BRINGING THE SCHOOLHOUSE INSIDE THE BOX – A TOOL FOR ENGAGING, INDIVIDUALIZED TRAINING

Brian Magerko, Ph.D.
Games for Entertainment and Learning Lab
Michigan State University
417 Communication Arts Bldg., East Lansing, MI 48824
magerko@msu.edu

Brian S. Stensrud, Ph.D.
Lisa Scott Holt, Ph.D.*
Soar Technology, Inc.
3361 Rouse Road, Suite #175, Orlando, FL 32817
{stensrud, lholt}@soartech.com

ABSTRACT

The Interactive Storytelling Architecture for Training (ISAT) is designed to address the limitations of computer games for advanced distributed learning (ADL) and to fully realize the potential of games to become engaging and individualized training environments. The central component of the ISAT architecture is an intelligent *director agent* responsible for individualizing the training experience. To achieve this, the director tracks the trainee's demonstration of knowledge and skills during the training experience. Using that information, the director plays a role similar to that of a schoolhouse trainer, customizing training scenarios to meet individual trainee needs. The director can react to trainee actions within a scenario, dynamically adapting the environment to the learning needs of trainee as well as the dramatic needs of the scene. This paper describes a prototype implementation of the ISAT architecture in the combat medic training domain, with an emphasis on the design of the director agent.

1. INTRODUCTION

Current research on playing commercial computer games shows evidence of cognitive, social, and perceptual benefits (Gee 2003; Gee 2005; Johnson 2005; Prensky 2006). These benefits provide ample support for the creation of games that are specifically targeted for training domains.

Games provide engaging digital worlds that can take advantage of the processing power of a computer to offer a highly interactive experience. They can be created for learning experiences that are too costly to recreate in the real world or that benefit from adding interactivity to the learning task (Hill et al. 2003; Winn et al. 2003; Revolution 2005; Hazmat Hotzone 2005; Magerko et al. 2005). However, as learning games emerge as a new educational tool, they still suffer from the same weakness as other media: a lack of individualized aid like that of a teacher or tutor in a one-on-one situation (Beal et. al 2002). Without appropriate oversight, computer games can result in negative training, since no human tutor is present to guide the trainee and provide the most appropriate training experiences. In the freedom of the game environment, the student is able to perform many actions, some of which are appropriate for learning and others that are not.

Individualized tutoring for computer-based instruction has previously been explored in the realm of *intelligent tutoring systems* (Koedinger et al. 1997). Intelligent tutoring systems (ITSs) employ a cognitive model of the student interacting with the learning system. As a student progresses through the material, the system executes *knowledge tracing*, which builds a model and generates hypotheses about the student's proficiency in the skills being taught, and presents material that addresses the student's weaknesses. ITSs then use the model to tailor problem sets and teaching tips to that particular student's perceived pedagogical needs. As a student interacts with the learning environment, the system executes *model tracing*, which means it continually forms a hypothesis about what strategies the student is using to solve a problem. If a student commits an error, then the system can offer helpful advice that is tailored to that student's specific needs.

While intelligent tutoring has proven effective in non-game based learning applications, little work has examined how student models can be used to tailor computer games to provide more engaging and effective training experiences (Riedl and Stern 2006; Johnson et. al 2004). This paper presents the Interactive Storytelling Architecture for Training (ISAT), an architecture that provides tailored training experiences within an interactive story experience (Magerko et. al 2005). ISAT builds on our previous work in interactive drama to provide trainees with a dramatic experience that is tailored to their needs, based on the choices they make in the game environment and on ISAT's model of trainee knowledge (discussed in detail in Section 4.3). By tailoring the game based on dramatic and pedagogical needs, the system attempts to maximize *the effectiveness and the engagement* of the training experience. The following Sections describe in-depth how ISAT works and how the main component of the architecture, the director agent, makes decisions based on these inputs to alter a trainee's learning experience.

2. RELATED WORK

2.1 Pedagogical Agents

Agent-based technologies have been used to realize not only interactivity but a range of support and guidance in game- and simulation-based training systems. The software agent is designed to interact (perceive and act) with the trainee in the environment. Trainees learn and practice new skills in a virtual world, and the instructional system reacts through the use of artificial intelligence.

