Branching Storylines in Virtual Reality Environments for Leadership Development

Andrew Gordon, Mike van Lent, Martin van Velsen, Paul Carpenter, and Arnav Jhala

The Institute for Creative Technologies, The University of Southern California
13274 Fiji Way, Marina del Rey, CA 90292 USA
{gordon, vanlent, vvelsen, carpenter}@ict.usc.edu, ahjhala@unity.ncsu.edu

Abstract

Simulation-based training is increasingly being used within the military to practice and develop the skills of successful soldiers. For the skills associated with successful military leadership, our inability to model human behavior to the necessary degree of fidelity in constructive simulations requires that new interactive designs be developed. The ICT Leaders project supports leadership development through the use of branching storylines realized within a virtual reality environment. Trainees assume a role in a fictional scenario, where the decisions that they make in this environment ultimately affect the success of a mission. All trainee decisions are made in the context of natural language conversations with virtual characters. The ICT Leaders project advances a new form of interactive training by incorporating a suite of Artificial Intelligence technologies, including control architectures, agents of mixed autonomy, and natural language processing algorithms.

Computer-based Leadership Development

The success of organizations and their missions can often be attributed to the leadership aptitude of its members. Many of the traits and skills needed to be an effective leader are learned only through experience. Organizations such as the US Army view strong leadership ability as essential to their success, and have devoted substantial training resources toward leadership development. The broadening of the types of missions that soldiers are assigned and the new responsibilities of junior officers and enlisted soldiers has increased the need for leadership development within the Army, encouraging them to explore the use of computer-based training to supplement more traditional classroom-based instruction.

To date, most of the Army's computer-based training involves the use of constructive simulations, where soldiers practice the planning and execution of missions in simulated environments. In these simulations, predictive models of physical and character behavior are used to calculate the effects of trainee decisions, allowing for low-stakes practice and evaluation of a soldier's performance. However, the skills associated with successful leadership are tightly coupled with the subtleties of human behavior, involving intellectual flexibility, adaptability, and a capacity for operating in ambiguous and stressful environments. While some research efforts are beginning to incorporate more comprehensive models of human behavior in Army training simulations (e.g. Rickel et al., 2002), the fidelity of these behavior models have not yet achieved the level necessary for training across the broad range of leadership skills.

Hill et al. (2003) developed an alternative to constructive simulations for computer-based leadership development that used a case-analysis model of instruction. In their TLAC-XL application, trainees analyzed a challenging leadership scenario presented as a short video production, and engaged in an after-action analysis with a virtual mentor and virtual characters from the fictional storyline. While scripted cinematic sequences demonstrated that fictional scenarios could be constructed that promote deep and relevant discussion about leadership, the case-analysis approach did not allow the soldiers being trained to practice making leadership decisions for themselves in a learn-by-doing environment.

The ICT Leaders Project

The ICT Leaders Project, a collaboration between the University of Southern California's Institute for Creative Technologies and Paramount Pictures, is a research effort aimed at allowing trainees to practice making leadership decisions in the context of complex fictional scenarios realized in a virtual reality environment. In our first prototype, the user plays the role of a US Army captain commanding a company of soldiers on a peacekeeping mission in Afghanistan. The situation, which parallels the story described in Hill et al. (2003), involves providing security for a food distribution operation. Rather than relying on scripted video, however, the ICT Leaders project presents the story and fictional scenario in a virtual reality environment based on a commercial game engine, where cinematic scenes are interwoven among conversations with animated virtual soldiers and civilians in the environment.

