatre layer defines a set of high level Java classes and interfaces corresponding to the various participants of a theatrical performance. It is at this level where the majority of the programming for individual participants is accomplished. These base classes are extended using inheritance to define specific objects, behaviors and methods for a given performance. The major classes include:

- Actor – has methods for describing position and orientation within virtual space as well as a means to react with the stage environment.

- Director – directs the behavior of an actor or actors.

- Stage Manager – has the ability to define, set, and trigger lighting and staging cues.

- Stage Elements – objects controllable by the stage manager. Can include lights, sets, props, etc.
- Audience – has the ability to set their own view as well as respond to the performance.

When designing participants that are visually present in the virtual world, it is expected that the 3D model and/or pre-scripted animation sequences will be designed outside of the programming process and imported into the virtual theatre system.

4.2.2 MUPPETS

The Virtual Theatre Layer is defined on top of MUPPETS[1] (Multi-user Programming Pedagogy for Enhancing Traditional Study), a collaborative virtual environment (CVE) originally designed for enhancing student education in the areas of programming and problem solving [17]. Like a modern massively multiplayer online game, MUPPETS provides distributed access to a shared virtual space. The core system is written in C++ and includes a complete Java programming environment. Whereas the MUPPETS core performs object rendering in C++ for efficiency, control of objects within the virtual space is implemented by writing simple programs in Java. MUPPETS also includes an Integrated Development Environment (IDE) for rapid prototyping and immediate visual feedback.   The IDE exists within the same virtual space as the objects allowing programmers to build, and test their code without having to leave the virtual space or stop the execution of the native C++ core.

The primary class for objects present in the MUPPETS virtual space is the MuppetsObject. All MuppetsObjects have a reference to some piece of geometry, either a geometric primitive or a custom model. MUPPETS has native support for the importing of models and animations from popular modeling packages thus facilitating programmatic control of externally developed models.

Being a CVE, MUPPETS exists as a networked virtual world. All objects are shared across several clients allowing for a shared experience for all those interacting in the virtual space. The complexities of message passing between MuppetsObjects are transparent to the user as all of the networking required to manage the distributed space is maintained behind the scenes. [18]

4.2.3 Peripherals

Peripherals are used by human participants to control elements on the virtual stage and represent the connection between the physical space and the virtual space. This interface can utilize common devices such as a mouse or keyboard or can employ more specialized hardware devices. For these more specialized devices, Java wrappers for the manufacturer-supplied drivers are created using the Java Native Interface (JNI) [19].   The resultant classes are then available for use by objects in either the virtual theatre or the MUPPETS layer.

In our prototype implementation, we have incorporated a single node, 6 degrees of freedom "Flock of Birds"[2] motion tracker (FOB) and a 5DT data glove.

4.2.4 Example

In order to illustrate the interplay between objects in the various levels, we present an example of a typical scenario. In our test performance *What's the Buzz?* (Which is described in more detail in the next section), one of the lead actors on the virtual stage is a swarm of bees, whose motion is determined by a basic flocking model [20].   One of the steering behaviors contributing to the motion of the bees in the swarm involves following a lead bee whose motion is controlled by a human performer equipped with a single node motion-tracking device. The objects involved in the interaction are illustrated in Figure 4.

In this diagram, classes are color-coded based upon the layer in which they exist in the architecture.   For this particular performance example, classes are required for both the Bee, which is an Actor in the virtual space, and the Flock that acts as a Director for a group of bees.   The Flock will control the motion of each bee based on the behavioral model encoded within it, which accounts for the position of a lead bee that is being controlled externally by a human participant using a motion-tracking device.   The physical motion tracker device is encapsulated in software by the MotionTrackerController object, which can be queried for the current tracker position and orientation.