An intelligent agent can act within or outside the space of the environment. If the agent acts within the space of the environment, the interactions reflect actions in the world and contribute to the realism of the experience. The agent can also outside the space of the environment, serving a different role (e.g., that of an instructor or guide). The interactions of pedagogical agents provide explicit tutorial guidance in training systems (Rickel & Johnson 1999; Johnson 2001), and these developments have demonstrated the pedagogical power of using software agents in knowledge-based learning environments (Moreno, Mayer & Lester 2000).

Intelligent agents can interact with the learner directly (e.g., as a coach or learning companion) or indirectly by altering the environment. In most cases, pedagogical agents have been represented explicitly in the simulation environment as an animated virtual tutor or as another helpful character. The explicit representation of a tutor-like character may be appropriate for some training but it has the serious drawback of compromising the realism of the training scenario. In addition, interrupting the execution of a task to give guidance or feedback can interfere with learning in some cases.

2.2 The Tactical Language Trainer

The Tactical Language Trainer (TLT) is a game-based language trainer that teaches trainees proper verbal, body language, and cultural skills for different languages (Johnson et. al 2004). The first language domain approached is Levantine Arabic for military training. The TLT uses an explicit pedagogical agent that interacts directly with the trainee. The TLT builds a learner model to track the trainee's proficiency in the skills being taught. This model informs an explicit pedagogical agent (e.g., an aide to the trainee in the game) that offers guidance and help when needed according to the model. This approach incorporates elements of intelligent tutoring (e.g., knowledge tracing) in a full-scale game for learning. However, it does not take full advantage of the storytelling potential of games seen in interactive drama applications.

2.3 IN-TALE

IN-TALE is an ongoing project in employing interactive narrative techniques for training (Riedl and Stern 2006). The agent in IN-TALE does not interact directly with the trainee but instead coordinates training goals in the game environment to deal with unexpected trainee actions that may occur via replanning algorithms. It uses a reactive planning language to create a *social simulation* of non-player characters that can both provide a rich world for a training exercise as well as characters that can be commanded by the director to help fulfill training goals.

IN-TALE employs techniques found in other interactive drama systems, namely MIMESIS and Façade, to create an engaging experience that adapts as the player makes choices and executes actions in the world (Young et. al 2004; Mateas and Stern 2003). While IN-TALE's strength does lie in its flexibility to adapt the training content, it does not have a model of trainee knowledge or skill that can be used to influence the decisions made during these adaptations. The use of such a model can be a powerful tool for scenario adaptation, as discussed in Sections 5.1 and 5.3.

3. ISAT OVERVIEW

3.1 Architectural Design

The Interactive Storytelling Architecture for Training (ISAT) combines storytelling and instructional strategies to provide realistic and engaging *individualized* training that can transfer to improved performance in the real world. ISAT provides individualized training through the *real-time adaptation of stories*. We are developing general techniques and components (an architecture) that will enable existing and new training systems to dynamically adapt content to support training goals and increase trainee engagement. The ISAT architecture is focused on an internal and imperceptible agent, the *director*, in a constructive simulation environment.

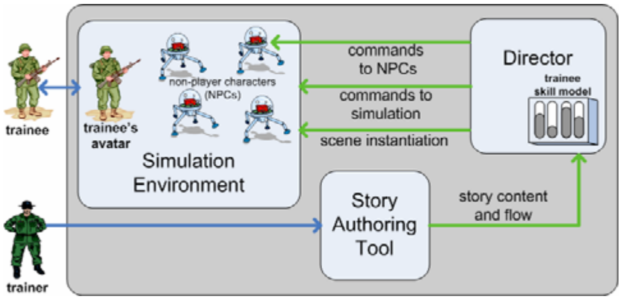

Fig. 1. ISAT architecture diagram.

ISAT builds on the notion of a pedagogical agent that guides user experience. The director agent in ISAT does not play an explicit instructional role but instead interacts with the trainee indirectly through the environment. The director does not provide direct pedagogical guidance and feedback to the trainee; rather, it structures the environment to give the trainee appropriate experiences. The director commands the environment to respond in ways that are dramatically and pedagogically relevant for the individual trainee.

Because the director is not represented explicitly, the trainee's sense of engagement and responsibility for decision making is naturally and effectively supported. Furthermore, as the trainee's skills develop, the director takes fewer and less frequent remedial actions, purposely fading out the instructional support and helping the trainee develop the ability to act independently.

