

Reusable, Lifelike Virtual Humans for Mentoring and Role-Playing

Edward M. Sims, Ph.D.
Vcom3D, Inc.
Orlando, FL
EdS@vcom3d.com

William Y. Pike
RDECOM-STTC
Orlando, FL
Bill.Pike@us.army.mil

ABSTRACT

Lifelike, interactive digital characters, serving as mentors and role-playing actors, have been shown to significantly improve learner motivation and retention. However, the cost of modeling such characters, authoring and editing their interactions, and delivering them over limited-bandwidth connections can be prohibitive. This paper describes a framework, authoring tools, and Web-based run-time environment that support the creation of training scenarios using digital virtual humans and other reusable 3D components. By conforming to the Humanoid Animation (H-Anim), Extensible 3D (X3D), and ADL Shareable Content Object Reference Model (SCORM) specifications, these 3D components are designed to promote reuse and interoperability at several levels. Recently, these software tools were used to develop prototype lessons in foreign language and cultural familiarization for use at the Defense Language Institute Foreign Language Center (DLI/FLC). These lessons include simulations in which the student, taking the role of a U.S. soldier, interacts with the local population of a foreign culture in authentic situations, using both English and the local dialect. Digital virtual humans are used to represent not only the soldier, and the role-playing actors; but also a mentor who reviews language and cultural learning points, and provides remediation. Using these prototype lessons as examples, we review the potential advantages of the technology in other training applications.

ABOUT THE AUTHORS

Ed Sims is CTO and co-founder of Vcom3D, an Orlando-based company that develops interactive visualizations for Web-based learning. Dr. Sims holds a B.S. in Mathematics from the College of William and Mary and the M.S. and Ph.D. degrees in Systems Engineering from Rensselaer Polytechnic Institute. He has been awarded five patents in the area of real-time visual simulation.

Bill Pike is a Science & Technology Manager for Creative Learning Technologies at the US Army Research, Development and Engineering Command's Simulation and Training Technology Center in Orlando, FL. He has led research projects in advanced distributed learning technologies such as intelligent tutoring systems, game engine-based simulation, and handheld training platforms. Mr. Pike earned his Master's Degree in Computer Engineering from the University of Central Florida. He holds a Bachelor's Degree in Systems Science from the University of West Florida and is currently working toward a Ph.D. in Modeling and Simulations at the University of Central Florida. He is also a Naval Reserve officer, currently assigned to US Special Operations Command at MacDill Air Force Base, FL.

Reusable, Lifelike Virtual Humans for Mentoring and Role-Playing

Edward M. Sims, Ph.D.
Vcom3D, Inc.
Orlando, FL
EdS@vcom3d.com

William Y. Pike
RDECOM-STTC
Orlando, FL
Bill.Pike@us.army.mil

INTRODUCTION

Interactive virtual worlds provide a compelling and effective medium for experiential “learn-by-doing” training. Both the motivational and instructional aspects of this medium can be further enhanced in many applications when these worlds are populated by life-like virtual humans that participate in delivering instruction, or act out roles in an interactive story.

In this paper, we describe the capabilities, authoring and run-time environments, and example applications for a class of reusable virtual human actors that have been designed to serve as mentors, collaborators, and role-playing actors in Advanced Distributed Learning (ADL) courses. This framework for developing instructional applications of virtual humans addresses significant issues that have posed technical and economic barriers in the past.

ROLES AND REQUIREMENTS FOR VIRTUAL HUMAN ACTORS

Roles for Virtual Human Actors

The potential uses for virtual humans in instructional simulations are nearly limitless, but it is helpful to distinguish some primary roles, and the capabilities needed to support these roles. We characterize these roles as mentor, teammate (or collaborator), and role-playing actor.

As a *mentor*, a virtual human’s goal is to provide knowledge, or to ask questions that guide the student to discover and construct knowledge. Such mentors may:

- Demonstrate how to accomplish a task.
- Provide feedback on the student’s performance.
- Encourage the student to try new approaches.