In order to interoperate within the MUPPETS virtual environment, both the Actor and the Director classes are defined as subclasses of MuppetsObjects.   Note that MuppetObjects need not have a visual presence in the shared virtual space. Such is the case of a Director who is present in the space, yet participates in the performance from "backstage".
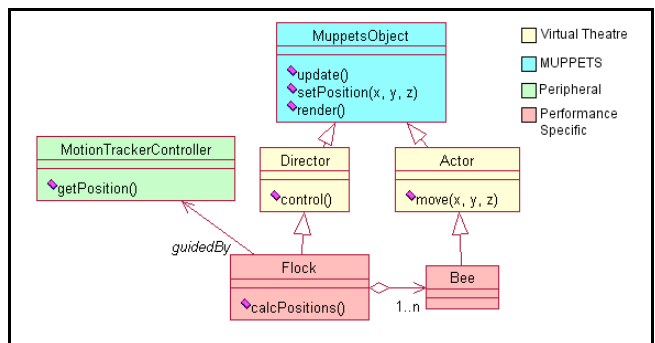

Figure 4 -- Class diagram of example object interaction

Figure 5 shows the sequence of calls made during a single update of the real time animation cycle.

---

[1] http://muppets.rit.edu

[2] "Flock of Birds" is a registered trademark of the Ascension Technology Corporation.
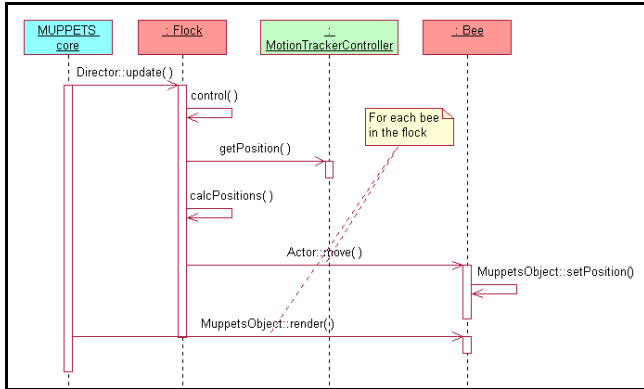
Figure 5 -- Sequence diagram for update cycle

The MUPPETS core initiates the update cycle. At each time step, the `update` method is called on all active MuppetsObjects. A Director's default action for `update` is to call the `control` method, which defines the directorial behavior for subclasses of Director. Since Flock extends Director, when `update` is called on the Flock, the Director's `update` is called, which, in turn calls the Flock's `control` method. The role of the Flock is to determine the motion of the Bees under its control. To do this, the Flock first queries the MotionTrackerController to obtain the current position of the motion-tracking device. Once known, this position is used in conjunction with the flocking model encoded by the Flock object (`calcPositions`) and each Bee is positioned by a call to `move`. Since the Bee is a subclass of Actor, when `move` is called on the Bee, the Actor's `move` method is called which, by default, will call the MuppetObject method `setPosition`, thus registering the new position of the Bee in the MUPPETS virtual space for the next time the Bee is rendered.

Note that in defining the objects required for this performance, teams can focus on the definition of the Flock and Bee object concentrating solely on redefining appropriate methods from the Director and Actor classes in the Virtual Theatre Layer. Although the infrastructure does not prohibit access to the MUPPETS layer, this access is usually not required.

5. *WHAT'S THE BUZZ?*: A DISTRIBUTED VIRTUAL PERFORMANCE

In the Spring of 2004, a set of companion courses, one given by the Department of Computer Science, and the other in the School of Design, were offered at RIT, to build the artistic and technical components for the interactive, distributed performance of *What's the Buzz?* Technical components were built on a prototype implementation of the framework described above. Artistic elements were created using Maya and Photoshop.

5.1 Story
The basic story behind *What's the Buzz?* involves a flock of bees in search of nectar. The first step in the development of the story was to identify 'characters' that could be controlled using the limited input resources available to the team. A bee was selected as the main character because it could be easily controlled with the single node FOB. The addition of the data glove allowed the single bee to evolve to a swarm of bees. The scene transitions in the story were inspired by the classic Chuck Jones cartoon, *Duck Amuck*, where Daffy Duck is presented with a set constantly changing backgrounds at the hands of a mysterious, devious animator. The evolution of the backgrounds in this animation happen behind the character and in each case it takes a few moments for Daffy to realize that he is in a new environment. In our project the progression is somewhat different. The bees face the audience with the curtains closed behind them while the scene changes. In this story, the bees navigate through a series of predefined settings, each with a distinct characteristic flower. The swarm's efforts in obtaining nectar are thwarted by each of the flowers in different ways.