The user experience in the ICT Leaders application is structured around a series of scripted cinematic scenes rendered in the virtual environment of the game engine. These scenes move the storyline forward and present challenging leadership problems where a decision needs to be made by the user. These problems are always presented

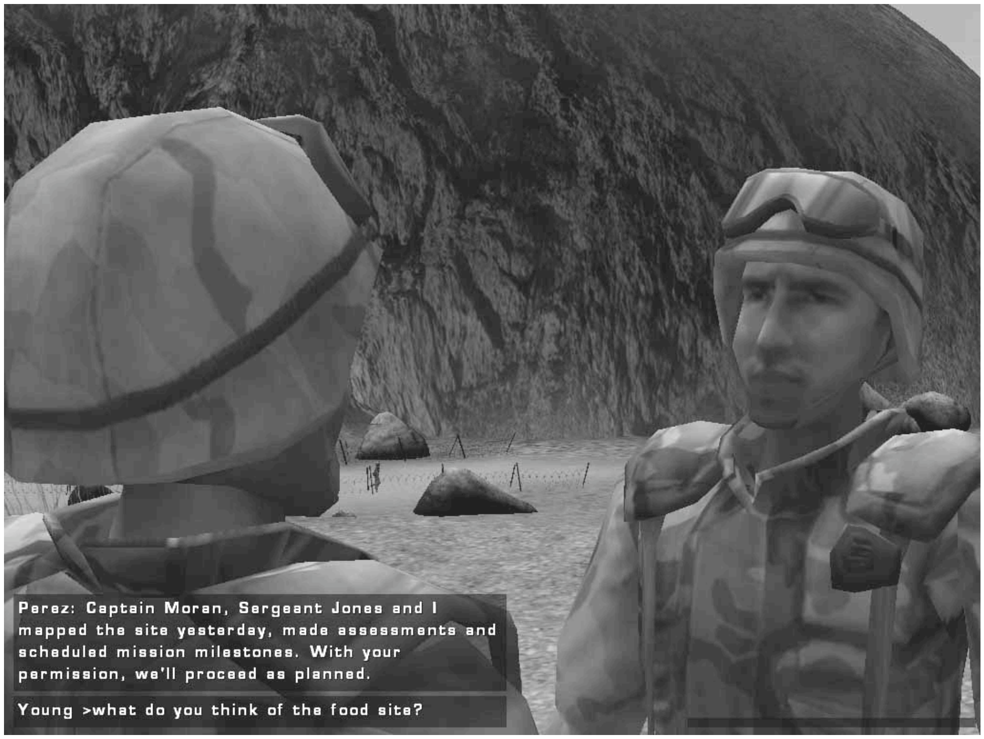

Figure 1. The ICT Leaders Project

to the user by storyline characters, and the primary user interaction involves text-based conversations with these characters. The user has the opportunity to raise questions and make comments concerning the problem, but in order to move the storyline forward the user must present a decision to the virtual character. The choice that the user makes has a direct effect on how the storyline unfolds, where different choices cause the experience to follow different paths in a branching storyline structure. In all, there are 63 decisions that make up the branch points of our current application prototype, and each decision in the storyline has been motivated by a specific leadership point. These leadership points were identified through directed interviews with experienced company commanders. Each point challenges the expectations that novices have when adopting a leadership role.

The ICT Leaders project uses a commercial game engine, Epic Game's Unreal Tournament 2003, to create the virtual environment for the user experience. Using the standard "mod" editor that ships with this product, we have created a system that is very far removed from the original first person shooter design. This approach has allowed us to take advantages of advances in technology within the computer game industry and collaborate effectively with other research groups that use this same platform for their research prototypes.

A screen shot of the ICT Leaders application is presented in Figure 1.

AI Application Advancements

The ICT Leaders project draws on previous work in a number of fields to extend two technology areas: story-based training simulations and video game engines. The ICT Leaders application is the result of a novel combination of these technologies as well as innovative applications of artificial intelligence (AI) techniques. By combining the branching storyline structures used in story-based training simulations with a rich virtual world realized using a commercial game engine, the ICT Leaders application immerses users into a rich training context while keeping the experience focused on pedagogical training objectives. This combination capitalizes on the strengths of both approaches, improving the level of immersion of story-based training simulations and the degree of pedagogically relevant user interaction in video games. The ICT Leaders application incorporates a story management agent that controls the current state of the user experience within this branching storyline structure, and monitors the users decisions to support virtual mentoring and after-action review.

