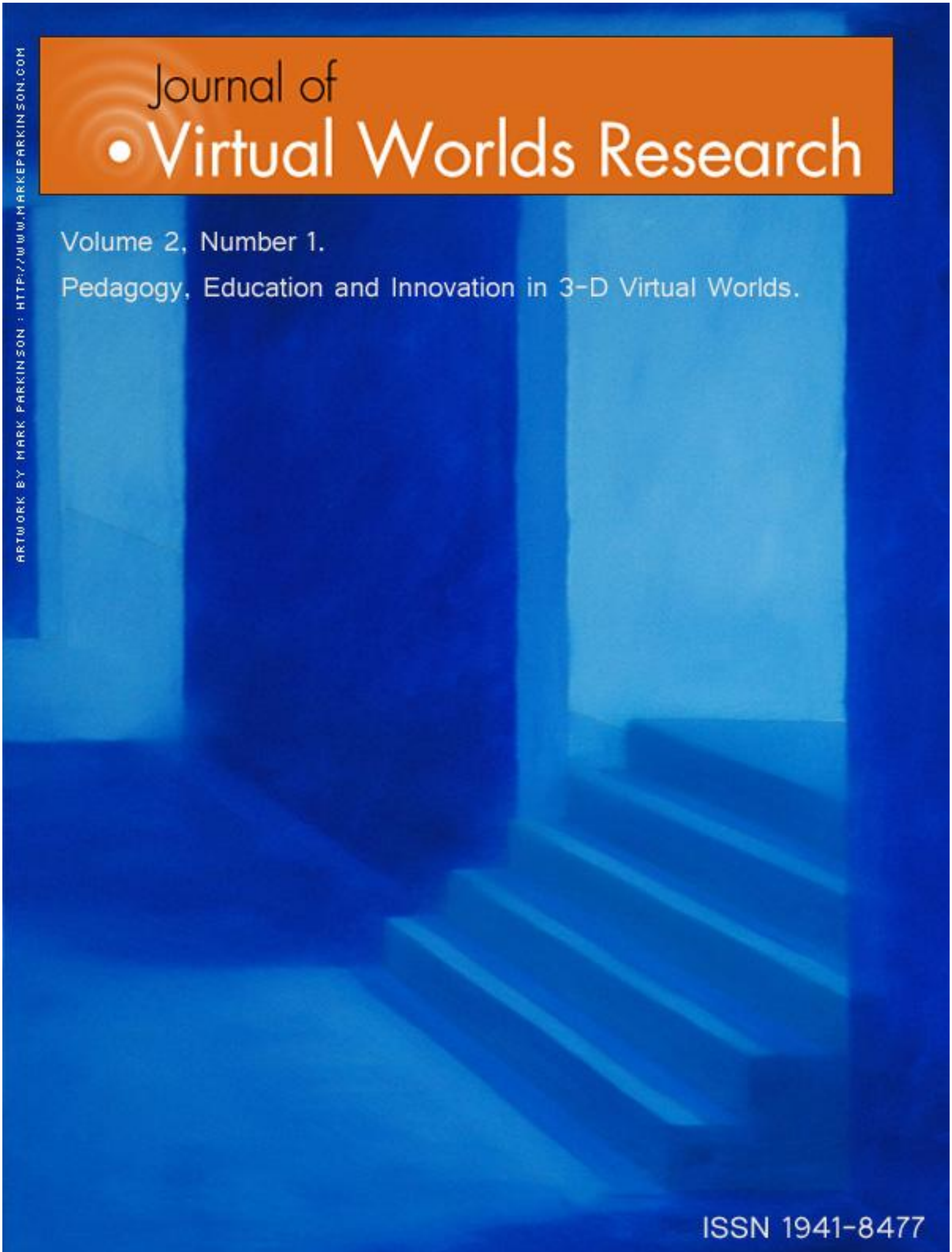
Journal of Virtual Worlds Research

Volume 2, Number 1.

Pedagogy, Education and Innovation in 3-D Virtual Worlds.



ISSN 1941-8477

Vol. 2. No.1

“Pedagogy, Education and Innovation in 3-D Virtual Worlds”

April 2009

This issue was sponsored, in part, by the Singapore Internet Research Centre, the Department of Radio, TV & Film at the University of Texas at Austin, and the Texas Digital Library Consortium.

The Journal of Virtual Worlds Research is owned and published by the Virtual Worlds Research Consortium, a Texas non-profit corporation.
(http://vwrc.org)

Journal of Virtual Worlds Research
jvwresearch.org ISSN: 1941-8477

Vol. 2. No.1
ISSN: 1941-8477
"Pedagogy, Education and Innovation in 3-D Virtual Worlds"
April 2009

An integrated framework for simulation-based training on video and in a virtual world

By David Chodos; Parisa Naeimi and Eleni Stroulia, University of Alberta, Canada

Abstract

Becoming a skilled professional requires both the acquisition of theoretical knowledge and the practice of skills relevant to one's profession. When learning by doing, students consolidate their knowledge of domain-specific facts by applying them as necessary to accomplish the tasks involved in their profession. Simulation-based learning methods are a family of methods that enable this learning mode. New computer related technologies, including high performance networking, high definition displays, distributed multiplayer game engines, and virtual worlds, bring new opportunities for simulation-based learning methods and systems. In this work, we describe our software framework for specifying simulation-based lesson plans and their implementations on two different platforms: a video based tool and a virtual world environment. We discuss the software architecture of the system, illustrate its functionality with an example lesson on how to conduct oneself in corporate interviews, outline our plans for experimental evaluation, and argue for its usefulness in today's efforts to creatively use virtual worlds for educational purposes.

Keywords: training; standard patient; annotated video; workflow.

An integrated framework for simulation-based training on video and in a virtual world

By David Chodos; Parisa Naeimi and Eleni Stroulia, University of Alberta, Canada

"For the things we have to learn before we can do them, we learn by doing them." This quote from Aristotle's "Nichomachean Ethics" succinctly summarizes the compelling intuition behind learning-by-doing pedagogical methods. Most of us learn best "on the job" by trying things out and reflecting on our own experiences, even sharing our experiences with other people who may also learn from them. Learning by doing enables us to consolidate our knowledge of domain-specific facts as well as practice and refine the skills necessary to accomplish the related tasks. This is why post-secondary education institutions include in their curricula capstone project courses, internships, co-op terms, and practicum courses, in the hopes of better preparing their students to professionally conduct themselves in their actual practice after school.

Simulations provide another means for enabling learning by doing. A simulation retains the essential elements of the real situation, abstracts away the less relevant details, and places the learner in the role of one of the participants. In return, the learner brings to bear all his or her knowledge relevant to the situation, to make the decisions required of his or her role, and, generally, to act in accordance with this role. New technologies give rise to many more opportunities for simulation-based learning, in which a person is placed in a scenario or situation and is directly responsible for the changes that occur as a result of his or her decisions. Recent developments in software, multiplayer games, the internet and virtual reality have created richer, more life-like learning experiences for more learners.

Virtual worlds, in particular, are emerging as a platform with huge potential for teaching and learning, in general, and simulation-based training, in particular. Many universities have established a presence in Second Life, including University of Florida, Princeton, Vassar, the Open University (UK), Harvard, Australian Film Television and Radio School, Stanford, Delft University of Technology, and AFEKA Tel-Aviv Academic College of Engineering, just to name a few. At the same time, researchers and educators are grappling with a number of questions around the adoption of virtual worlds for educational purposes. These questions range from how to best exploit this new technology, to how to adapt teaching pedagogy to a virtual classroom, to deciding what types of learners might benefit the most from learning what type of subject matter in them.

Among the most important questions include developing new pedagogical theories and models of how students learn in virtual worlds and corresponding methods for assessing the effectiveness of education in these worlds. Freitas and Neumann (2009) recently proposed an exploratory learning model, adapted from Kolb's experiential learning model (1984), to guide teachers in rethinking how they teach in 3D and immersive spaces and how to "choreograph" the learning sequences that include peer interactions and exchanges. In our work, we are interested in developing a software framework that will enable the parallel development of simulation-based lessons on different platforms. In this manner, we hope to effectively reuse the effort that goes in the collection of relevant materials and the development of the learning objectives to be fulfilled by a lesson and, more importantly, to establish a test bed for comparatively assessing the effectiveness of learning across the various platforms. In this paper, we discuss our framework that includes the MERITS component for specifying lesson plans and the AVA and SLICE components for developing simulations to deliver these lesson plans through interacting with a video player (in the case of AVA) and in a virtual world like Second Life (in the case of SLICE).

The rest of this paper is organized as follows. Section 2 reviews related work on computer based training, using video, and in virtual worlds. Section 3 describes our model for specifying training objectives, course and lesson structure, and the desired (or the expected erroneous) learner behaviors in the context of our video based and in-world training systems. Section 4 explains the video based and in-world training systems and discusses the learners' run-time interaction with them in the context of an illustrative example. Section 5 discusses our plans for future work and Section 6 concludes with a summary of our work to date and the lessons we have learned from it.

Background and Related Work

The term *simulation-based training* refers to a collection of training methods, all of which aim at bridging the gap between classroom knowledge and actual practice, by placing the learner in realistic situations in the context of which he/she has to bring to bear his/her knowledge (of facts, tasks and procedures, and collaboration strategies) to solve a problem.