ISAT's design (see Figure 1) includes the trainee, the human trainer who authors the training content, the virtual simulation environment populated with synthetic or non-player characters (NPCs), and the director agent. The trainer authors story content with both dramatic and instructional principles in mind. The director is responsible for managing the trainee's experience in response to the training scenario content, the trainee's actions in the simulation environment, and the skill model of the trainee, which tracks his mastery across a set of domain skills. The director can influence the trainee's experience by giving synthetic characters certain actions, by altering the environment (e.g., spawning characters or environmental sounds), or through selection and performance of scenario content.

3.2 Environment

Our current prototype system is implemented using the 91W10 Tactical Combat Casualty Care Simulation (TC3), a combat medic training system developed by Engineering and Computing Simulations, Inc. (ECS). The TC3 combat medic trainer is based on the same systematic approach for Basic and Advanced Trauma Life Support that is used to train Emergency Medical Technicians (EMTs). The TC3 is an immersive 3D simulation in which trainees treat casualties in tactical situations that are dictated by the principles of Tactical Combat Casualty Care.

Fig. 2. Treatment menus in TC3 trainer.

The trainee sees the simulated world from a first-person point of view, and can move around in a 100-ft. radius circle of terrain representing a courtyard in an Afghani village. As a result of an improvised explosive device, the trainee encounters various casualties which he must prioritize and treat. Trainees can implement a variety of medical procedures by selecting them from menus associated with different body parts (see Figure 2).

4. ARCHITECTURAL DETAILS

4.1 Trainer's Role as Author

One of the major differences between interactive training systems and interactive drama is the inclusion of a human trainer or knowledge expert that is directly needed within the authoring process. ISAT includes a story authoring tool, called *Scribe*, which allows the trainer to create scenarios using a visual representation of ISAT's story representation (described in Section 4.5). This bypasses the common problem of domain experts and educators being removed from the design process because they do not have the relevant skill set to directly encode their own expertise (e.g. lacking programming skills). Details on Scribe's design can be found in Medler and Magerko (2006).

4.2 Integrating the Director with the Architecture

Functionally, the ISAT director operates by inducing behaviors and actions from characters and objects within a training scenario. As these actions are ultimately performed within the simulation, it is necessary that the director be tightly coupled with that simulation. At the same time, it is desirable for the simulation to be a modular and replaceable component within the ISAT architecture. While the director maintains a significant amount of domain-specific knowledge that it uses to make decisions in the world, the mechanics by which those decisions are made is a constant and independent of the domain implementation.

The director agent is written in *Soar* (Laird et al. 1987), a cognitive architecture used to develop knowledge-intensive agents capable of operating in real-time environments. Given appropriate knowledge about the training domain, the agent is encoded with a set of actions it can perform within a simulation of that domain. Such actions might include:

- Creating (*spawning*) characters and objects
- Inducing movement
- Conducting dialog
- Changing the environment (e.g., weather)

The director must also be aware of events occurring within the scenario that it does not control, including:

- Where the trainee and other characters are in the world, their orientation and velocities
- When a trainee performs and action, the specifics and immediate effects of that action
- Simulation time

Once an exhaustive set of actions is specified, each action is implemented as a function call in the director. Within the simulation, then, a set of handlers are implemented that read in each function call and translate that call to a set of actions in the simulation required to implement that call. Similarly, a set of handlers are created to detect relevant events in the world and report them in a sanitized format to the director. By implementing these mappings in the simulation, we allow the director agent to remain impartial to the environment.

4.3 Skill Model Representation

The *skill model* is a representation of the trainee's proficiency levels in a set of domain-specific skills as calculated and perceived by the director. As we will discuss in Section 5, the skill model allows the director to alter the training scenario and its contents to match the needs of the trainee.

The skill model keeps a record of trainee proficiency in each of a set of domain-specific skills. Each record includes both a skill *score* and a skill *weight*. The skill score represents the trainee's proficiency, while the skill weight represents the degree to which the trainee has been tested in that skill. A low score with a low skill weight, for instance, might be representative of a novice trainee making his first attempt executing that skill, while a low score with a high skill weight likely indicates a trainee who has learned the skill incorrectly or is having trouble mastering it.