In developing the story, we were constrained by the set of peripherals available to us. Although limiting, this constraint also provided a challenge to tell the story using only the data glove and a single 6 DOF motion tracker along with the tools that we could develop.

The project was designed to be improvisational with a minimal script. As a prototype, this seemed the most logical approach. It helped to limit the scope of the project to a level doable in the limited amount of time available in a ten-week quarter.

5.2 Emotional Elements
As suggested in the example presented in Section 4.2, a flocking model determines the motion of the bees in the flock. The flocking model not only incorporates basic steering behaviors such as separation, alignment, cohesion and following [21] but also includes an emotive model that drives the motion of the bees to convey one of a set of predefined emotions. [22] Thus, external control of the flock is achieved not only by specifying the position of a lead bee, but also by specifying an emotion that the motion of the flock, as a whole, should express.

Music is also used to convey emotion during the performance. An Orchestra object was developed that dynamically creates and performs a musical score based upon emotion. A generative model is used, inspired by the work presented in [23].

Note that the emotion of the flock and the emotion of the music generated by the orchestra may be controlled by two separate participants in two separate physical places. Thus, there is no guarantee that the emotion of the flock and the music will coincide. This adds further possibilities of the interplay between the two participants controlling these elements.

5.3 Visual Elements
When designing elements that will be present on the virtual stage, it was important that these visual elements have artistic merit as well as being functional models. Part of the excitement of attending a theatrical performance is enjoying the beauty of the sets and lighting. Unfortunately, many virtual pieces suffer from poor models and limited lighting.

All models used in the performance of *What's the Buzz?* were designed using Maya for modeling and animation and Photoshop for image manipulation. These visual elements were created by students in the School of Design where the emphasis is on aesthetics in addition to function.

The flowers (Figure 6) and the bees (Figure 7) were created as an exercise in a character animation course. They were rigged using skeletons and pre-scripted animated sequences were created and passed on to the programmers. A verbal description of the action and the particulars of each key frame

were provided. Both the models and the animations were imported into MUPPETS and used during the performance.
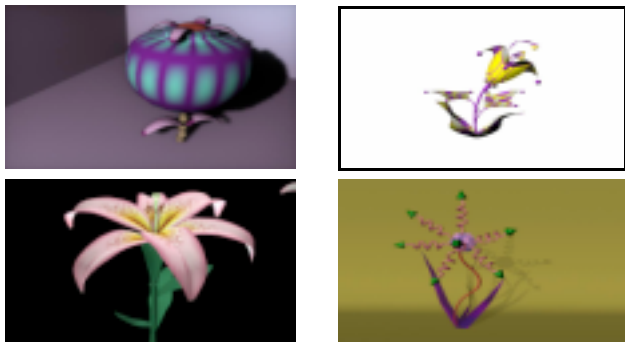

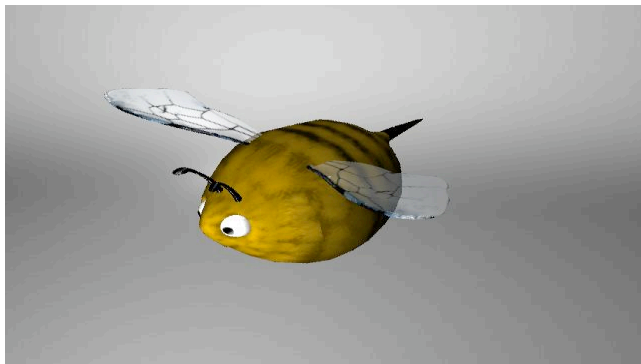**Figure 6 -- Flower Models**

Figure 7-- Bee model

The stage and stage elements for each scene were created in a beginning, modeling course. Avatars representing audience members in the virtual theatre were also created and placed in the virtual space. (Figure 8)

Creation of models for an interactive 3D space presented challenges for both the artistic and technical teams. Much emphasis was placed on creating 'lightweight' data using the fewest number of polygons possible. In addition, issues such as the creation of a model complete from all viewpoints had to be addressed. This was quite an educational experience as the artistic teams learned to work within the constraints of a real time, interactive system and the technical teams learned to adapt the models given to them in order to successfully import them into the Virtual Theatre system. This kind of teamwork only emphasizes the collaboration between creative personnel, an interaction similar to that which occurs in typical theatrical productions.