From an educational psychology standpoint, simulation-based training is supported by the situated cognition theory, proposed by Brown et al. (Brown et al., 1989). According to this theory, knowledge is not a set of abstract concepts to be absorbed by the student; instead, knowledge is dependent on the context and culture in which it is used. Adhering to situation-cognition principles, Collins et al. developed the cognitive-apprenticeship model of educational practice that incorporates the situated nature of the knowledge being conveyed to students (Collins et al., 1991). This model was later evaluated by Järvelä, who found it to be effective within a technologically rich learning environment (Järvelä, 1995). These theories and studies support the value of simulation-based training, which is based upon presenting students with knowledge and teaching skills in a context similar to that within which they will be using those knowledge and skills.

Our group works in close collaboration with health sciences educators who are interested in enhancing their curricula with a variety of computer-assisted training methods, with a particular emphasis on simulation-based training. There are a range of modalities of simulation-based training in health sciences. Mannequins and part-task physical trainers are used for specializations where tactile and physical interaction with the patients is necessary for diagnosing and treating them, such as nursing or surgery. On the other hand, standardized patient actors play the roles of patients, communicating through their verbal responses and physical reactions, mannerisms and emotions with the (teams of) health science student(s), who are responsible for assessing, diagnosing, and treating them. This type of simulation is essential for training healthcare professionals in taking medical history and carrying out clinical conversations with patients, their families, and their colleagues. 2-D and 3-D desktop visualizations and simulations are being used to enhance in-class teaching of complex physiological phenomena (Holzinger et al., 2009). And more recently, virtual world simulations are being developed as more cost-effective alternatives for training students on tasks involving interaction with patients and collaboration in the context of multi-professional healthcare teams.

Table 1: Comparison of Virtual Worlds.

	Personal Space	Arbitrary Positioning	Gestures	Custom Gestures	Physical Touching	Direct Touching	Obstruction	Determinable Eye Gaze	Controllable Eye Gaze	Staring	Voice Chat	Non-verbal Aspects	Chat Pauses	Late Arrival	**Total Score**
Content Sharing															
Google Earth	N	N	N	N	N	N	N	N	N	N	N	N	N	N	**0**
MS Virtual Earth	N	N	N	N	N	N	N	N	N	N	N	N	N	N	**0**
Qwaq	**Y**	**Y**	**Y**	N	**Y**	N	?	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	N	**10**
Media Sharing															
There	**Y**	**Y**	**Y**	N	**Y**	**Y**	**Y**	P	P	**Y**	**Y**	**Y**	**Y**	**Y**	**12**
Vside	**Y**	**Y**	**Y**	N	**Y**	**Y**	**Y**	**Y**	P	**Y**	N	N	**Y**	**Y**	**10.5**
Games															
EVE Online	**Y**	N	N	N	**Y**	N	**Y**	N	N	N	**Y**	**Y**	**Y**	**Y**	**7**
Neopets	N	N	N	N	N	N	N	N	N	N	N	N	N	N	**0**
ourWorld	**Y**	N	**Y**	N	P	N	N	N	N	N	N	N	**Y**	**Y**	**4.5**
Puzzle Pirates	N	N	N	N	N	N	N	P	P	**Y**	N	N	**Y**	**Y**	**4**
RuneScape	N	N	**Y**	N	N	N	N	P	P	**Y**	N	N	**Y**	**Y**	**5**
World of Warcraft	**Y**	**Y**	**Y**	N	N	N	N	P	P	**Y**	**Y**	N	**Y**	**Y**	**8**
Simulation and Training															
3B	**Y**	N	**Y**	N	P	N	N	**Y**	P	**Y**	N	N	**Y**	**Y**	**7**
ProtoSphere	**Y**	**Y**	**Y**	N	P	N	N	**Y**	P	**Y**	**Y**	**Y**	**Y**	**Y**	**10**
Social Networking															
Active Worlds	**Y**	P	**Y**	N	P	N	N	**Y**	P	**Y**	**Y**	**Y**	**Y**	**Y**	**9.5**
Entropia Universe	**Y**	**Y**	**Y**	N	N	N	N	**Y**	P	**Y**	N	N	**Y**	**Y**	**7.5**
Habbo Hotel	P	**Y**	P	N	N	N	**Y**	P	P	**Y**	N	N	**Y**	**Y**	**7**
IMVU	N	**Y**	**Y**	**Y**	N	N	N	N	N	N	N	N	**Y**	N	**4**
Kaneva	**Y**	**Y**	**Y**	N	P	N	N	P	P	**Y**	N	N	**Y**	**Y**	**7.5**
Karga	**Y**	N	**Y**	N	**Y**	**Y**	**Y**	**Y**	P	**Y**	N	N	**Y**	N	**8.5**
Lively	**Y**	**Y**	**Y**	N	**Y**	**Y**	N	N	N	**Y**	N	N	**Y**	**Y**	**8**
Second Life	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**14**
Vivaty	**Y**	**Y**	**Y**	N	**Y**	N	N	N	N	N	N	N	**Y**	N	**5**

Clearly, a substantially larger number of learners can benefit from a program based on desktop simulation than from traditional standardized patient actor simulation programs. Moreover, since participants can be geographically distributed, students in rural areas have the same interaction and educational opportunities as those in metropolitan centers. These advantages, while certainly quite compelling from both a technological and educational standpoint, are highly dependent upon the verisimilitude of the simulation platform used. In our work, we have chosen to focus initially on video and Second Life (as an example of the new breed of virtual worlds) as two alternative platforms for computer-based simulation training. The former provides a high degree of realism of the simulated environment albeit at a cost to the interactivity between the learner and the enacted scenario. The latter enables a greater degree of interaction but somewhat sacrifices the realism of the simulated environment and the expressiveness of the (automatically simulated) characters participating in the scenario through their avatars.

To lessen this shortcoming of virtual worlds as much as possible, we comparatively examined twenty-four virtual worlds, as shown in Table 1. There is a substantial variation in the degree to which the simulated environments and avatars are realistic, somewhat less variety in the types of verbal communication possible among avatars, and a substantial variance in the types of non-verbal cues that these worlds can simulate. Based on the six broad types of non-verbal communication behaviors (proxemics, kinetics, haptics, oculesics, vocalics, and chronemics), we identified fourteen specific behaviors (depicted as labels on the columns of Table 1) and used them as an instrument for comparatively evaluating the capabilities of twenty-four virtual worlds (listed as labels for each of the rows in Table 1). At the time this review was conducted in Summer 2008, Second Life was the only virtual world to have features that would enable the enactment of all fourteen categories of behaviors, though that may no longer be the case. Currently, this includes animated gestures, in-world voice chat, and a highly customizable avatar. However, more subtle forms of expression, such as custom facial expressions, are not currently possible, and may be a stumbling block for certain kinds of simulations.

Another challenge in adopting a virtual world as a simulation-based training platform is the degree to which the behavior of the character standing in for the standardized patient can be automated. While simple actions such as moving, talking, and listening for phrases have already been implemented, more complex actions may prove more difficult to automate. A related issue is the expressiveness of these automated characters. Today, creating an automated character that can move realistically and act shy, nervous or angry is still impossible in all virtual worlds. However, much of the related work (including ours) relies on the assumption that this lack of expressiveness will not fundamentally destroy the sense of social presence (Wheeler, 2005) required to make these simulation believable. Furthermore, the technology is continuously improving and the field assumes that this problem will eventually become less significant. In the two following sections, we review previous research on the usage of videos and virtual worlds in training, with a special interest in the health-sciences domain.

Video-based training

Video has been long used for teaching, usually for demonstration of expert (or problematic) performance for students to mimic (or avoid). This use is relatively passive – the student simply watches the performance – and it is the responsibility of the instructor to focus the students' attention to the more pertinent aspects of the demonstration and to guide their reflection on how the demonstration relates to the rest of their knowledge and skills. Recently we have witnessed a wave of research on how to use annotated video in an effort to support or even automate the role of the instructor. In the following paragraphs, we will describe several video annotation systems being used in a variety of contexts.

The Shakespeare Video Annotation System (SVAS) "brings us closer to realizing the potential of digital media to transform education across the humanities, arts, and in other subjects in which there is a need for flexible access to multimedia archives and the need for a rapid, conversational pace as the exchange of ideas converge" (Donaldson et al., 2008). The system enables users to compare literary classics to their corresponding versions in modern media. It allows students to view multiple movies (for instance, different Shakespeare performances) at the same time and add text annotations to different performances and remotely share their online commentaries and discussions. A product of the project is XMAS, the "Cross Media Annotation System," which provides a multimedia essay editor (where movie clips can be seamlessly introduced into an essay), an online discussion component with references to video clips, and a mechanism for annotating a variety of media, such as DVDs, images, texts, and streaming video.