During an exercise, the director monitors the actions of the trainee and specifically his attention to the various skills. If a certain skill is performed incorrectly, the director will lower the appropriate skill score. As skills are demonstrated, the director will increase the weight of each indicating that the trainee has had more opportunities to demonstrate proficiencies. Skill scores with higher weights fluctuate more gradually as the director has more samples by which to calculate a score. Future iterations of the skill model design will include the affects of recency and decay on the skill values (e.g., testing a student's skill a few scenes ago is substantially different than a few days ago).

4.4 Integrating the Director with the Authoring Tool

The ISAT director is responsible for the performance of content within training scenarios. While the director is encoded with a significant amount of knowledge about the training domain, the details and mechanics of individual scenarios are generated using the *Scribe Authoring Tool*, introduced in Section 4.1. Using this tool, a human trainer can construct a scenario (or *story*) for his domain that will achieve his desired training objectives. It is the director who is then responsible for executing the components of that scenario within the training environment.

Details of ISAT story representation are provided in Section 4.5. Stories are encoded using XML, which the director is capable of loading directly into its working memory for use at execution-time. Skill model values can also be generated by the tool and passed to the director at execution-time for trainees who have previously been engaged in the training.

4.5 Story Representation

ISAT's story representation is directly inspired by our previous representation work in the Interactive Drama Architecture (Magerko 2006). As described in Section 4.4, the scenario is hand-crafted by a human trainer. The main story element in this representation is a *plot point*. *Plot points* are equivalent to scenes in a screenplay, where a short period of time passes in a specific location. They are connected through ordering constraints to create a partially-order of transitions between them. A collection of partially-ordered plot points is a complete *scenario*.

A plot point P is defined as *P* → *{C, A, D, E, S}*. *C* is the set of first-order *preconditions* that logically describe facts about the world to set the stage for a plot point. These preconditions describe how the world need be configured in order for the plot point's *events* to occur. For example, in one of the initial scenes in ISAT's current implementation with the TC3 trainer, the trainee arrives on the scene of a bomb explosion in an open marketplace as part of a quick response force (QRF) squadron. There are only civilian casualties. The rest of the squad quickly secures a perimeter as the trainee is ordered to tend to the wounded in the marketplace. There is one casualty motionless on the ground, and two wounded, obviously alive and injured. This initial setup for the scene is logically described in the following first-order logic clauses:

```
C1: SpawnInjured(civilian, dead, SP011,
    civ011)
C2: SpawnInjured(civilian, {no_leg,
    no_forearm, burn}, SP012, civ012)
C3: SpawnInjured(civilian, SP013,
    civ013)
```

C2 is read as a goal for the director to spawn a live civilian at spawn point `SP012` with unique identifier `civ012` that has one of three different injuries: a missing leg, a missing forearm, or a burn. Preconditions may also contain variables that are instantiated by the director at run-time, which is a director capability currently being designed.

A is the set of *actions* that describe what is true once the events in this plot point are finished. Predicates are shared between *preconditions* and *actions* just as one would expect in a planning language. This affords us the option of applying planning algorithms to the construction and adaptation of scenarios for dealing with problematic player behaviors (Riedl and Stern 2006) in the future.

D is a Boolean descriptor that is *true* if plot point *P* has been performed and is essentially *done*. This variable is initially set to *false* and is changed once all member of *A* for that plot point have been executed.

E is the set of *events* that occur within a plot point. *Events* are happenings in the world initiated by either the player or the non-player characters. *Events* provide the trainer with the means to create temporally-constrained scripts for the atomic actions that the Trainer intends to occur during a plot point. In other words, if a plot is similar to a scene in film, then *events* describe the atomic dramatic actions (or "beats"*)* that occur within that scene.

Events are described with the same predicates used in *C* and *A* but with an additional temporal logic to describe how individual *events* should overlap. There are three different kinds of temporal constraints: *finite*, *random*, and *relative*. *Finite* events occur within a specific amount of time in seconds (e.g., the ambush happens 60 seconds after the beginning of the plot point). The r*andom* constraint specifies a duration between 0 to X amount of seconds, where X is specified by the Author (e.g., random(30) means the event could start at any time between 0 to 30 seconds from the current point). *Relative* is a temporal constraint that relies on other events to be triggered. For example, event C could be given a constraint to begin immediately after the events A and B have occurred.