Figure 8 -- Stage Model with virtual audience

Finally, lighting for each scene was designed at an interactive session between the lighting designer and the stage manager. During this session, the placement, direction, color, and intensity of the lights for each scene was defined and saved as cues within the Stage Manager object.

5.4 Human Participants

A number of human participants contribute to the distributed performance as illustrated in Figure 9. It is assumed that each of these participants may be in different physical spaces.
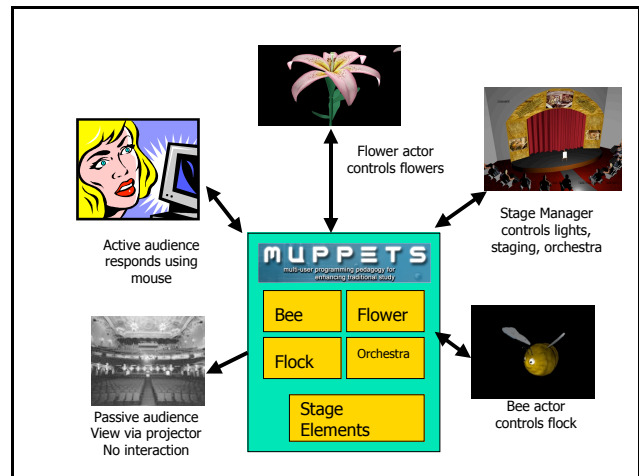

Figure 9 -- Human participation in What's the Buzz?

Each of the participants runs a copy of MUPPETS locally on a PC Workstation, sharing in the same virtual space via the MUPPET CVE. The flock and orchestra models as well as the Bee and Flowers are coded as MUPPETS Objects and are present in this shared virtual space.

5.4.1 Bee Actor

The bee actor is human actor equipped with a data glove and a 6 DOF single node FOB motion tracking device. (Figure 10) The FOB is used to direct the position of the flock in the virtual space. The data glove is used to recognize gestures, which, in turn, are used to set the emotion for the flock.

Figure 10 - Bee Actor controlling the virtual flock with a motion tracker and data glove

5.4.2 Flower Actor

Like the Bee Actor, the Flower Actor is a human who controls the flower in each one of the scenes. Each flower can be seen as a puppet, each having a set of pre-scripted animations associated with it. The role of the flower actor is to trigger and control each of the flower's predefined animations. Since the level of control for the flowers is far less than that of the flock, the interface is less complex. The flower actor controls each flower using a mouse and keyboard.

5.4.3 Active Audience

Each active audience member views the performance on his/her own monitor attached to an individual workstation. Audience interaction is performed using the mouse. Mouse motions are used to guide the view of the audience member where as the mouse buttons are used for applause and/or booing. Like in real theatre, each active audience member is assigned his or her own separate seat location within the virtual theatre.

The audience can respond to the performance in two ways. During the performance of a vignette, each audience member may applaud or boo by using the mouse buttons. Mouse clicks trigger an appropriate audio event, which is heard by all participants including the human performers who are controlling the motion of the virtual actors, as well as both the active and passive audience members. Between scenes, when the curtain is closed, easels containing posters of each of the flowers will remain on stage. Audience members choose which scene the flock should investigate next by focusing their gaze on a particular flower's poster.

5.4.4 Passive Audience

The passive audience is much like the active audience except that a passive audience member does not have the ability to interact or respond to the performance. In essence, the passive audience views the performance from a single viewpoint, much like viewing a live TV show or a recording of a live theatrical event.

5.4.5 Stage Manager

The Stage Manager is human who controls staging and lighting by triggering predefined cues. Although the stage manager shares in the same virtual stage as the other participants, he / she has the option of viewing the environment from different perspectives. The stage manager also has a simple GUI superimposed on the view of the scene used for triggering cues.