Butler and his colleagues have used video annotation to facilitate assessment of educational events such as presentations, seminars, and interviews (Butler et al., 2006). Current educational systems tend to use presentations and seminars in their assessment procedure, especially in higher education. These methods encourage deep, as opposed to surface, learning by guiding the learner to focus on the significant aspects of the presentations, relating it to his (her) previous knowledge, and integrating everything in a coherent whole (Marton and Säljö, 1976). However, unlike exams, which can be kept forever, it is difficult to keep records of these transient events. Butler's system provides the facility to keep some records of transient events during presentation. For example, each user can attach notes or comments to timelines, and defined annotations will be attached to video later. In other words, this system will capture the transient events for later use and will make the process of assessment less complex.

eSports is a video annotation system used for distance sport coaching. Supposing that coach and players are in different locations, they can login to the system and watch a video at the same time. Then they can add annotations to interesting and important shots. In this way, they can share their ideas and discuss together, as if they were sitting together in a real classroom.

Video Traces provides users with a simple user interface to change video play speed and sequence, freeze video frame, or point on different parts of the video (using a "finger tool") while discussing and talking about video content. The user can add comments on parts of video or particular interest. All user changes, including audio comments, gestures, and video playback changes are overlaid on the original video, to produce as a "video trace." This system has been particularly helpful in teaching dance disciplines, where the focus is on learning by doing, through facilitating critical evaluation and reflective thinking both for dancers and choreographers.

The KLIV system brings video-based learning content on a mobile platform, to be shared by peers. In a really interesting application, nurses produce short video clips on best practices, which are stored on a server and accessed by students through their handhelds within the hospital.

As is evident from the variety of applications mentioned above, video – raw as well as annotated – is used in a range of contexts to enhance teaching and learning. However, current video annotation systems typically use annotations to provide additional explanatory comments to the original video footage. While this is really useful, it does not fundamentally change the passive nature of video as a platform. In our work, we propose *actionable annotations*, supporting the user's interaction with the video stream and his/her response or action relative to the content presented. In this manner, video instead of being yet another media for content becomes a simulation platform where learners are brought into a situation and have to make decisions about it based on their knowledge. We have developed our idea of actionable annotations in the Actionable Video Annotation (AVA) system, which in addition to standard passive annotations, supports several types of actionable ones, thus providing an interactive simulation environment which can be of great use in education.

Training in virtual worlds

In spite of the relative recency of the virtual worlds phenomenon, several educational projects that use virtual worlds in various ways have already been reported. A large number of educational institutions are experimenting with setting up virtual campuses, with areas for students to meet and work, and classrooms in which to hold meetings and lessons. Sometimes integrated with 2-D learning-management systems, such as Moodle, these virtual campuses are intended to motivate further social interaction among students and instructors and to provide an opportunity for creative activities in the virtual world that can enhance learning.

In addition to such "general" experiments in establishing a presence in a virtual world, several specific virtual world education experiments for providing better instruction on selected subject areas have been reported. These experiments cut across subject areas and use a broad range of technologies. Vergara and his colleagues at the University of New Mexico teach medical students about evolving hematomas (Vergara et al., 2008). They have developed a virtual character, nicknamed "Mr. Toma," to provide students with a chance to interact with a person and other associated objects in a 3-D, multi-user virtual environment (MUVE). Several rigorous studies of the system's effectiveness have demonstrated that it is equally effective as conventional, paper-and-pencil education methods. Furthermore, it offers additional advantages, including the chance to collaborate with geographically dispersed students and an increased sense of immersion when using the MUVE system. A considerable amount of effort was put into ensuring that the content was presented accurately and effectively, including consulting with an interdisciplinary team of subject matter experts.

Similarly, Adamo-Villani has developed a photorealistic 3-D virtual laboratory for an undergraduate course on microcontroller technology (2006). The project is aimed at students in electrical engineering, and is particularly concerned with offering a lifelike lab experience for students, such as those with physical disabilities, who would not otherwise have access to a lab. Because the project is restricted to teaching students about microcontrollers, the interface – and, indeed, the simulation as a whole – is quite closely tailored to this context and the interactions are limited to those implied by the microcontroller being constructed. The interface is largely two-dimensional, with camera controls enabling students to pan across an item or zoom in on a particular feature. The simulation is not accessible online, and thus there are no opportunities for collaboration with other students.

In a different context, Carpenter (2006) has developed a 3-D crisis-communication training tool to provide communication students with opportunities to practice what are, in a standard classroom setting, largely theoretical approaches to dealing with crises. Through the immersive tool, students get hands-on training and can experience events, rather than absorbing and interpreting them through written information. The tool uses facial modeling for virtual characters, a range of story settings, and virtual reality based user interface devices (a head tracker and wand) to provide an immersive experience for the student. The tool uses a narrative, storyboard-based technique to deliver the educational content, where each student is offered a set of choices at key points in the story. Afterwards, the students are debriefed and the instructor analyzes and evaluates their choices. Because the system uses storyboards to structure the educational content, a student's interaction with the system is largely pre-determined and quite rigid. As well, the system does not support collaborative learning, since it is meant for use by one student at a time.

Moving away from using virtual environments to teach discipline-specific knowledge and skills, Jones focused on the effect of 3-D online learning environments on online discourse (2005). He found that courses using 3-D learning environments attained a high level of online discourse more quickly than those that used conventional web-based systems exclusively. He also analyzed some of the barriers to the adoption of 3-D learning environments, such as the cost of integrating new technology into existing systems, the time and effort required to create new educational materials, and the perception that 3-D environments are meant for gaming, not education. These barriers were weighed against advantages such as high levels of student motivation, improved academic efficacy, and the appeal of an immersive, stimulating 3-D environment. To this list of advantages, the author is able to add improvement of online discourse, which may help push some educators or institutions towards integration of 3D environments into their programs.

In a similar vein, Cai (2008) has taken a broader view of the issue, examining the potential of virtual worlds for any kind of training program. He compared several virtual environments – Second Life, ActiveWorld, OpenSim, and the Torque game engine – in terms of their fitness for educational activities and analyzed various common learning activities with respect to their implementation in a virtual environment. He also presented a development lifecycle for creating virtual learning environments and analyzed several virtual learning projects at IBM according to these analytical tools.

There is clearly substantial excitement about Second Life and the socialization opportunities it affords with its large user population. At the same time, specialized virtual worlds exist that support education and/or professional training simulations – Forterra the most recognizable among them. To our knowledge, our work with SLICE is unique in attempting to provide a dynamic-simulation system within Second Life, which should be of interest to the education institutions that have adopted Second Life for their virtual campuses.

An Extendible Framework for Simulation-based Training

The long-term objective of our work is to develop an extendible framework for specifying lesson plans and their potential delivery in a variety of computer-assisted methodologies. As computer technologies improve and evolve, including mobile personal computers, interactive displays, and haptic devices, to name a few examples, we envision new ways to include them in teaching and learning. And as we creatively come up with novel pedagogical methodologies, the need to comparatively assess their cost effectiveness will become ever more pressing. This is why in our work on simulation-based training we have focused on separating the lesson specification from the specification of its technology-specific delivery details.

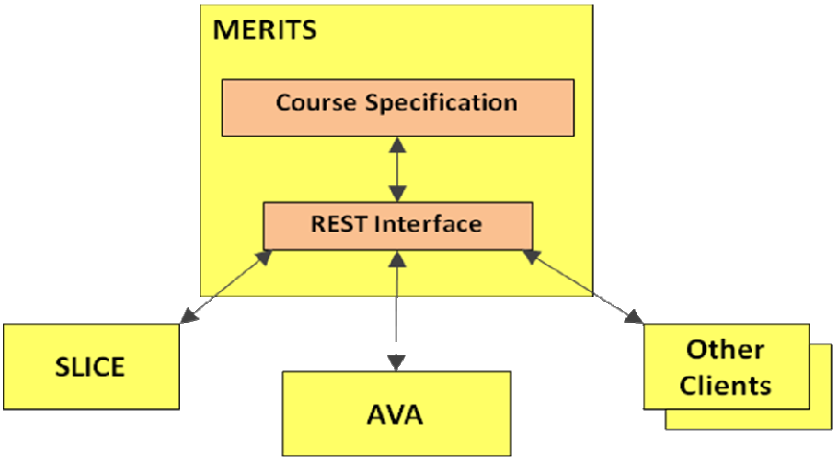


Figure 1: High-level Software Architecture of Our Simulation-based Education Framework.

As shown diagrammatically in Figure 1, the MixEd Reality Integrated Training System (MERITS) component is responsible for maintaining course and lesson plan specifications. These specifications are accessible by a collection of platform-specific clients, through a REST (Representational State Transfer) API. To date, we have developed two clients: Actionable Video Annotation (AVA) and Second Life Integrated Curriculum Environment (SLICE). Each of these clients consists of two components: one to be used by the instructor to define how the course concepts are implemented in terms of the specific interaction mechanisms supported by the platform and the second one to be used by students to take the course.

Course specification in MERITS

The MERITS component organizes the course specification around three types of knowledge:

(a) an organizational structure, in which the content is stored;
(b) record keeping entities to monitor the student's educational progress; and
(c) domain-specific concepts of the course.

The organizational structure is a simple hierarchy, as shown in Figure 2. A course is subdivided into modules, each of which, in turn, contains an ordered sequence of educational items, of a variety of types, such as lessons and scenarios. A lesson contains a static piece of educational content and thus, may contain elements such as text, multimedia components (e.g., audio clips or videos), and questions. A scenario, on the other hand, encapsulates an interactive educational experience and thus, consists of components that describe this interaction. Within a question, one may have multiple options (for a multiple choice question) or a text-entry field for a question with an open-ended answer.