Lester et al. (1997) have shown that by providing an embodied presence for a mentor, it is possible to significantly enhance the students’ motivation and perception of the learning experience – an effect that they have called the *persona effect*.

As a *teammate*, a virtual human assists in collaborative skills that require that two or more individuals work together to perform a task. Examples range from assembly and maintenance tasks to team tactics. The virtual teammate makes it possible for the student to practice collaborative tasks, even though a real collaborator is not available. In most applications, the virtual teammate will not only need to act in coordination with the student, but also to communicate instruction, intentions, or problems encountered.

As a *role-playing actor*, a virtual human acts out a role in an interactive story that teaches the student some aspect of human communication or interaction. Some of the most important experiential training needs involve learning how to interact with other persons to discover their identity, background, and intents; to solve a problem for them, or to elicit a response or action on their part. For example, a student might interview or interrogate a virtual human at a checkpoint or port of entry. In this application, the virtual humans will need to interact not only with the student, but also with each other.

In each of these roles, the virtual humans serve as “stand-ins” for roles that might otherwise be played by real human instructors or fellow students, who are not available due to insufficient local availability of instructors or fellow students with the required skills; or due to the high cost of recording, editing, or delivering live-action video. To serve in these roles, the virtual humans must portray high fidelity, convincing, lifelike behaviors.

Requirements for Virtual Human Actors

From the preceding discussion, it is clear that, in order to perform as mentors, teammates, and actors, virtual humans need to be capable of a wide range of logically, physically, and visually convincing behaviors, including interactions with the student, with objects in the environment, and with each other. Among the most important of these interactions is speech. However, this speech should be coordinated with non-verbal communications and actions. As shown by Koneya and Barbour (1976), in face-to-face

communication, only 7% of all information is communicated in words. The rest is communicated in gesture, body language, facial expression, and intonation.

Ideally, a virtual human actor should be able to perform tasks when given the same types of commands that a director would give to a live actor, for example “Hand your ID to the soldier, look and point to the south, and say ‘I have a family in the next town’ with a concerned expression.” These commands should be in the form of simple, declarative statements that the virtual humans “understand” without needing detailed descriptions of procedures.

Based on the evaluation of several use scenarios, we identified the list of basic capabilities shown in Table 1, which summarizes the actions that all virtual human actors developed for this system will need to perform, and upon which higher-level behaviors are based.

Table 1. Virtual Human Basic Capabilities

Action	Capabilities
Speech	Speak recorded voice in any language, or synthetic voice using common commercial text-to-speech “engines”, with proper lip-sync.
Facial Expression	Form facial expression representing the seven basic emotions, plus 33 grammatical markers (e.g., questioning, introduction of new topic)
Focus of attention	Direct gaze toward any point in space, or toward any object or person.
Gesture	Use any gesture selected from a library of over 2000 one- and two-arm motions, including parameterized motions such as “point at” or “wave to”.
Manipulation of Objects	Grasp, manipulate, and connect designated objects in the environment.
Posture	Shift posture to illustrate a point, to conceal the body, to prepare for an action, or just to maintain balance.
Locomotion	Walk or run between any two commanded locations.

These actions are not all inclusive of the actions that a virtual human might take. However, for a broad range of training tasks, they provide the basic components from which the majority of more complex behaviors can be composed. One key to simplifying the description of virtual human actions in this system is that the methods for handling objects (e.g., tools, equipment, and

weapons) are included the models of those objects. Even when tasks that cannot be composed from these base actions must be performed, it is possible to prototype most of a training scenario from the actions, and to add the additional behaviors after a rapid prototype of the rest of the scenario has been created.

A COMPONENT-BASED FRAMEWORK FOR REUSABLE 3-D COMPONENTS

Before describing the implementation of the virtual humans, it is helpful to first describe the 3D component framework of the virtual worlds in which they act. The infrastructure upon which we have created this framework is based on three specifications – X3D, SCORM, and XML - that promote interoperability, reuse, and extensibility.