The Stage Manager is responsible for choosing and cueing scene changes. Between scenes, while the curtain is closed, the stage manager receives feedback from the audience as to which flower's poster is receiving the most visual attention. The ordering of the scenes in the piece is variable and entirely determined by the stage manager during the course of the performance. The intention is that the scenes are chosen based upon audience feedback, however, the stage manager has final say and may choose to ignore audience feedback if he wishes.

The stage manager also plays an additional role as musical conductor as he is the one that specifies the emotional mood of the music being generated by the Orchestra.

5.5 Example Interaction

Figure 11 shows a still from the "swamp scene", one of the vignettes from *What's the Buzz?* In most of the scenes, the flock, following the lead bee, controlled by the Bee Actor, navigates the scene until it locates the flower. Once located, the interplay between the flock and the flower on the virtual stage begins. In this particular scene, the flower repels the flock by spraying fire at the flock catching several of the bees aflame. Having knowledge of the minimal script, the Bee Actor knows that fire will be emitted by the flower, but being a live performance, he does not know exactly when this will happen nor does he know what action might trigger the flames. There is real acting going on here. The Bee Actor "becomes" the flock and improvises with the Flower Actor who "becomes" the flower. The Bee Actor can convey an emotion for the flock not only via the set of preset emotions provided by the flocking model but also by the motion of the lead bee. The stage manager also plays a part in the improvisation by controlling the mood of the music played by the orchestra, which is turn may affect the whims of the human actors and thus their virtual counterparts. Finally, though rudimentary (only applause and boos), the response of the audience may affect the actions of any one of the other participants, thus changing the entire interplay within the vignette.

Figure 11 -- Still from the "swamp scene"

6. CONCLUSIONS AND FUTURE WORK

Working within a theatrical context is a time proven means to tell a story. This paper studied theatrical storytelling and proposed a means and system for translating the process into a virtual space. We feel that the translation of the theatrical components and processes from a physical setting to a virtual space was effective. Keeping within the theatrical paradigm made the process of storytelling in a distributed virtual space just as intuitive and effective. In addition, "virtual theatre" proved to be a unique application for a more general collaborative virtual reality system.

"What's the Buzz?" served as an initial project and proof of concept in the area of virtual theatre at RIT. The prototype system implementation, as well as the tools developed for this production, will create the technical foundation for future productions.

Grant funding has been received to support further development of this project for the coming school year. We are already in the process of developing story and staging ideas for the next virtual production, which will occur in the spring of 2005. In this next production, we are looking to incorporate a more traditional physical theatrical space, which will work in conjunction with its virtual counterpart. Details will be published on the project Web site at http://www.cs.rit.edu/~vtheatre.

What's the Buzz? Credits

Technical Direction: Joe Geigel. MUPPETS Development: Dave Parks, Dennis Colburn, Jim DiCamillo, Jon Heise. Stage Manager: Chris Boyd, Lomax Escarment. Flocking: Kathy Law, David Rusbarsky. Networking: Roderic Campbell, Jon Heise, Ryan Kidder. Audio/Music: Ryan Becker, Dennis Colburn. Hardware: Damian Eads, James Tranovich.

Artistic Director and Story: Marla Schweppe. Actor Models: Christian Crews, Rohith Lasrado, Scott Niejadlik, Nathanael Rode, Rebecca Rogers, Aviraj Sehmi, Calvert Waller, Mary Zinno. Scenery: Alexander Cutting, Terrence Kennedy, Keith Lopez, Marla Schweppe. Stage Model: Garth McGinley, Shannon Clouston, Marie Lawlor, Cheryl Stanley.

Theatrical Director: Walter Wolf. Lighting Design: Marla Schweppe Bee Actor: Ryan Kidder. Flower Actor/Stage Manager: Lomax Escarment. Stage Crew: Damian Eads.

Poster/Program Cover: Shinhyun Kang.

REFERENCES

[1] Cohen, R*., Theatre, 6th Edition*,. McGraw-Hill, 2002.

[2] Sauter, W., *The Theatrical Event: Dynamics of Perfomance and Perception*, University of Iowa Press, 2000.

[3] Matsuba, S.N., Roehl, B., "Bottom, Thou Art Translated: The Making of VRML Dream", *IEEE Computer Graphics and Applications*, 19 (2), March/April 1999, 45-51.