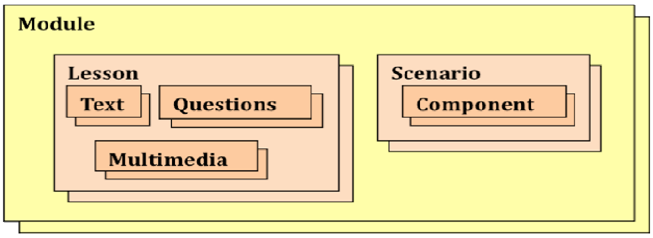


Figure 2: Structural Entities.

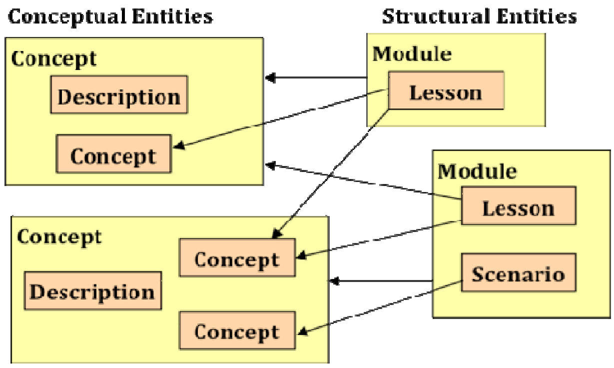


Figure 4: Conceptual Entities.

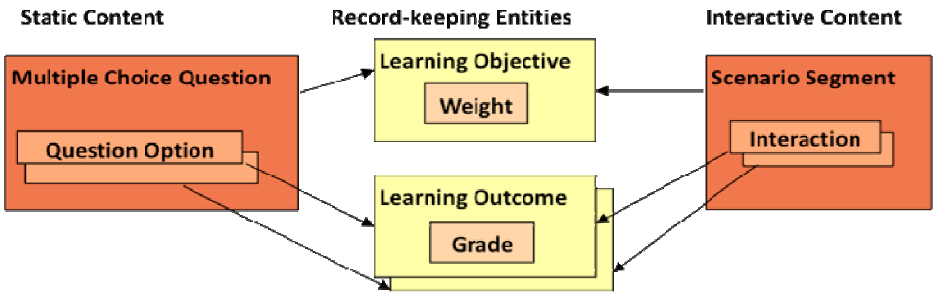


Figure 3: Record Keeping Entities.

For example, a module could encapsulate a course on how to conduct job interviews. This module, in turn, would contain several lessons and modules that would convey the course material. The course could start with a lesson about how to prepare to conduct an interview. This lesson could contain photos of an office, a video showing how to greet an applicant, and text descriptions of the various questions that one might ask. This lesson could be followed by several scenarios which would give the user a chance to practice conducting an interview. Finally, the module might also contain a lesson about how to conclude an interview and analyze the applicant's responses, which would contain appropriate text and multimedia resources. The record keeping entities are of two types: learning objectives and learning outcomes. A learning objective encapsulates the record keeping involved in assessing a student's competence in a particular area. A learning outcome, on the other hand, corresponds to a particular input from the student indicating their competence. The relationship between the record keeping entities and student assessment is shown in Figure 3.

Finally, the abstract conceptual entities are intended to capture the domain-specific knowledge of the course and in particular, around the competencies that are being taught. Conceptual entities serve some of the same function as modules in that they organize a course into broad, theme-based sections. However, while modules are organized in such a way as to convey the educational material in discrete, ordered units, conceptual entities have a much looser structure. They do not have an order associated with them and may be associated with each other or even nested within each other to create conceptual hierarchies. Thus, the instructor may begin creating a course by first establishing the set of concepts to be conveyed and then organizing modules, lessons, and scenarios around these foundational concepts. The relationship between conceptual and organizational entities is shown in Figure 4. Note that structural entities may be connected to multiple conceptual entities and that the hierarchy of a structural entity might not match that of a related conceptual entity.

As is evident from the preceding paragraphs, fully specifying a course involves substantial work by the content creator, who needs to decide the set of relevant concepts, organize the appropriate educational content, and establish the record keeping mechanisms needed to assess a student's progress in learning the concepts. Moreover, for any existing course, most of this work has already been done – lecture notes have been created, assignments devised, and exams prepared. Thus, requiring that this information be respecified in a new system threatens the system’s adoption potential. Further, it is quite likely that this information is already in digital form, thanks to the prevalence of learning management systems (LMS) such as BlackBoard, WebCT, and Moodle. Thus, taking advantage of an established e-learning standard is crucial to encouraging the adoption of the MERITS system.

The Shareable Content Object Reference Model (SCORM), initially developed by the Department of Defense, is an XML-based standard for structuring content in an e-learning system. It is followed by all major LMS systems and ensures a basic level of interoperability between the systems. The MERITS system currently uses a small subset of the concepts and entities described by the SCORM standard and because of the conceptual and organizational overhead involved in SCORM compliance, it does not directly conform to the SCORM standard. However, we are investigating the development of a “translator” for the MERITS system that would convert between SCORM-compliant educational content and the format used by MERITS. This would allow us to take advantage of the interoperability offered by SCORM compliance without unnecessarily complicating the underlying structure of the MERITS system.

Student-behavior specification

The development of the lesson plan by the instructor is done in the context of MERITS, since it is client independent. On the other hand, the specification of the student’s interaction with this material depends on the client platform; therefore, it is specified in each of the MERITS clients.

Behavior specification in AVA

Current video annotation tools support a variety of annotation types from very simple annotations such as text labels to some more advanced objects such as descriptive figures, drawings, and images. However, all these annotation types have a common trait; all of them reveal a passive behavior. This means that the user only perceives the extra content provided by these annotations while watching the annotated video. Although passive annotations are very useful, we propose that video could be transformed into a simulation platform with interactive actionable annotations, guiding the learner to act on the video based on his or her knowledge and the context set up by the video content. The AVA system implements this novel proposal.

AVA’s annotations include passive annotations which only add a set of informative, explanatory notes to the video and actionable annotations which demand an appropriate response from the learner watching the video and appropriately react to his/her response. For example, the learner may be asked to answer a fact-finding or comprehension question based on the current video scene or to point to a relevant point in the video image. In response, AVA may pop out an overlay label with additional information on the learner’s response, change the playing head position to a different scene where additional relevant information can be found, or ask a follow-up question.

AVA defines an XML-based language for specifying *passive annotations,* such as simple text, URLs, and images, and *actionable annotations,* such as questions and decision points and their corresponding operations. All annotations and related operations are stored in an XML file that accompanies the original video as metadata. This document contains three different elements, *annotations*, *assigned annotations*, and *frames*, as shown in Figure 5.

The annotations element, which roughly corresponds to the concept entity of MERITS, is defined in terms of a unique *annotation id*, its *type (*whether it is a label/URL/image to be shown, or a multiple-choice or open-end question to be answered, or an interaction to be performed on the video image), its *value*, and its *owner* information.

```
<?xml version="1.0" encoding="UTF-8"?>
<AnnotationSpec lessonId="">
    <Body>
        <Annotations>
            <Annotation annotOwner=" " annotType="" annotValue="" id=""/>
        </Annotations>
        <AssignedAnnotations>
            <AssignedAnnotation annotId="" binded=" " from="" tag="" to="" xPos="" yPos=""/>
        </AssignedAnnotations>
        <Frames>
            <Frame frameNo="">
                <OnEntry>
                    <Operation annotId="" frameNo="" id="" tag="" xPos="" yPos=""> </Operation>
                </OnEntry>
                <OnAnswer tag="">
                    <Answer answerValue="">
                        <Operation annotId="" frameNo="" id="" tag="" xPos="" yPos=""> </Operation>
                    </Answer>
                </OnAnswer>
                <OnExit>
                    <Operation annotId="" frameNo="" id="" tag="" xPos="" yPos=""> </Operation>
                </OnExit>
            </Frame>
        </Frames>
    </Body>
</AnnotationSpec>
```

Figure 5: The AVA Video Metadata Schema.

An *assigned annotation* element associates an annotation to a sequence of frames, starting at *from(T1)* and ending at *to(T2)*, and to a specific *x*, *y* location. At run time, this annotation is superimposed at location *x*, *y* on the learner's video during the defined time period. By separating the actual annotations from their assignments, it is possible to assign a single annotation to several different time slots or have multiple annotations shown at the same time, just as a single concept may be communicated in different ways or a statement may communicate multiple concepts. Each time an annotation assignment is enacted, a unique value is generated for "tag" attribute to distinguish between the various assignments.