Extensible 3D (X3D) (Web3D Consortium, 2004) is a successor to the Virtual Reality Modeling Language (VRML97). To meet the evolving requirements of Web-based training, modeling and simulation, and other market areas, X3D has adopted the following design objectives:

- Separate the runtime architecture from the data encoding
- Support a variety of encoding formats, including the Extensible Markup Language (XML)
- Add new graphical, behavioral, and interactive objects
- Provide alternative application programmer interfaces (APIs) into the 3D scene
- Modularize the architecture into components
- Define subsets of the specification (“Profiles”) that meet different market needs
- Allow for the specification to be implemented at varying levels of service

X3D provides an excellent basis for a Web-based runtime for 3D Learning Objects. Primary advantages of the X3D standard include:

- It is backwards compatibility with existing VRML97 content.
- The XML encoding is extensible, is compatible with most content management systems, and can be transformed to other formats using Extensible Style Sheet transformations (XSL-T).
- APIs are provided for Web-page integration.

However, specifying the X3D language as the “standard” for 3D component implementation is not sufficient. By its nature, X3D supports multiple APIs and many methods of aggregation, as well as two distinct encodings (VRML97 and XML). What is

needed is an X3D Profile tailored to the needs of Modeling and Simulation and Advanced Distributed Learning. This is fully consistent with the goals and specification of X3D. Such a profile can be used to enforce:

1. A standard encapsulation of components that promotes reuse and aggregation, and that hides unused internal structures.
2. A required level of service that supports common simulation requirements.
3. A standard API for communication between 3D components and SCOs. This API must describe not only message and data types, but also their use and meaning.

We have developed a 3D component framework as a synthesis of several existing standards, where:

- **X3D** is the language in which 3D components are described.
- **SCORM** guides the requirements for metadata and structure for aggregating assets.
- **XML** provides a unified encoding for both 2D and 3D content, as well as metadata.

Furthermore, we have defined and implemented extensions to X3D that support object appearances and behaviors commonly found in military simulations. To date, we have created the following new node types:

- A Gravity Node provides proper terrain following for vehicle and human models moving over the terrain.
- Collision and Collidee nodes test for inter-object collision and visibility.
- A LayerGroup node allows coplanar surfaces to be assigned rendering priorities.
- A MultiTexture node allows multiple texture images to be combined and assigned to a single surface.

In this synthesis of standards, a 3D component is:

- a **SCORM Asset** represented as an **XML-encoded X3D File**,
- where each X3D object and its behaviors are wrapped into a **PROTO** that can be aggregated into higher constructs through a **SCORM Manifest**,
- with associated **SCORM metadata** (appropriately extended),
- with optional sound and image **media files**, as well as optional **SCRIPT Nodes** that represent complex behaviors,
- and which exposes a consistent run-time **API**.

Figure 1 illustrates the elements of the 3D component architecture.

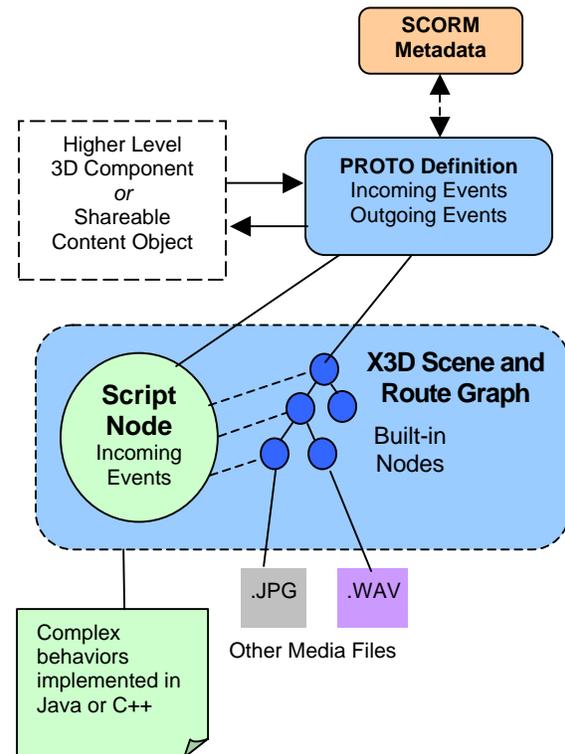


Figure 1. 3D Component Architecture

Basic behaviors can be represented in XML syntax within the X3D scene graph, using the 53 built-in node types. More complex behaviors may need to be represented in a full-featured programming language, such as Java or C++. The X3D standard provides a mechanism to embed these complex behaviors using a Script node, which includes fields for incoming and outgoing events, as well as a reference to an URL that contains the program code. The program can perform a wide variety of actions including sending out events (and thereby changing the scene), performing calculations, and communicating with servers elsewhere on the Internet.