S is the set of skills tested by *P*. Each plot point is annotated by the Trainer with a set of possible skills that plot point could teach. For example, our plot point about an ambush on the medic's squad would test such skills as "prioritizing casualties," how to "find temporary fighting positions," and how to properly "extract a casualty" that is under fire. How the director agent uses *S* to select plot points is discussed in Section 5.3.

5. DIRECTOR CAPABILITIES

5.1 Skill-based Direction

A central role of the ISAT director is to assist the trainee in real-time by providing timely and personalized feedback relevant to the task at hand. To do this, the director must not only have knowledge of the correct execution for skills tested but also some notion of the trainee's proficiency at those skills and awareness of what the trainee is doing in the scenario. Domain knowledge is encoded within the director *a priori*. The director monitors the trainee's actions and presence through its connection with the simulation environment. The skill model, introduced in Section 4.3, provides the director with a hypothesis of the trainee's proficiencies.

Skill-based direction allows ISAT to individualize the training experience by identifying and responding to specific errors. The director responds to errors by executing actions within the environment. The director's choice of action will depend on not only the nature of the skill error but also the state of the trainee skill model at the time of the error. The director's actions can be *direct* or *indirect* depending on the needs of the situation. An example of direct skill-based direction would be inducing the virtual squad leader in the training environment to yell at the trainee for making the mistake and ordering him to fix or un-do it. If the director were to instead adversely modify the state of one of the trainee's teammates (e.g., worsen his condition due to the incorrect application of a tourniquet), that would be a type of indirect skill-based direction.

The ability to choose specific, skill-based direction strategies based on the state of the trainee's skill model allows the director to *scaffold* and *fade* his feedback. For example, while the director might interject direct in-game feedback to a novice trainee who has made an error in a certain skill (like having a character shout at him about the error), it might use a more indirect approach in reaction to the same error by a more experienced trainee. Access to the skill model affords the director a great deal of flexibility when choosing a particular feedback method.

5.2 Reactive Direction

One of the pitfalls of immersive training environments is that students are given the opportunity to execute actions which are peripheral to or otherwise outside the training experience. In the TC3 combat medic training simulation, for instance, trainees are free to navigate the world. While this certainly makes for a realistic experience, it does not always serve to achieve the objective of good training.

Reactive direction is a technique by which ISAT can prevent the trainee from navigating outside the training space. To implement this technique, the director must first detect when the trainee is in a situation where he does not have access to training content (e.g., is in an area where there are no casualties). Then, the director must choose and implement a strategy to subtly (e.g., not using a 'talking head') guide the trainee back into the content of the story. An example of this would be to direct an associated character (e.g., a teammate) to walk over to a trainee who has wandered outside the story content (e.g., into a portion of the map where no story is taking place) and point him in the correct direction.

5.3 Scene Selection and Performance

Though the ISAT Director is given the set of scenes (or plot-points) from Scribe as described in Sections 4.1 and 4.5, it is the director's responsibility to select and perform those plot-points in an appropriate order so as to personalize the scenario to fit the needs of the trainee.

The selection of plot-points is controlled by two components: dramatic and pedagogical relevance. Plot-points are selected for dramatic relevance to ensure that the story presented to the trainee is both believable and engaging. Plot-points are similarly selected for pedagogical relevance so that the training content delivered to the trainee more closely matches what will most benefit him.

In their description, plot-points are annotated with the set of skills that they test. When the director chooses a plot-point, it matches the list of tested skills from each plot-point with the current state of the skill-model. If the trainee is weak in a large set of skills that happen to be tested by a particular plot point, the director may be inclined to choose that point depending on its dramatic relevance to the story. ***Fig. 3*** illustrates this process.

To perform a selected plot point, the director simply executes commands to fulfill the unfulfilled preconditions and the associated events for that selected point.