The ICT Leaders application utilizes AI-controlled non-player characters, divided into two categories with different levels of AI fidelity for each. Background characters move through the world and perform actions autonomously based on a library of behaviors. The behavior of foreground characters is more tightly constrained by the underlying branching storyline. Using natural language, the user interacts with these foreground characters via a text-based interface, enabled by AI natural language processing techniques. This is contrast to current commercial video games, where the standard approach to incorporating natural language is to have the user select an utterance from a short menu of possibilities. This standard approach is undesirable from a leadership development perspective, as the user will not have a list of choices when faced with real-life leadership dilemmas. By incorporating contemporary statistical text classification techniques, the ICT Leader application gives users the challenge of deciding what to say while retaining the ability to respond appropriately given the context of the situation.

The next five sections of this paper describe each of the various AI-related technologies used in the ICT Leaders application to create an effective training application for leadership development within the US Army. First, the branching storyline structures that underlie the user experience are discussed. Second, a mixed-fidelity approach to AI-based virtual characters is described. Third, we describe the statistical natural language processing techniques used to support natural language dialogue with the virtual characters in the environment. Fourth, we discuss a capacity for virtual mentoring in the application to support the leadership training objectives. Fifth, we describe the multi-agent control architecture that manages the flow of interaction between heterogeneous software components. A preliminary evaluation of the ICT Leaders project is then briefly discussed, followed by some directions for future research within the context of this project.

Branching Storylines

The user experience of the ICT Leaders application can best be compared to the classic *Choose-your-own-adventure* paperback novels published by Bantum Books (Packard, 1979), but where decisions are made in the context of text-based conversations rather than by multiple choice. While branching storylines of this sort have been used in computer-based training applications for many decades, the ICT Leaders project employs a particular method of authoring branching storylines called *Outcome-driven Simulations* developed at Northwestern University's Institute for the Learning Sciences in the 1990s (Gordon, 2004). As an alternative to traditional constructive simulations, Outcome-driven Simulations are branching storylines where every decision that a trainee makes in the training experience is motivated by a specific training point, and where the consequences of every action will lead the trainee to pedagogically important outcomes.

The methodology for authoring Outcome-driven Simulations begins with the identification of training points through the collection of anecdotes from experienced practitioners of the skill to be trained (Ferguson et al., 1992; Johnson et al., 2000). For the ICT Leaders project, 63 anecdotes were collected from experienced US Army captains who had been company commanders. Each of these anecdotes was analyzed to identify its underlying point, which we formulated as expectations that were violated by the experience as retold by the practitioner (as in Schank & Abeleson, 1995). These points were then recast as decisions to be made in hypothetical situations, where the choice that a person would take hinges on whether or not they believe the point expressed in the anecdote. These decision formulations were the basis for each branch point in the resulting branching storyline, and each hypothetical situation was instantiated in the context of a fictional situation that grounds the user experience. In the ICT Leaders project, this fictional situation is a security mission for a food distribution operation in Afghanistan to be conducted by a US Army company of soldiers.

After instantiating decision formulations into the fictional context, an overall branching storyline was constructed by connecting the choices available in each decision formulation with the situations presented in another, where every path through the branching storyline is a coherent narrative. While many possible branching storyline topologies are possible in Outcome-driven Simulations, the 63 decision formulations in the ICT Leaders project were arranged as an ordered sequence of five trees, representing five chapters in a larger narrative. Within each chapter, 12-14 decisions are arranged with exactly two choices per decision and with an average of 4 different decisions for any given path through the chapter. Each of the last choices in the chapter leads to the first decision in the subsequent chapter, except for the fifth chapter, which has many unique final outcomes. This structure allows the branching storyline in the ICT Leaders project to have roughly 20 trainee decisions for any given route through the full application.

Rather than presenting the trainee with a list of choices at every decision point, as was done in previous Outcome-driven Simulations, users of the ICT Leaders applications indicate their choices in the context of natural language conversations with the virtual storyline characters. After being presented with a situation by a virtual character, the user is allowed to type free-text statements, either to ask questions, make comments about the situation, or to indicate a choice that they wish to make. To support this natural language interaction, statistical machine learning techniques are used to identify the features of the user text that are most predictive of the class of the user utterance, causing the story to move forward in the case of choices and informative responses from the virtual characters in the case of questions or comments.