The *frames* element keeps a record of all operations (showing or hiding annotations, at the entry or exit of a frame, or upon receiving from the learner an answer to an actionable annotation) needed to manage the annotations relative to the video sequence. For example, for an annotation assigned to time *T1* to *T2*, two cue-points (two frames) are inserted in frames section. The "*ShowAnnotation*" operation is added to "*OnEntry*" part of cue-point *T1*, and "*HideAnnotation*" operation is placed on "*OnExit*" part of cue-point *T2*. At run-time, the annotation is superimposed on the video between times *T1* and *T2*. For all actionable annotations such as questions, in addition to two previous mentioned frame elements, one frame element called "*OnAnswer*" is added. The operations in this subsection vary depending on the actionable annotation type and the operations defined by user in response to possible viewer actions. Moreover, a "*Pause*" command is automatically added to "*OnEntry*" section of the ending cue-point in order to stop the video to receive the viewer action. For instance, for a multiple choice question annotation, an operation like "*Seek*" or "*Continue*" may be assigned to each question option. Then during video play, AVA stops the video on the end time of displaying question to ask the learner for an answer, and depending on how the given answer compares to the answer specified by the teacher, the player will jump to some specific points or continue playing video. A number of operations may be assigned for each option (or in general, each viewer action). For example, conditioned on a specific viewer action, the AVA player may use "*ShowAnnotation*" command to show another conditional annotation, and then "*Continue*" or "*Pause*" to receive another user action (multiple operations per viewer action is not implemented in this version).

The AVA system is implemented in the service oriented style, so it can be easily integrated with other applications and connects to MERITS via the MERITS REST APIs to retrieve questions and store the learners' answers. Two different sets of services are provided by AVA. The first set includes the services for defining annotations and operations that directly manipulate the content of the XML file associated with the video content. The second set of services is used at run time by the AVA video player to retrieve annotations and operations and appropriately control the interaction with the learner.

Behavior specification in SLICE

In the SLICE component, the concepts defined in MERITS are communicated to the learners via interactive workflows between the student and scripted characters, created by educators (or domain experts). SLICE provides a simple web-based toolkit for efficiently creating such workflows, which is currently implemented using a custom workflow engine. However, we are moving towards expressing these workflows in Business Process Execution Language (BPEL). BPEL is, in essence, an XML-based method for describing workflows. While the simplest workflow will describe a linear sequence of actions (e.g., B follows A), one can also specify more complex workflows which may contain actions which will only occur under certain conditions, for example, or actions which may occur in parallel. As BPEL has become more widely used, variations have emerged which extend its original focus – business processes – to areas such as web services (WS-BPEL) and processes involving human-computer interaction (BPEL4People). BPEL specifications can be executed by several existing execution engines and can be interactively constructed through corresponding modeling tools.

An important advantage of using a workflow-based scheme for specifying character behavior is that it can be used to describe both macro- and micro-level behaviors. That is, on a small scale, the educator can describe simple, atomic actions such as moving from one place to another or listening for a certain phrase. However, these actions can be combined to create composite, complex actions such as carrying out an interview or assembling a piece of (virtual) equipment. Thus, simple, broadly applicable actions may be composed in various ways to create complex actions that are customized for particular contexts or educational programs.

Once the workflow has been specified, it is stored in an online database and is accessed by the automated character within Second Life to guide its behavior when interacting with a student. These characters may converse with the student, move around, or perform more complex actions. The range of character behaviors is limited only by the workflow specification and the implementation of these behaviors in Second Life. That is, an avatar could be told to scratch its head, as long as the character's avatar in Second Life knows how to perform this action. The behaviors used in different educational contexts will likely vary a great deal, and thus, the system will eventually store both a general repository of behaviors for all contexts, and a smaller collection of behaviors relevant to a particular context. The process of creating, storing and running a workflow is shown in Figure 6.

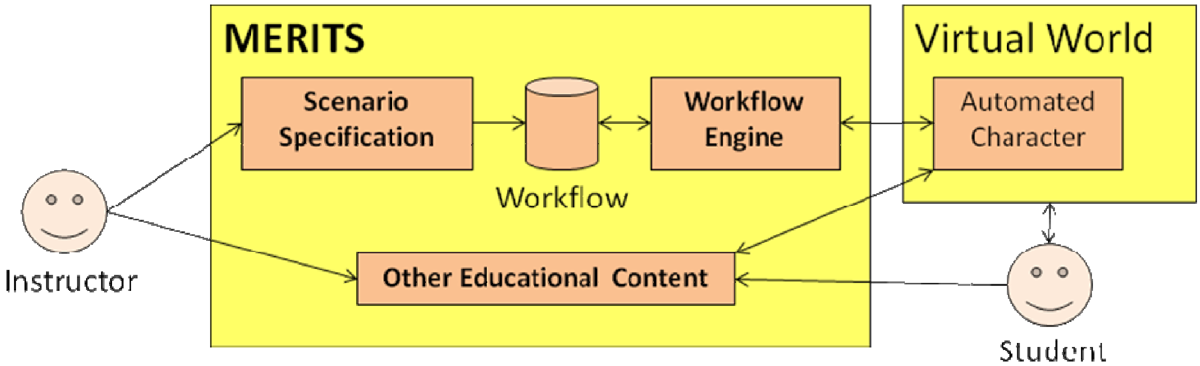


Figure 6: Workflow Process.

The "Corporate Interview" Scenario

In this section, we discuss how the student interacts with each of the two training systems (AVA and SLICE) at run time, in the context of a "training for a corporate-interview" example. There is a lot of online advice on the topic, on issues ranging from how to prepare for such an event, how to dress for it, and how to answer specific types of questions that usually arise during the interview. In fact, we found a sequence or related videos on YouTube, which we have used as an example on which to base our first application of our systems.

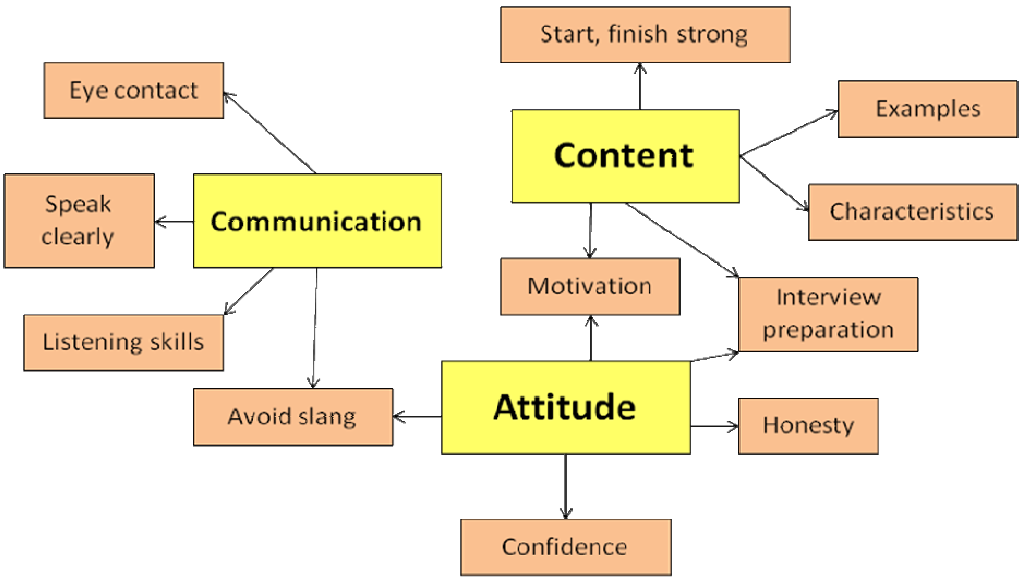


Figure 7: Core Concepts for Job Interview Training.

The videos from Denham Resources, a human resources training company in Fresno, California, show job applicants giving good and bad answers to common interview questions. The answers are annotated with brief explanations of what makes that answer good or bad, as the answer is given. For example, after an applicant giving a “bad” answer laughs nervously, an annotation is shown telling the viewer that laughter indicates nervousness or dishonesty. The video series covers questions such as “Why do you want to work here?” and “What is your biggest weakness?”

Using this material as a starting point, we came up with a set of principles and core concepts, diagrammatically depicted in Figure 7. These concepts were organized around three core ideas: communication, content, and attitude. More specific concepts such as “avoiding sarcasm” and “making good eye contact” were then grouped within these three core concepts. While some concepts (such as preparation and avoiding slang) fell under more than one category, most of the more specific concepts fit nicely into a single category. Using these principles, we created appropriate structural, conceptual, and record-keeping entities within the MERITS system. For a description of these entities, see the following section.

MERITS entities

We have created an interview training module which encapsulates all of the educational content related to training a student in conducting job interviews. Within that module, we have created scenarios for “good” and “bad” answers to five common interview questions: “Tell me about yourself,” “Describe your biggest failure,” “Describe a time you went above and beyond at work,” “What is your biggest weakness?,” and “Why should we hire you?” These scenarios contain components that allow the student to pose each question to an automated character, the character to respond with a “good” or “bad” answer, and the student to then make observations about the character’s answer. We have also created lessons for each of these interview questions to store static information pertaining to that question. Specifically, each lesson contains multiple choice questions about that interview question, which can be presented as video annotations by the AVA system. Finally, it should be noted that both the student’s observations (in a virtual world), and their answers to the multiple-choice questions (using the AVA system) are connected to record keeping entities, which allow these student actions to be tracked and graded by the MERITS system. For a listing of a representative segment the MERITS entities created for this module, please see the Appendix, which presents a representative selection of these entities. A complete listing is available at http://www.cs.ualberta.ca/~chodos/training/showEntities.php.