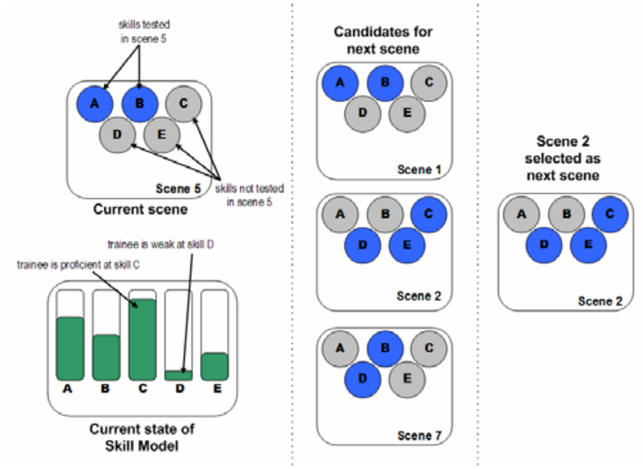

Fig. 3. Scene selection in ISAT.

6. FUTURE WORK

6.1 Evaluation of ISAT

To evaluate the overall training effectiveness of ISAT, we are currently planning two separate evaluation studies to be conducted in spring of 2007. The first study will focus mainly on the implicit actions of the director. The second will focus more directly on the achievement of training objectives in the 91W10 combat casualty course.

The first study will be conducted using medical students from Michigan State University. Although the medical students will not know the specific principles of combat casualty care, they will be familiar enough with the medical treatment options to interact meaningfully in the simulation.

The director manipulates scenes according to a pedagogical approach based on training goals and a dynamic model of the trainee's skill. These manipulations of the trainee's experience are intended to be seamlessly integrated so that the trainee is unaware of the director's presence. Interviews and questionnaires will be administered to focus on this aspect of the training system. Do trainees recognize instructional interventions within the environment? Do the director's actions disrupt the immersive nature of game play or interrupt the trainee's execution of tasks?

The second study will be conducted using actual military trainees. Students in the 91W10 course at Ft. Sam Houston will be using the ISAT-free TC3 trainer as part of their course. We will arrange for a subset of the students to use the ISAT version of the TC3. By comparing TC3 use with and without the director's pedagogical scenario management, we can assess the director's added value to the training system.

To determine the effectiveness of the training system with and without the director, the evaluation will be closely tied to the 91W10 course training objectives. The training objectives will be used to develop an assessment instrument. The study will use a pre/post control group design to obtain a measure of learning outcomes. Learning gains for the two groups can then be compared. Post-training interviews with both groups will shed some light on the effects of ISAT components and will also be used to further probe trainees' understanding of the training material.

More detailed process and usability data will be collected through individual observations during training. These observations will provide important information about how trainees interact with the system and where they have difficulty. Think-aloud protocols will be used to reveal trainees' thinking and probe their knowledge as it develops during training. This process data will not only be very useful for improving the training, but will also help to explain any learning gains measured by the pre-/post-tests. Because individual observations are costly and think-aloud methods can interfere with cognitive tasks, these evaluation methods will only be used on a small subset of participants.

6.2 Unified Trainee Model

Our work with ISAT and previous work on IDA has focused on modeling the player's skill, knowledge, and behavior (Magerko et. al 2005; Magerko 2006). While each of these modeling dimensions has obvious benefit, we will engineer a combination of these three approaches into a single, unified trainee model. This unified model will combine uses of the individual models to provide a broad, robust trainee model that will allow the director to execute more intelligent actions.

The major benefits we anticipate from this unified model approach are: 1) more intelligent prediction, 2) more accurate skill models, and 3) more intelligent predictive and skill-based direction. Current prediction does not make use of the information hypothesized in the skill or knowledge models. When predicting trainee behavior, information about what knowledge the trainee has of the world and of the trained tasks could help inform the predictive quality of the model.

6.3 Semi-Autonomous Agents

Our current work on ISAT is fairly non-commital about the behaviors of the non-player characters used in the scenarios. ISAT assumes that the characters can receive direction, but makes no commitment to whether or not they are autonomous and, if they are, how they transition from their autonomously-selected goals to new goals (Assanie 2002). ISAT does also not explore different modalities for interacting with these characters, such as through natural language (Johnson et. al 2004). Therefore, future work will examine how directable, semi-autonomous characters can be incorporated into the ISAT architecture (Blumberg and Galyean 1995; Magerko 2006; Riedl and Stern 2006).