An example of the type of user interaction supported by this approach follows, in which a user must make a leadership decision regarding the appropriateness of the current mission plan. Presented as text here, this example is realized as a fully animated sequence of camera and virtual character actions in the virtual world that we have created within Unreal Tournament 2003.

Situation: *The user, playing the role of Captain Young, the company commander, is engaged in discussion with his First Sergeant, Jones, and Command Sergeant Major*

Pullman, who accompanying the Captain during the mission on a related Civil Affairs task.

Pullman: Captain, I don't want to go stepping on any toes, but I was looking over your terrain. I'm sure this site got chosen because of the road access. But it's sure got some problems. I would encourage you to bump up your security perimeter eastward, and put more distance between the wire and the road. […]

Jones: Captain, myself along with the XO and the platoon leaders, we put a lot of thought into this configuration. Not saying the CSM has an inferior plan, but company personnel collaborated very closely in strategy and tactics.

Pullman: Captain, your First Sergeant heads a great team, but if you make these changes, it'll pay off. Just give it some thought. [exits]

Jones: Captain, would you like changes made to the security perimeter?

User: `Jones, don't you think that these changes are a good idea?`

Jones: Captain, I'm sure the CSM will tell you plans get changed all the time. I just think there was some real good collaboration on the prep for this operation.

User: `Jones, are you suggesting that making these changes might be bad for the morale of the soldiers?`

Jones: Speaking freely, sir, I think letting the CSM waltz in and dictate tactics is a bad idea. This company should be rewarded for its good work here.

User: `Yes, but the CSM has a lot of experience. Let's see if we can incorporate his advice into our plan.`

Jones: I'll get right on it, Captain. This company prides itself on its flexibility.

Having made the decision to incorporate the advice of the CSM, the conversation ends and the next decision situation in the branching storyline structure is presented to the user. This particular dilemma was motivated by an anecdote from an experienced Army captain who emphasized the importance of sticking with the plans that were generated collaboratively by those who are actually going to execute them. The choice made the user in this example runs counter to this teaching point. By tracking the choices that are made within the Outcome-driven Simulation, users can be prompted to reflect on these choices in an after-action review component at the end of the experience.

Mixed-fidelity AI Agents

There are two classes of agents in ICT Leaders: scripted agents and autonomous agents. Scripted agents are essential to the progression of the story. The autonomous agents fill the role of the background characters. Although the background characters are not essential to the story, they provide a much richer, realistic environment, which is necessary to properly engage the user in the evolving situation.

Both types of agents are written using Unreal Tournament's built-in object-oriented scripting language, UnrealScript. Complex agent behaviors were realized by inheriting and modifying the existing bot code that is used to control the non-player characters in the original commercial version of the game. Such primitive behaviors include path planning, turning to face the direction the agent is moving, and obstacle avoidance, among many others. Although the high-level behaviors of the existing bots did not meet our specific needs, they were easily replaced with new behaviors while leaving the more primitive behaviors intact. Using inheritance, we were able to concentrate on the development of higher-level behaviors.

Scripted Character Agents

Scripted agents were developed to play the major character roles in the story with which the user can communicate. Like Hollywood screen actors, these agents are required to perform actions at pre-defined locations at specific times in order to present the fictional situation as an engaging narrative. These actions include the animated delivery of pre-recorded audio clips, gesture and gaze control in single-party and multi-party dialogs, and interacting with the virtual physical environment through scripted body animations. Although many of the primitive bot behaviors were retained, the majority of these character actions were carefully scripted during the authoring process.

In authoring the ICT Leaders application, one of the major problems we dealt with concerned the exact timing of the scripted behavior of these agents, particularly with respect to the delivery of pre-recorded audio clips. An audio clip that is played even half a second too soon may overlap with the previous line that was delivered, while playing it a half of a second too late results in too much of a pause in-between the spoken lines. Problems in timing can easily be compared to watching a movie where the soundtrack is mis-cued with the film. To avoid the distractions that this would cause, fine adjustments in timing of scripted action were made on a subjective basis.