Video-based training with AVA

The complete lesson, with a manual on its use, is available at http://www.cs.ualbert.ca/~chodos/training/ava.html. As discussed earlier, a course in AVA system is composed of a series of video segments plus a set of commentary (passive) and actionable annotations. The sequence of presenting video segments depends on the lesson plan and student actions (answers) on actionable annotations. Therefore, in general, a video segment may be played several times whereas another segment may be not reached at all.

Let us consider, for example, the short job interview course that is designed to help the students experience the actual interview sessions. The course is composed of several video clips presenting typical answers of interviewee to the sample interviewer questions. The student sits behind the system in the role of an interviewer and picks one of the available answers to a specific question through a multiple choice actionable annotation (see Figure 8).

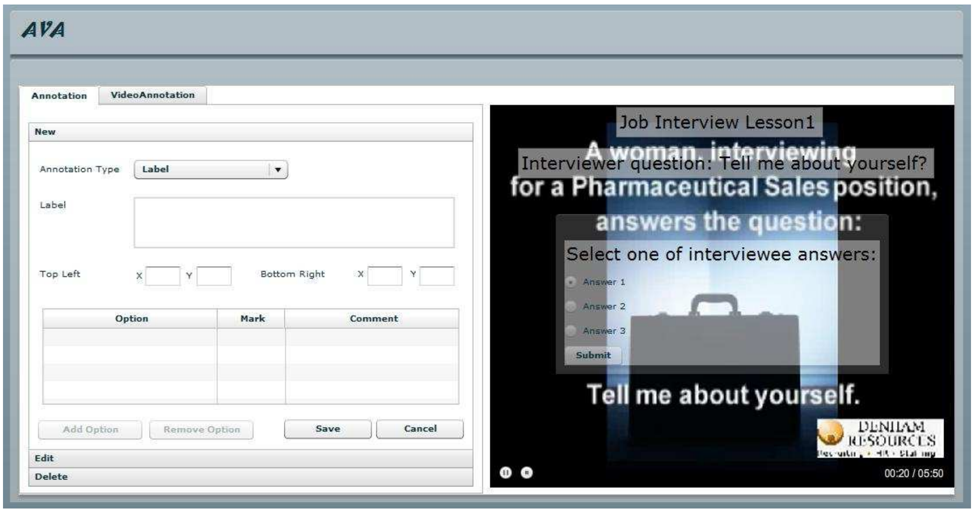


Figure 8: AVA Screen Shot 1.

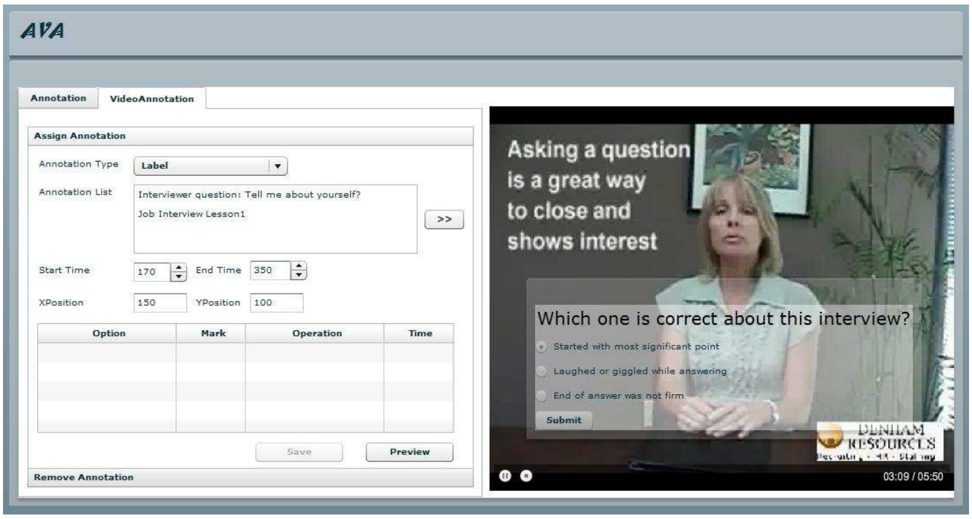


Figure 9: AVA Screen Shot 2.

In response to the student action, the video player jumps to the corresponding video segment and plays the sample interviewee answer. During or at the end of the interviewee answer, the student is asked to present her/his observations by answering one or more multiple choice questions (see Figure 9). Depending on the accuracy of the student observations, the player may let him (her) watch other samples or replay all or some parts of the current sample to notify the points possibly missed by the student.

In a complete course on job interview training, the questions and interviewee answers to those questions may be chosen according to a predefined specific order, and the students not only may be asked to give their observations on each specific question but also may be asked to give their overall observation on the total interview session. In this way, the job interview session environment will be simulated for the student; instead of simply watching the interview and the running commentary, the student may, in fact is asked to, provide his (her) own input on interesting aspects of the interview process, thus engaging more with the training process.

Training in Second Life with SLICE

The complete lesson, with a manual on its use, is available at http://www.cs.ualberta.ca/~chodos/training/SLICE.html. The course, when viewed by the student within a virtual world, is conveyed through a series of interactions with scripted characters. While other modes of interaction – such as multiple choice quizzes, for example, or conversation with an instructor – are possible within a virtual world, these interactive workflows are a very engaging way of presenting certain kinds of educational material and are particularly effective within a virtual world.

Figure 10: Screenshot of Virtual Office.

In the context of training students to conduct job interviews, the student is placed in an office setting and is joined by an automated character representing a job applicant. Figure 10 is a screenshot of the student and applicant in the virtual office.

The student can ask the applicant any of several standard job interview questions. For each of these questions, the applicant is scripted to respond with either a "good" or "bad" answer, which the student should be able to evaluate. These responses can either be shown on screen (as in the following screenshots) or played as a series of audio clips. To demonstrate recognition of the relevant qualities of the interviewee's answers, the student can make observations at any point during the applicant's answer. A flowchart showing this interview process is shown in Figure 11. These observations currently operate using a simple keyword-based system, although we are planning on creating a more robust, natural language system in the near future. See Figure 12 for a screenshot of the interaction between the student and the applicant, which includes the student making both correct and incorrect observations about the applicant's answer. See Figure 13 for a list of the characteristics shown in one of the questions.

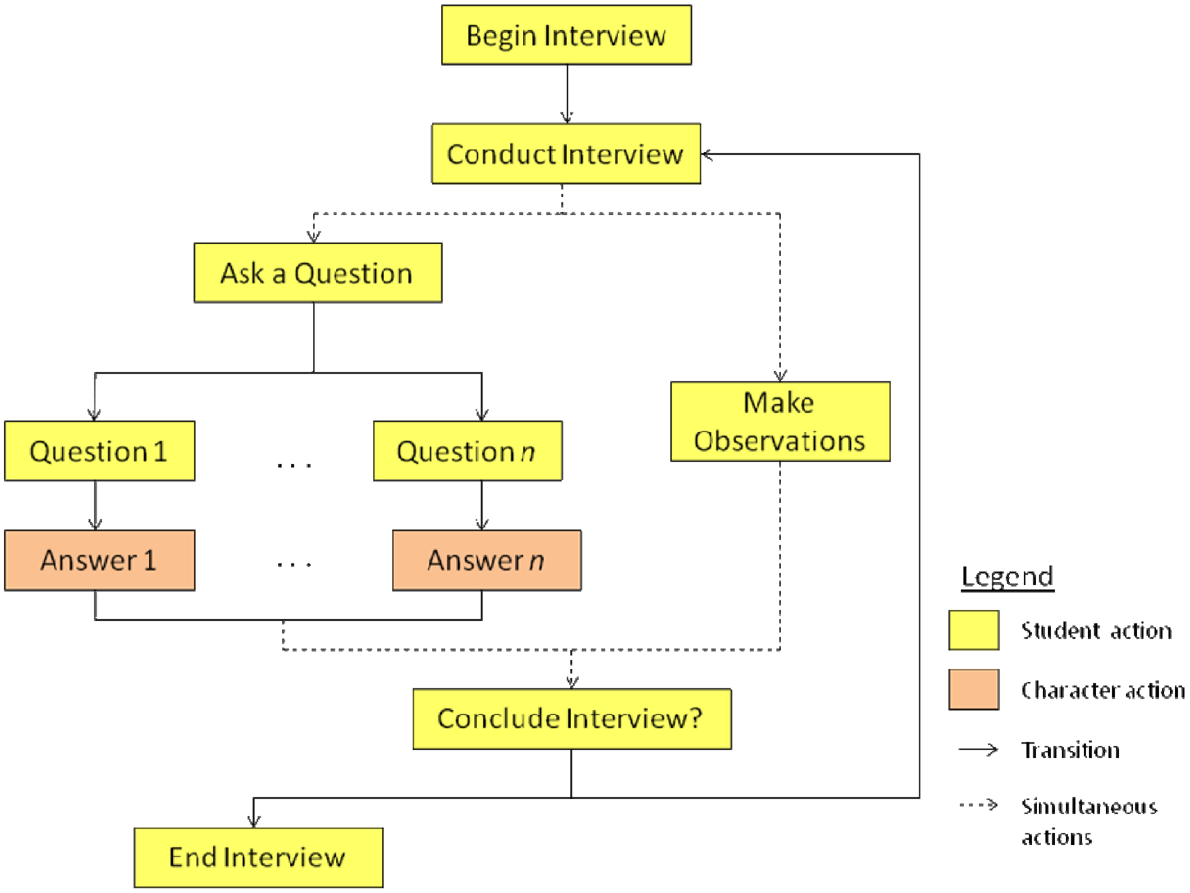


Figure 11: Flowchart of Interview Process.