CONCLUSIONS

ISAT is a novel approach to interactive training that employs techniques from two disparate fields, interactive drama and intelligent tutoring systems. The combination of these techniques provides an individualized experience for the trainee that maximizes both the engagement as well as the effectiveness of that experience. The *Scribe Authoring Tool* provides a useful means for a trainer using ISAT to visually encode training scenarios without the need for specialized programming skills. Future work includes the evaluation of the architecture as applied to the combat medic training domain.

ACKNOLWEDGEMENTS

This material is based upon work supported by RDECOM-STTC under Contract No. N61339-05-C-0142. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of RDECOM-STTC.

REFERENCES

Assanie, M. 2002: Directable Synthetic Characters. *AAAI Spring Symposium Series: Artificial Intelligence and Interactive Entertainment*, Palo Alto, CA.

Beal, C. R. et al. 2002: Intelligent Modeling of the User in an Interactive Environment. *AAAI Spring Symposium*

Series: Artificial Intelligence and Interactive Entertainment, Palo Alto, CA.

Beisaw, J. et al. 2002: General, Maintainable, Extensible Communications for Computer Generated Forces. *Proceedings of the Eleventh Conference on Computer Generated Forces and Behavioral Representation*, Institute for Simulation and Training, Orlando, FL, 563-570.

Gee, J., Ed. 2003: *What Video Games Have to Teach Us about Learning and Literacy*. Palgrave McMillan.

Gee, J., Ed. 2005: *Why Video Games are Good for Your Soul: Pleasure and Learning*. Victoria, Australia, Common Ground Publisher.

Hazmat Hotzone 2005: S.O. Studios, Pittsburgh, PA.

Hill, R. et al. 2003: Virtual Humans in the Mission Rehearsal Exercise. *Embodied Conversational Agents*.

Johnson, S. 2005: Your Brain on Video Games. *Discover Magazine*, July, 39-43.

Johnson, W. L. 2001: Animated Pedagogical Agents for Education Training and Edutainment. *International Conference on Advanced Learning Technologies*.

Johnson, W., Marsella, L., S., and Vilhjálmsson, H. 2004: The DARWARS Tactical Language Training System. *Proc. of the 26th Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, Orlando, FL.

Jones, R., Koss, F., Nielsen, P. and Taylor, G. 2000: Communication with Intelligent Agents. *Proc. of the 22nd Interservice/Industry Training Systems and Education Conference (I/ITSEC)*, Orlando, FL.

Koedinger, K., Anderson, J. R., Hadley, W. and Mark, M. 1997: Intelligent Tutoring Goes to School in the Big City. *International Journal of Artificial Intelligence in Education*, **8**, 30-43.

Laird, J. E., Newell, A. and Rosenbloom, P. 1987: Soar: An Architecture for General Intelligence. *Artificial Intelligence*, **33**, 1-64.

Magerko, B. 2006: Player Modeling in the Interactive Drama Architecture. Ph.D. Thesis, Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, MI.

Magerko, B. et al. 2005: Improving Interactive Training through Individualized Content and Increased Engagement. *Proc. of the 27th Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, Orlando, FL.

Mateas, M. and Stern, A. 2003: Façade: An Experiment in Building a Fully-Realized Interactive Drama. *Game Developer's Conference*, San Francisco, CA.

Moreno, R., Mayer, R., and Lester, J. 2000: Life-Like Pedagogical Agents in Constructivist Multimedia Environments: Cognitive Consequences of Their Interaction. *Proc. of the World Conference on Educational Multimedia, Hypermedia, and Telecommunications (ED-MEDIA)*, Montreal, Canada, 741-746.

Prensky, M., Ed. 2006: *Don't Bother Me Mom – I'm Learning*, Paragon House.

Revolution 2005: Education Arcade, Cambridge, MA.

Riedl, M. and Stern, A. 2006: Believable Agents and Intelligent Scenario Direction for Social and Cultural Leadership Training. *15th Conference on Behavior Representation in Modeling and Simulation,* Baltimore, MD.

Rickel, J. and W. L. Johnson 1999: Animated Agents for Procedural Training in Virtual Reality: Perception, Cognition, and Motor Control. *Applied Artificial Intelligence,* **13**, 343-382.

Young, R. M., Riedl, M. O., Branly, M., Jhala, A., Martin, R. J., Saretto, C. J. 2004: An Architecture for Integrating Plan-based Behavior Generation with Interactive Game Environments. *Journal of Game Development*, **1**, 51-70.