Other timing problems arose due to variation and uncertainty within the virtual environment of the commercial game engine. For example, it is sometimes difficult to predict how long it will take an agent to arrive at a destination in the virtual world due to variations in the path selected by the low-level AI path-planning algorithm or due to variations in processor speed of the computer running the application. In this case, we chose a solution that is analogous to traditional filmmaking practices, where we kept the scripted agents located nearby, just out of view of the camera, whenever they were not part of the current scene. By keeping them physically close, the variance in the timings due to navigating the environment were kept to a minimum.

Autonomous Character Agents

Autonomous agents were used to give a sense of life to the virtual environment. These agents are analogous to the extras that are used on a movie set. They do not have spoken dialog lines to deliver and are not required to be at any specific location at any specific time during the course of any given scene. In the ICT Leaders project, a wide assortment of autonomous characters were necessary to convincingly visualize a food distribution operation being supported by company of 120 US Army soldiers. Without the portrayal of soldiers patrolling ridge tops, working on fencing surrounding the food distribution site, or standing guard at security checkpoints, the experience would be greatly diminished.

As with movie extras, the primary responsibility of fully autonomous agents in the ICT Leaders applications is to avoid being a distraction from the main action of the scene. Some level of sophistication was necessary, however, to avoid the appearance of mindless repetitive action. For example, a soldier working on fences for crowd control in the background of a scene can be distracting if they are simply looping over a single animation without ever moving around the environment. Accordingly, behavioral control algorithms were authored for a wide range of classes of background characters in the ICT Leaders application, including the following:

- Soldiers working on fences for crowd-control
- Military Police soldiers guarding security checkpoints
- Soldiers patrolling ridgelines
- Soldiers simply moving around the environment
- Afghan civilians in the area of the food distribution site

Each type of background character in the application is completely autonomous, controlled by customized behavior specification tailored to the specific role of the agent in the story. For example, soldiers working on fences for crowd control select among one of the animations corresponding to working on a fence, loop over the animation for a random amount of time, move to another location along the layout of the fencing in the terrain model, and then repeat this behavior. Other roles played by autonomous agents are programmed similarly.

As is the case with the scripted agents, the low-level behaviors of the autonomous agents are inherited from the existing bot code. This includes path-planning behaviors used to move agents around the virtual environment. Although path nodes for navigation are placed manually in the terrain model, it is impossible to predict the exact paths that agents will take. Occasionally one of the autonomous agents moving around the environment will walk closely behind a scripted agent in a scene with which the user is engaged in conversation. Since care was taken when placing the path nodes, the path the autonomous character takes does not interrupt the scene. Instead, a much more positive impact is realized, where the seemingly close interaction between scripted and autonomous characters makes the environment seem much more alive to users.

Natural Language Processing

The natural language interaction component of the ICT Leaders project is supported by statistical natural language processing algorithms, where each decision point in the branching storyline is coupled with one text classifier trained to route user inputs into one of some number of utterance classes. We employed a Naïve Bayesian classification algorithm (George & Langley, 1995) implemented in the WEKA open source machine learning toolkit (Witten & Eibe, 1999). Training data and user inputs are encoded as large feature vectors, where each dimension in the vector corresponds to one of all of the single words (unigrams) and adjacent pairs of words (bigrams) seen in the training data. These feature vectors are constructed without using stop-list filters, without truncating the feature space, by ignoring punctuation and variation in case, and using features counts for feature values.

Hill et al. (2003) employed the same text classification approach that we use in the ICT Leaders project, and reported percentage accuracy rates averaging 65.72% given small training data sets averaging 194 instances across 13 classes. To compensate for the small training data sets that are likely to be collected and tagged over the course of the ICT Leaders project lifecycle, we are experimenting with an approach that has trainees classify their own utterances in the context of using the application. Instead of immediately selecting the class of highest confidence as the best classification of the user utterance, the ICT Leaders application collects the top three most likely classifications, and presents to the user a paraphrase of each class in a pop-up list. The user is then asked to select among this list of three choices which paraphrase best characterizes the utterance they have typed, or to select none-of-the-above. In selecting the most appropriate paraphrase, the user interface informs a centralized language processing service that performs the classification of the verification selection, and the original user text is then treated as additional training data in support of the verified class. It remains an open research question as to the effectiveness of this approach, and whether user verification of classifications can be legitimately used as additional training data to improve the statistical text classification performance. An evaluation of the relative merits of this approach will be conducted after enough user-verified data has been collected.