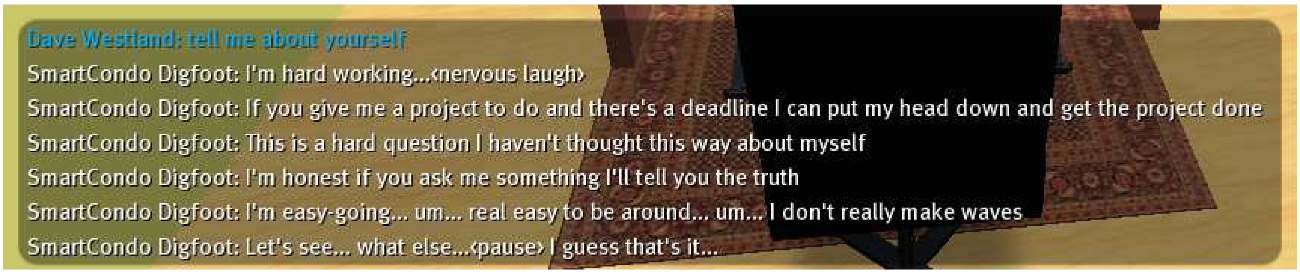


Figure 12: Interaction Between Student and Applicant.

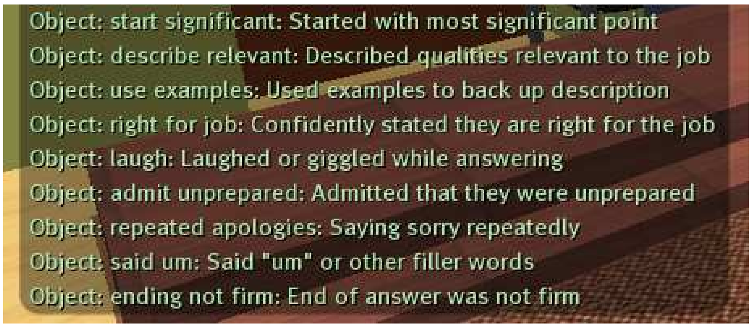


Figure 13: List of Characteristics.

Finally, it should be noted that the student's observations, including multiple choice answers or any other kind of feedback, are linked to the appropriate record-keeping entities defined within the MERITS system. Thus, the student's interactions in the virtual world are linked to, and thus become an integrated part of, a broader online course in interview training.

Future Work: Experimental Evaluation

The most important task to be accomplished in the short-term future is an experimental evaluation of our infrastructure. To assess the effectiveness of the virtual world based training scenarios, we propose comparing it with the video annotation system, using a common educational context. We will focus on the field of training human resources staff to conduct job interviews. Using previously generated material from a web based course as a starting point, we came up with a set of principles and core concepts. Using these principles, we created appropriate structural, conceptual, and record keeping entities within the MERITS system. Then, using a set of interview training videos from Denham Resources, we created interactive scenarios and annotated videos that address the educational goals of the program. Thus, a student can be taught the course material via either the virtual world or annotated video clients.

With this ability to deliver educational content using multiple delivery methods, we can take a class of students and divide them into several groups: conventional, virtual world, annotated video, and blended. The conventional group will not use any MERIT-based client and will act as a control group. The virtual world group will use the virtual world client exclusively, and the annotated video group will, similarly, use the video client exclusively. Finally, the blended group will have access to both clients and will be able to choose which client to use for each part of the course. The division of the students will be performed randomly, and will take potentially confounding demographics (e.g., age or gender) into account. At the same time, since students from the same course (and education background) will form all groups we hope to eliminate any other confounding variables, such as the educational environment and the course instructor. The experimental evaluation will investigate the following questions:

1) Does the use of a MERITS-based client have an effect on students' performance?
2) Is there a difference in students' performance when using the video-based client, as compared to the virtual world based client?
3) Does offering a choice of either client, as opposed to offering one client exclusively, have an effect on students' performance?

A key component in all of these questions, of course, is the measurement of students' performance in the course – that is, the extent to which the students are able to learn the material being presented. At a superficial level, students' grades and exam scores may be compared, but this offers a one-dimensional view of each student's performance. Rather, we propose a multi-faceted measure of the students' performance, based on the following metrics: academic performance, level of involvement, and aggregated quantitative survey results.

Summary and Conclusions

In this paper we discussed a software prototype designed to support simulation-based learning on multiple platforms, including video and Second Life. Our system consists of three related components.

(a) The MixEd Reality Integrated Training System (MERITS) component serves as the repository of platform-independent information about courses, lesson plans and their associated materials, as well as information about the students' performance when taking these lessons.

(b) The Actionable Video Annotation (AVA) supports instructors in specifying passive and actionable annotations on video, through which to communicate the MERITS lesson plans. It also includes a player component that, at run time, controls the video play, the delivery of the specified annotations and the interaction with the learner.

(c) The Second Life Integrated Curriculum Environment (SLICE) component guides instructors in specifying workflows through which to enact the MERITS lesson plans in Second Life and through a reengineered Second Life client, controls the environment and the automated workflow characters at run time letting the learners play the remaining workflow roles.

Admittedly, the prototype is not mature and unfortunately, we do not yet have experimental evaluation results on its effectiveness for teaching and learning. Nevertheless, we believe that the software architecture we have developed clearly delineates the boundaries between the different types of information necessary for developing simulations for learning and flexibly supports, and hopefully more in the future, environments for simulation-based learning.

Bibliography

Adamo-Villani, N., Richardson, J., Carpenter, E., & Moore, G. (2006). A photorealistic 3d virtual laboratory for undergraduate instruction in microcontroller technology. In *ACM SIGGRAPH 2006 Educators Program.* Boston, Massachusetts, July 30 - August 03, 2006. New York: 21.

Aldrich, C. (2005). *Learning by doing: A comprehensive guide to simulations, computer games, and pedagogy in e-learning and other educational experiences*. April 2005, San Francisco, California: Wiley: Pfeiffer.

Brandt, E., Björgvinsson, E., & Hillgren, P-A. (2004). Self-produced video to augment peer-to-peer learning. Learning and Skills Research: A journal for further education and lifelong learning. *Learning and Skills Development Agency*, p. 27-34.

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher,* 18, p. 32-42.

Butler, M., Zapart, T., & Li, R. (2006). Video annotation – Improving assessment of transient educational events. In *Proceedings of the Informing Science and IT Education Joint Conference*, p. 19-26.

Cai, H., Sun, B., Farh, P., & Ye, M. (2008). Virtual learning services over 3D internet: Patterns and case studies. In *Proceedings of the 2008 IEEE international Conference on Services Computing - Volume 2,* July 7 - 11, 2008. Washington, DC: IEEE Computer Society, p. 213-219.

Carpenter, E., Kim, I., Arns, L., Dutta-Berman, M. J., & Madhavan, K. (2006). Developing a 3D simulated bio-terror crises communication training module. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*. Limassol, Cyprus, November 1 - 3, 2006. New York: ACM, p. 342-345.

Cherry, G., Fournier, J., Stevens, R. (2003). Using a digital video annotation tool to teach dance composition. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 5.

Collins, A. (1991). Cognitive apprenticeship and instructional technology. In L. Idol & B. F. Jones (Eds.) *Educational values and cognitive instruction: Implications for reform*. Hillsdale, NJ: Lawrence Erlbaum, p. 121-138.

De Freitas, S. & Neumann, T. (2009). The use of 'exploratory learning' for supporting immersive learning in virtual environments. *Computer Education*, 52, p. 343-352.

De Lucia, A., Francese, R., Passero, I., & Tortora, G. (2009). Development and evaluation of a virtual campus on Second Life: The case of SecondDMI. *Computuer Education,* 52, p. 220-233.

Donaldson, P. Shakespeare Video Annotation Project. Retrieved from http://icampus.mit.edu/projects/Shakespeare.shtml and http://web.mit.edu/shakspere/xmas.html

Fielding, R. T. & Taylor, R. N. (2000). Principled design of the modern Web architecture. In *Proceedings of the 22nd international Conference on Software Engineering,* Limerick, Ireland, June 4 - 11, 2000. ACM, New York: ACM, p. 407-416.

Holzinger, A., Kickmeier-Rust, M. D., Wassertheurer, S., & Hessinger, M. (2009). Learning performance with interactive simulations in medical education: Lessons learned from results of learning complex physiological models with the HAEMOdynamics SIMulator. *Computer Education,* 52, p. 292-301.

Järvelä, S. (1995). The cognitive apprenticeship model in a technologically rich learning environment: Interpreting the learning interaction. *Learning and Instruction,* 5, p. 231-259.

Jones, J. (2005). Accelerating online discourse via 3D online learning environments. In *ACM SIGGRAPH 2005 Educators Program,* Los Angeles, California, July 31 - August 04, 2005. New York: ACM, 42.

Kolb, D. (1984). *Experiential learning: experience as the source of learning and development.* Englewood Cliffs, New Jersey: Prentice Hall.