Using this component architecture, we have developed complex classes of objects, including terrains, vehicles, and virtual humans with corresponding “animation engines” – behaviors implemented in Java or C++ and attached to Script Nodes. These animation engines implement complex behaviors that may be performed by any instance of the corresponding class of objects.

Figure 2 illustrates the synergy and fine-grained integration that this approach allows.

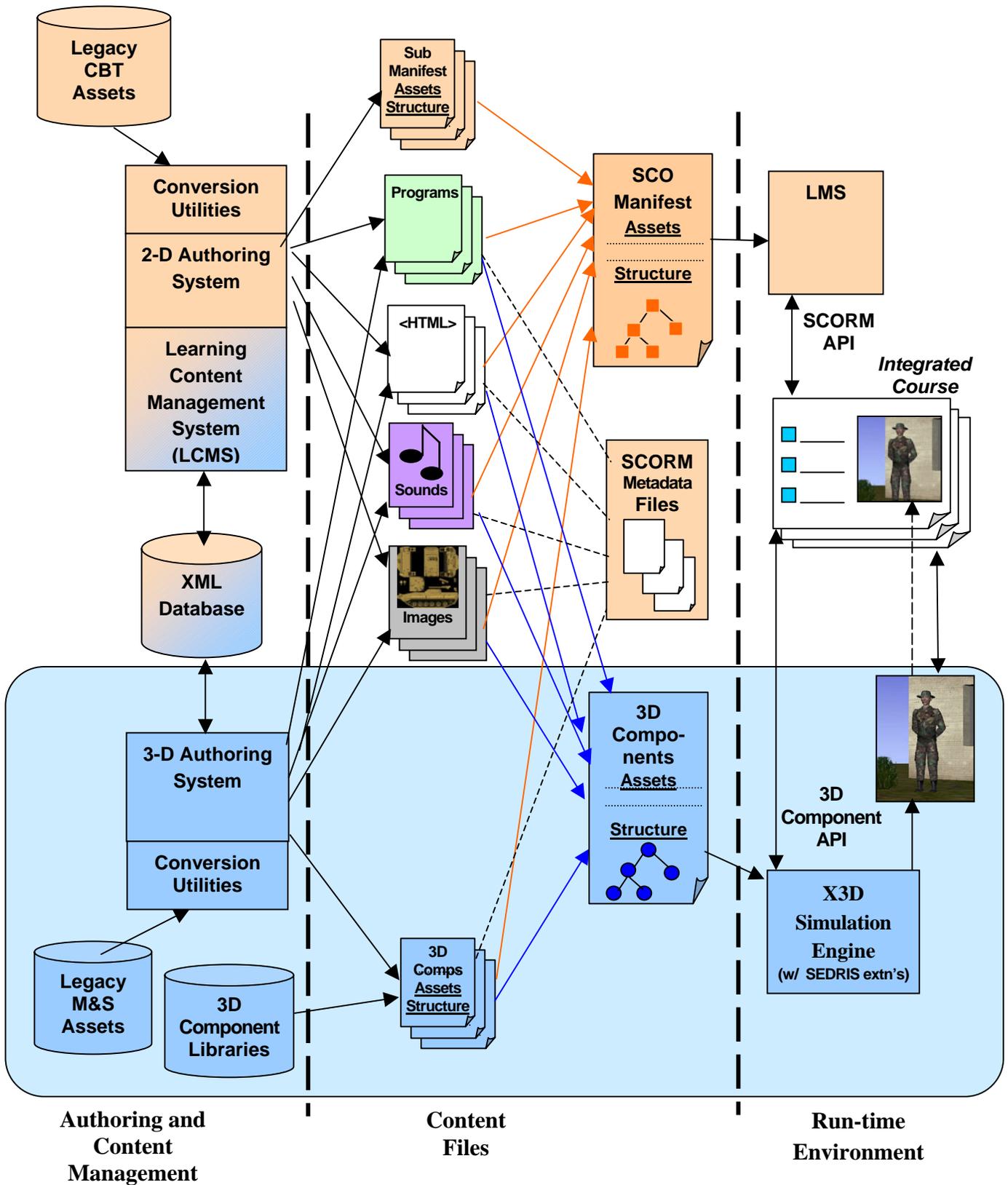


Figure 2. 3D Simulation Component Framework

Referring to Figure 2, the elements of traditional 2D Advanced Distributed Learning (ADL) content are color-coded in orange. In a standard ADL implementation, Web pages are authored using a **2D Authoring System** that typically works together with a **Learning Content Management System (LCMS)**. The Authoring System is used to create a series of "Learning Objects" stored in an **XML Database**. For Web delivery, these objects are transformed into HTML pages that may have embedded **images**, **sounds**, and even small **programs**, in the form of Java applets, ActiveX controls, or JavaScripts. Each deliverable course asset is paired with an XML **SCORM Metadata file** that provides information on the asset's type, origin, intent, and relationship to other assets. This metadata is used to search for appropriate assets, simplifying reuse, as well as content management. A **SCORM Manifest**, also in the form of an XML file, identifies hierarchical structures of assets. A Shareable Content Object (SCO) is a self-contained learning object, with all of its supporting files, that provides a JavaScript interface, called the **SCORM API**, to a **Learning Management System (LMS)**. After launching a SCO, a student navigates through a sequence of Web pages. Depending on the student's actions and responses, scoring and other information may be provided back to the LMS through the SCORM API.