Virtual Mentoring

The ICT Leaders application includes a virtual mentor that occasionally interjects suggestions to assist the user. The virtual mentor is not a character in the scenario but acts as an assistant offering comments and suggestions. The virtual mentor's comments appear as text in a pop-up window in the lower left corner of the screen shown in

Figure 1. While these interjections risk breaking the user's immersion in the scenario, their timing is carefully designed to occur only when the user is already having difficulties that threaten the immersion or when the user repeatedly fails to understand the intended leadership points. These two instances highlight the two ways the virtual mentor guides the user.

First, the virtual mentor steps in when the user enters a phrase that the natural language processor cannot match to any of the pre-defined choices. This could be caused by the user not understanding the decision before him, the entering of a phrase that is not easily classified into one of the pre-defined utterance classes, or when the appropriate utterance isn't in the three classes with the highest confidence. In the current system the virtual mentor has no means of determining the specific cause of the problem. To compensate, the virtual mentor covers all possibilities by interjecting a pre-authored comment giving the user clear guidance as to the allowable choices and expected phrasing of each. While this approach quickly and easily gets the user back on track, there is a risk that users will take advantage of these interjections in order to quickly identify what they need to say to move the story forward, without considering the leadership issue at hand. Future work may lead to the development of a virtual mentor that can determine the underlying cause of the problem and offer more tailored advice that doesn't uncover too much about the available choices.

The second situation in which the virtual mentor steps in to assists the user only occurs when the user is going through the experience for the second time (or any subsequent time). If the user makes the choice that is counter to the point of the original leadership anecdote at a decision point in one run and then in a later run reaches the same decision point and makes the same decision, again countering the leadership anecdote, the virtual mentor interjects to steer the user towards the choice that supports the leadership anecdote. The fact that the user has repeatedly made the choice countering the leadership point suggests the user doesn't understand the point and is in danger of reinforcing a misconception. The virtual mentor interrupts the user's choice with an explanation of the leadership point and the supporting choice and gives the user a chance to take back the decision that was just made. By breaking the flow of the experience and forcing the student to think about the decision, the virtual mentor can prevent the user from reinforcing a decision that counters the intended leadership point. Still, users have the option of ignoring the advice of the virtual mentor and sticking with their choices if, for example, they are trying to explore specific parts of the branching storyline.

Control Architecture

A service-based approach was used in the ICT Leaders application to coordinate a variety of heterogeneous software components that were originally designed to operate as stand-alone systems. By implementing each

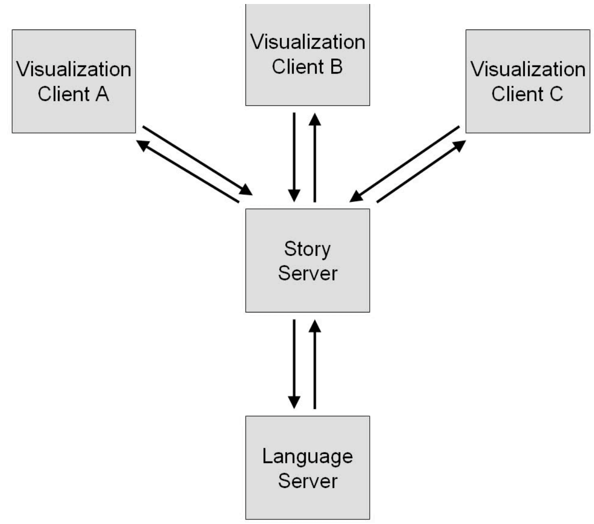

Figure 2. Distributed Control Architecture

component as a service or wrapped to become a service, the collection of these components could serve each of the functional requirements of the system. Three primary services comprise the ICT Leaders application. First, a visualization service presents the user with a view onto the virtual world and provides a user interface for interaction on this virtual stage. Second, a story management service controls the ordering of scenes presented to users by tracking their location through the branching storyline structure that underlies the experience, taking into account the decisions that users make. Third, a language processing service is used to interpret the natural language text given by users, providing to the story management service an indication of the best classification of each statement. Figure 2 diagrams the flow of information among these three service types.