Marton, F. and Säljö, R. (1976). On qualitative differences in learning — 1: Outcome and process. *British Journal of Educational Psychology,* 46, p. 4-11.

Marton, F. and Säljö, R. (1976). On qualitative differences in learning — 2: Outcome as a function of the learner's conception of the task. *British Journal of Educational Psychology,* 46, p. 115-27.

Vergara, V., Caudell, T., Goldsmith, T., Panaiotis & Alverson, D. (2008). Knowledge-driven design of virtual patient simulations. *Innovate (Journal of Online Education),* 5, p. 1-7.

Wheeler, S. (2005). Creating social presence in digital learning environments: A presence of mind? Queensland, Australia: TAFE Conference.

Zhai, G., Fox, G. C., Pierce, M., Wu, W., & Bulut, H. (2005). eSports: Collaborative and synchronous video annotation system in grid computing environment. In *Proceedings of the Seventh IEEE International Symposium on Multimedia,* December 12 - 14, 2005. Washington, DC: IEEE Computer Society, p. 95-103.

Appendix: A Selection of MERITS Entities

- <message>
- <module moduleID="4" moduleTitle="Interview Training">
- <items>
<itemListDescription>
A module is composed of items, which may be either "lessons" (static content) or "scenarios" (dynamic content). In either case, each item has an order within the module, and may have associated learning objectives and concepts.
</itemListDescription>
- <lesson lessonID="7" itemOrder="1">
<lessonTitle>Interview Question: Tell Me About Yourself</lessonTitle>
<lessonText>This lesson explores the interview question "Tell me about yourself."</lessonText>
- <textSegments>
<textSegment segmentOrder="1">
A good answer will: <UL> <LI>start with most significant point <LI>describe qualities and behaviour relevant to the job <LI>use examples to "paint a picture of success" <LI>mention traits that will make the applicant successful <LI>share qualities used in position <LI>the applicant will confidently say that he or she is right for the job </UL>
</textSegment>
<textSegment segmentOrder="2">
An applicant answering poorly may: <UL> <LI>giggle, which can be a sign of nervousness <LI>fail to back up claims with examples <LI>admit unpreparedness, which shows lack of seriousness <LI>say sorry repeatedly, which amplifies nervousness <LI>repeatedly use "um" and other filler words <LI>not end in a firm tone, which may cause the applicant to seem disinterested </UL>
</textSegment>
</textSegments>
- <videoSegments>
<videoDescription>
Each video entry stores basic information about the clip. A link to each video is presented within the relevant lesson.
</videoDescription>
- <video videoID="2">
<videoName>Tell Me About Yourself (good answer)</videoName>
<videoLength>00:01:30</videoLength>
<videoURL>http://www.youtube.com/watch?v=qR-IhZJOq3U</videoURL>
</video>
- <video videoID="3">
<videoName>Tell Me About Yourself (bad answer)</videoName>
<videoLength>00:01:30</videoLength>
<videoURL>http://www.youtube.com/watch?v=tDhbLdFJAF4</videoURL>
</video>
</videoSegments>
- <lessonObjectives>
<objectiveDescription>
Each objective encapsulates a particular skill or concept to be conveyed by the lesson. The objective contains a set of "outcomes", which are student inputs that indicate understanding of the skill or concept.
</objectiveDescription>
- <lessonObjective objectiveID="14" objectiveWeight="1">
<objectiveName>Tell Me About Yourself assessment</objectiveName>
- <lessonOutcomes>
<outcomeDescription>

Each outcome encapsulates a specific student input that indicates the student's understanding of a skill or concept. This outcome might be an answer on a multiple-choice question, for example, or an observation or action within a scenario.
</outcomeDescription>
- <lessonOutcome outcomeID="41" pctMark="1">
<outcomeName>Described skills and abilities</outcomeName>
<outcomeText>That is correct.</outcomeText>
</lessonOutcome>
- <lessonOutcome outcomeID="42" pctMark="0">
<outcomeName>Did not describe skills and abilities</outcomeName>
<outcomeText>No, that is incorrect.</outcomeText>
</lessonOutcome>
</lessonOutcomes>
</lessonObjective>
</lessonObjectives>
- <questions>
<questionDescription>
Each question entity stores a question and a set of answers. These answers are linked to outcomes -- and, thus, to lesson objectives. The questions are displayed within the MERITS system, and are also shown as interactive video annotations via AVA client.
</questionDescription>
- <question questionID="6" objectiveID="14">
<questionName>Tell Me About Yourself analysis</questionName>
<questionText>
Did the applicant describe skills and abilities that are relevant to the job? (good answer)
</questionText>
- <options>
- <option optionID="14" optionOrder="1" outcomeID="41">
<optionName>Yes</optionName>
<optionText>
The applicant did a good job of describing their skills and abilities.</optionText>
</option>
- <option optionID="15" optionOrder="2" outcomeID="42">
<optionName>No</optionName>
<optionText>The applicant did a poor job of describing their skills and abilities.</optionText>
</option>
</options>
</question>
</questions>
- <lessonConcepts>
<lessonConceptDesc>
A lesson may have one or more concepts with which it is associated.</lessonConceptDesc>
<lessonConcept conceptID="7" />
<lessonConcept conceptID="12" />
<lessonConcept conceptID="18" />
<lessonConcept conceptID="19" />
<lessonConcept conceptID="16" />
</lessonConcepts>
</lesson>
- <scenario scenarioID="8" itemOrder="6">
<scenarioTitle>Tell About Yourself</scenarioTitle>
- <components>

```
          <componentDescription>
A scenario is made up of components, which describe either actions taken by a programmable avatar, or reactions by this avatar to actions taken by the user. The state of a scenario determines which components are acted upon; hence the "start state" and "end state" parameters given for each component. Also, each component can be associated with a graded outcome, which will cause a grade to be recorded when that component is acted upon.
          </componentDescription>
          - <component componentID="23">
            <componentName>Tell About Yourself Speak First Phrase Good</componentName>
            <componentType>Speak</componentType>
            - <parameters>
              <parameter paramType="SpokenPhrase" paramValue="Well I have six years of medical sales experience, covering a large area to bring to this position" />
            </parameters>
          </component>
          - <component componentID="25">
            <componentName>Hear Tell About Yourself Question</componentName>
            <componentType>Listen</componentType>
            - <parameters>
              <parameter paramType="ListenPhrase" paramValue="tell me about yourself" />
            </parameters>
          </component>
            <componentName>Tell About Yourself Speak First Phrase Bad</componentName>
            <componentType>Speak</componentType>
            - <parameters>
              <parameter paramType="SpokenPhrase" paramValue="I'm hard working...<nervous laugh>"
/>
            </parameters>
          </component>
        </components>
        - <scenarioObjective objectiveID="21" objectiveWeight="1">
          <objectiveName>Tell Me About Yourself scenario assessment</objectiveName>
          - <scenarioOutcomes>
            - <scenarioOutcome outcomeID="31" pctMark="1">
              <outcomeName>Correct Observation: Tell Me About Yourself</outcomeName>
              <outcomeText>That observation is correct!</outcomeText>
            </scenarioOutcome>
            - <scenarioOutcome outcomeID="32" pctMark="0">
              <outcomeName>Incorrect Observation: Tell Me About Yourself</outcomeName>
              <outcomeText>
That observation does not match the answer given by the applicant.</outcomeText>
            </scenarioOutcome>
          </scenarioOutcomes>
        </scenarioObjective>
        - <scenarioObservations scenarioType="good">
          <scenarioObservationDescription>
A scenario may have one or more observations with which it is associated. These observations will be recognized as correct, and are linked to appropriate graded outcomes. Note that the scenario may include different cases (for example, the avatar giving either a "good" or "bad" answers), in which case the observations will only be correct for the specified case.
          </scenarioObservationDescription>
          - <observation observationID="1">
```

<keyword>start significant</keyword>
</observation>
</scenarioObservations>
- <scenarioObservations scenarioType="bad">
- <observation observationID="6">
<keyword>admit unprepared</keyword>
</observation>
</scenarioObservations>
</scenario>
</items>
</module>
- <observations>
<observationDescription>
Observations are specific characteristics of a scenario, which the student may be asked to identify. The student refers to these observations using the specified keywords.
</observationDescription>
- <observation observationID="6">
<observationKeyword>admit unprepared</observationKeyword>
<observationDescription>Admitted that they were unprepared</observationDescription>
- <observationConcepts>
<obsConceptDescription>
An observation may have one or more concepts with which it is associated.
</obsConceptDescription>
<observationConcept conceptID="15" />
</observationConcepts>
</observation>
</observations>
- <concepts>
<conceptDesc>
A concept encapsulates an key idea in a module. A concept may be independent, or may form part of a hierarchy.
</conceptDesc>
- <concept conceptID="3">
<conceptName>Attitude</conceptName>
<conceptDescription>
Encapsulates factors relating to the applicant's attitude</conceptDescription>
<noParents>No parents found for this concept.</noParents>
</concept>
- <concept conceptID="7">
<conceptName>Avoid "ums"s</conceptName>
<conceptDescription>
The applicant should avoid using "um" and other "filler" words.</conceptDescription>
- <conceptParents>
- <parent conceptID="1">
<conceptName>Communication</conceptName>
</parent>
</conceptParents>
</concept>
</concepts>
</message>