[4] Lewis, M, "Bowen Virtual Theatre", *Proceedings of the SIGGRAPH 2003 conference on Web graphics*, (San Diego, CA, 2003)

[5] Slater, M.,Howell, J., Steed, A., Pertaub, D-P, Gaurau, M., "Acting in Virtual Reality", *Proceedings of the third international conference on Collaborative virtual environments*, (San Francisco, CA, September 2000), 103 - 110 .

[6] Farley, K, "Digital Dance Theatre: The Marriage of Computers, Choreography and Techno/Human Reactivity", *Body, Space and Technology,* 3(1), 2002, (http://www.brunel.ac.uk/depts/pfa/bstjournal/3no1/journal3_1.htm)

[7] Sparacino F; Davenport G; Pentland A, "Media in performance: Interactive spaces for dance, theater, circus, and museum exhibits", *IBM Systems Journal*, 39(3-4), 2000, 479-510.

[8] Pinhanez, C.S, Bobick, A.F., It / I: A Theatre Play Featuring and Autonomous Computer Graphics Character", *Proceedings of the sixth ACM international conference on Multimedia: Technologies for interactive movies*, (Bristol, United Kingdom, September 1998), 22-29

[9] Moltenbrey, K, "Staged Magic", *Computer Graphics World*, December 2003, 31 – 34.

[10] Tomlinson, B, Downie, M., Berlin, M., Gray, J., Lyons, D., Cochran, J., Blumberg, B., "Leashing the AlphaWolves: mixing user direction with autonomous emotion in a pack of semi-autonomous virtual characters*", Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, (San Antonio, TX, 2002), 7-14

[11] Perlin, K., Goldberg, A., "Improv: a system for scripting interactive actors in virtual worlds", *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (New Orleans, LA, 1996), 205 – 216.

[12] Benford, S., Greenhalgh, Craven, M., Walker, G., Regan, T., Morphett, J., Wyver, J., "Inhabited Television: Broadcasting Interaction from with Collaborative Virtual Environments", *ACM Transactions on Computer-Human Interaction*, 7 (4), December 2000, 510-547.

[13] Zellner, D. "Definitions and Directions of the Theatre" in *Theatre in Cyberspace*, S. Schrum, ed. Peter Lang Publishing, 1999, pp 21-27.

[14] Kauffmann,S., "Notes on Theatre and Film" in *Focus on Film and Theatre*, Hurt, J, ed., Prentice-Hall, 1974.

[15] Mateas, M.,. "An Oz-Centric Review of Interactive Drama and Believable Agents." Technical Report CMU-CS-97-156, Department of Computer Science, Carnegie Mellon University, Pittsburgh, USA, 1997.

[16] Benford, S., Greenhalgh, C., Reynard, G., Brown, C., Koleva, B., "Understanding and Constructing Shared Spaces with Mixed-reality Boundaries", *ACM Transactions on Computer-Human Interaction*, 5 (3), September 1998, 185-223.

[17] Phelps, A, Bierre, K.,Parks, D., "MUPPETS: multi-user programming pedagogy for enhancing traditional study", *Proceeding of the 4th conference on Information technology education*, (Lafayette, IN, 2003), 100-105.

[18] Phelps, A, Parks, D., "Fun and Games: Multi-Language Development", *ACM Queue*, 1 (10), February 2004, 46-56.

[19] Liang, S., *Java Native Interface: Programmer's Guide and Specification*, Addison-Wesley Professional, 1999.

[20] Reynolds, C. W. "Flocks, Herds, and Schools: A Distributed Behavioral Model", in *Computer Graphics*, 21(4) (SIGGRAPH '87 Conference Proceedings) 25-34.

[21] Reynolds, C.W., "Steering behaviors for autonomous characters". In *Proceedings of Game Developers Conference*, pages 763–782, 1999.

[22] Law, K, "Emotion in Flocking", Masters Project, Rochester Institute of Technology, Department of Computer Science, June 2004.

[23] Birchfield, D., "Generative model for the creation of musical emotion, meaning, and form", *Proceedings of the 2003 ACM SIGMM workshop on Experiential telepresence* (Berkeley, CA, 2003), 99 – 104.