The use of blackboard-style control architectures has become more prominent in heterogeneous agent-based systems, particularly in interactive multimedia applications (Vaysburd & Birman, 1997; Johanson & Fox, 2002; Hill et al., 2003). However, the linear flow of information and a lack in need of asynchronous event handling in the ICT Leaders application has allowed us to return to a more traditional functional control architecture.

Visualization Service

Visualization services in the ICT Leaders application provide each of the functions necessary to present to the user a view of the virtual environment. To reduce production costs and to capitalize on advancement in computer graphics within the commercial game industry, the ICT Leaders project used a modified version of Epic Games' Unreal Tournament 2003 as its visualization environment. The most important factor in selecting this software platform for visualization services was the ability to modify nearly all aspects of the commercial version of the game using a object-oriented scripting language and editing tools that ship with this commercial product. The

Liquid Narrative group at North Carolina State University had adopted this software platform for the development of their Mimesis system (Young et al., 2004), and we used the tools developed by this research group to control simulation events within Unreal Tournament 2003 from processes running outside of the game engine.

Story Management Service

The selection of events to be visualized within the virtual world is handled by a dedicated story management service running outside of the game engine. This service, which simultaneously handles multiple visualization clients, tracks the users' locations within the underlying branching storyline structure. The primary information that is communicated to the story management service is the text that is typed by users in the course of conversations with virtual characters in the visualization service. The story management service identifies the appropriate classifier to be used to process this text, a service that is then delegated to a separate language processing service. The results of this language processing service are then passed through the story management service to be verified by the user in the visualization service, where the user is asked to choose which classification best paraphrases the statement that they made. When this result is given to the story management agent, the location of the user within the branching narrative is updated, the next scripted scene is triggered, and the process repeats itself.

Language Processing Service

The language processing service has the responsibility of classifying text passed to it from the story management service to identify which of a fixed set of expected utterance classes best characters the user's statement. Each of these classes corresponds to a possible decision or virtual character response in the underlying branching storyline. After a user confirms which classification best represents what was originally stated, this confirmation is used as training data for future classifications by the language processing service.

Interaction Flow

All communication among the participating software services is encapsulated in XML messages, and is designed to be an expandable notation for a wide range of narrative events. The expressive nature of these formalisms are particular important for the visualization service, where the XML being parsed serves as a generic visualization representation. Each time the story management service communicates with a visualization client, a high-level scene description, formatted in XML, is being passed. By authoring an expressive XML schema for these messages, the ICT Leaders applications supports modularity among the services, which may be switched with new designs as our research efforts continue.

Preliminary Evaluation

Two types of evaluations are scheduled for the ICT Leaders project. Formative evaluations with members of our target training population (junior US Army officers) will be conducted in early 2004 to assess the usability of the ICT Leaders application. Summative evaluations, aimed at determining the effectiveness of the application for the purpose of leadership development in the US Army, will be conducted with members of our target training population in the summer of 2004.

Preliminary formative evaluations of the ICT Leaders application have been conducted with students and an instructor of a university Army ROTC program, where a group of undergraduates collectively used the application under the supervision of the instructor. These evaluators navigated the user interface and storyline without difficulty, and tended to discuss the most appropriate choice given a decision situation with each other rather than engaging the virtual characters in prolonged dialogs. These group discussions were supported by subtle interventions on the part of the instructor, yielding a effective mix of both human and virtual mentoring.

Future Work

In the production of interactive virtual environments for training applications there is a need for the development of automated and semi-automated tools for designers and creative artists that aid them in generating immersive and affective worlds. Following the production process used in developing the branching storyline for the ICT Leaders application, generalized tools could be built for rapid development of similar training applications. This project also motivates research in the development of computational models of techniques and rules followed by Hollywood filmmakers who have been successful in visual storytelling. At each step in the production process of this application we have identified potential areas for future work.