The elements that we add to this framework to create a 3D Learning Environment are highlighted in blue. This environment has parallel Authoring and Content Management, Content Data, and Run-time segments. **3D Components** are authored using a **3D Authoring System**, or may be converted from legacy Modeling and Simulation (M&S) Assets. 3D Components, like SCORM manifests, are XML files that describe 3D assets, and that may be aggregated hierarchically into a top-level 3D component that interfaces with a SCO through the 3D Component API. They represent 3D objects with geometries and appearances, which may be enhanced by *images*, *sounds*, and embedded *programs* that define complex behaviors. Unlike the Web pages of a SCO, which are viewed sequentially, 3D objects are generally presented simultaneously in a single 3D world, which is rendered by an X3D Simulation Engine (or "browser") that is usually embedded in a Web page.

RAPID DEVELOPMENT OF REUSABLE, REALISTIC HUMAN INTERACTIONS

Humanoid Animation

Within this framework, virtual human actors are one class of 3D component. They are interchangeable in that any behavior developed for one human model can be performed any other human model – at least within the limits of that model's physical size and limitations. We achieve this interchangeability and interoperability by developing all human models to a common Humanoid Animation (H-Anim) standard.

The Humanoid Animation (H-Anim) Working Group (Humanoid Animation, 2004; Macchi and Sims, 2002) of the Web3D Consortium is developing a standard way of representing humanoids, such that they may be interchanged among modeling, authoring, and run-time applications. (The term "humanoid" was adopted to indicate that the standard is broad enough to encompass other human-like creatures or robots.) The H-Anim specification defines a Humanoid node consisting of Segments connected by articulated Joints. Interchangeability is standardized by specifying:

- a standard hierarchy and naming convention for the joints and segments,
- a standard way of describing the geometry and appearance of each segment,
- a standard interface for controlling the articulation of the joints, and
- a standard default posture for the humanoid.