Creating the Branching Storyline

In writing the story, tools that help a writer quickly build and visualize sequences of interactive and non-interactive action make the overall experience more coherent for the user. Other research groups have developed systems for automated generation of stories in interactive virtual environments (Magerko & Laird, 2003; Young et. al., 2004). These automated story generation systems could be used for the generation of complex scenarios for training applications.

Autonomous Characters

For providing ambience to the virtual world, characters in the virtual world have behaviors that make them look believable. Tools for defining such behaviors and agents would save a lot of time in the production process. Semi-automated tools that convert script directives to character actions could be developed.

Autonomous Camera Control and Lighting

Training soft skills like leadership require the virtual environments to have expressive lighting and shot composition for communicating the importance and consequences of the decisions that are made by the user. With the increase in geometric complexity of the training environments there is a need to incorporate work in the field of expressive autonomous lighting (Seif El-Nasr, 2003) and camera control (Jhala, 2004). We are developing semi-automated tools that could be used by designers of such training applications with automated lighting design and shot sequences that could be structured to the preferences of the user.

Acknowledgments

This paper was developed with funds of the Department of the Army under contract number DAAD 19-99-D-0046. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the Department of the Army.

References

Ferguson, W., Bareiss, R., Birnbaum, L., & Osgood, R. 1992. ASK systems: an approach to the realization of story-based teachers. *Journal of the Learning Sciences* 1, 95-134.

George, J. & Langley, P. 1995. Estimating continuous distributions in Bayesian classifiers. *Proceedings of Eleventh conference on uncertainty in Artificial Intelligence*. 338-345, San Mateo, CA: Morgan Kaufmann.

Gordon, A. 2004. Authoring Branching Storylines for Training Applications. *Proceedings of the Sixth International Conference of the Learning Sciences (ICLS-04), Santa Monica, CA.*

Hill, R., Douglas, J., Gordon, A., Pighin, F., & van Velsen, M. 2003. Guided Conversations About Leadership: Mentoring with Movies and Interactive Characters. *Proceedings of the Fifteenth Innovative Applications of Artificial Intelligence Conference*, pp. 101-108.

Jhala A. 2004. Intelligent Cinematic Camera Control for Dynamic Narratives, *Master's Thesis, Department of Computer Science, North Carolina State University.*

Johanson B. & Fox A. 2002. The Event Heap: A Coordination Infrastructure for Interactive Workspaces. *Proceedings of the 4th IEEE Workshop on Mobile Computing Systems & Applications*

Johnson, C., Birnbaum, L., Bareiss, R., and Hinrichs, T. 2000. War stories: harnessing organizational memories to support task performance. *Intelligence* 11(1), 16-31. New York: ACM Press.

Magerko, B. & Laird, J. E., 2003. Building an Interactive Drama Architecture, *1st International Conference on Technologies for Interactive Digital Storytelling and Entertainment*, Darmstadt, Germany, March 24 – 26.

Rickel, J., Gratch, J., Hill, R., Marsella, S., Traum, D., & Swartout, W. 2002. Toward a New Generation of Virtual Humans for Interactive Experiences, *IEEE Intelligent Systems*, July/August 2002, pp. 32-38.

Schank, R. & Abelson, R. 1995. Knowledge and memory: The real story. In R. Wyer (ed) Advances in Social Cognition, Vol. 8. Mahwah, NJ: Lawrence Erlbaum Associates. pp 1-86.

Seif El-Nasr, M. & Horswill, I. 2003. Real-Time Lighting Design for Interactive Narrative. *Proceedings of the 2003 International Conference on Virtual Storytelling*.

Vaysburd A. & Birman K. 1997. Building Reliable Adaptive Distributed Objects with the Maestro Tools. *Proceedings of the Workshop on Dependable Distributed Object Systems, OOPSLA'97, Atlanta, GA.*

Witten, I. & Frank, E. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. San Francisco: Morgan Kaufmann.

Young, R. Michael, Riedl, Mark, Branly, Mark, Jhala, Arnav, Martin, R.J. and Saretto, C.J., *An architecture for integrating plan-based behavior generation with interactive game environments*, to appear in *The Journal of Game Development,* March 2004.