The H-Anim specification defines a hierarchy of 89 joints, with 89 corresponding segments. A simplified, 18 joint/segment hierarchy is shown in Figure 3, using the convention that a colon (:) separates each joint/segment pair.

```

HumanoidRoot : sacrum
sacroiliac : pelvis
|  l_hip : l_thigh
|  |  l_knee : l_calf
|  |  |  l_ankle : l_hindfoot
|  |  |  |  l_midtarsal : l_middistal
|  r_hip : r_thigh
|  |  r_knee : r_calf
|  |  |  r_ankle : r_hindfoot
|  |  |  |  r_midtarsal : r_middistal
v15 : 15
skullbase : skull
l_shoulder : l_upperarm
|  l_elbow : l_forearm
|  |  l_wrist : l_hand
r_shoulder : r_upperarm
|  r_elbow : r_forearm
|  |  r_wrist : r_hand

```

Figure 3. Simplified H-Anim Hierarchy

A compliant H-Anim model does not need to have all of 89 of the specified joint/segment pairs, which include 5 lumbar vertebrae, 12 thoracic vertebrae, and 7 cervical vertebrae, as well as 19 joints in each hand. However, any joint or segment that is included must use the prescribed naming convention, and no additional joints or segments may be added within the prescribed hierarchy. (The specification does allow, however, for “leaf” nodes to be added, representing articles of clothing, jewelry, tools, etc.) In this way, it is assured that animations and other run-time controls developed for one model may be applied to another, at least for all of the shared joints of the two models.

A Segment node is defined by:

- a segment Name,
- a Mass,
- a Center of Mass,
- a Moment of Inertia,
- a Bounding Box Center and Size, and
- one or more Children nodes, typically X3D Shape nodes, which may include such geometries as boxes, spheres, and indexed face sets.

A Joint node is defined by:

- a joint Name,
- a Rotation and Translation,
- a Scale and Scale Orientation,
- a Center (of Rotation),
- Upper and Lower Limits of Rotation, and
- some Children nodes, typically only the corresponding Segment node.

A Humanoid node stores Metadata about the Humanoid Model, as well as references to the Segment and Joint nodes of the model.

The last element of the specification is a Default Posture: the posture that the virtual human takes when all joint angles are set to zero. The Default Posture for an H-Anim model, as shown in Figure 4 is:

- Standing; arms at side
- Fingers down; thumbs at 45 degrees
- Eyes ahead and open; Eyebrows neutral
- Mouth closed
- Eyebrows neutral

Additional information on H-Anim can be found in (Humanoid Animation Working Group, 2004).

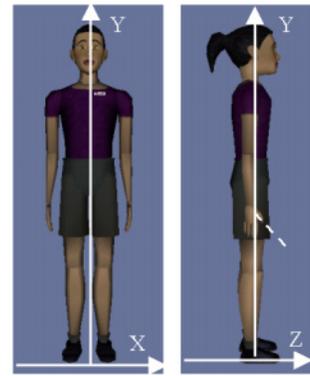


Figure 4. H-Anim default posture.

Using the H-Anim specification, Vcom3D has developed a library of library of over 30 interchangeable characters, aged 10 to adult, of both genders, various races and ethnicities, and representing military, law enforcement, and civilian personnel. Representative models are shown in Figure 5. Each character has 57 articulated joints that control a seamless, morphable skin.



Figure 5. Representative H-Anim Human Models

In its current form, the H-Anim specification only defines interfaces at the lowest level: motions of joints and control points. Although this is sufficient to provide a level of portability and interoperability, it is not the level at which we would normally command human actions. Therefore, we have extended the H-Anim requirements to include higher level behaviors, including those identified in Table 1. Lip-synched speech is supported by including 22 visemes, or mouth shapes for each model. These mouth shapes model the sounds of the International Phonetic Alphabet, allowing the characters to speak virtually any language or dialect. In addition, each virtual human model includes:

- 40 Facial expressions
- Over 100 pre-defined handshapes for gestures and grasping
- Inverse kinematic control of the hands/arms for goal-directed actions

We have also developed a library of over 2000 interchangeable behaviors for the characters, including common gestures, ambient motions (“fidgets”), look-at, point-at, and grasping behaviors, and walk cycles.

Authoring Content for Virtual Humans

Scenario-based simulations are developed from branching storyboards composed of four types of scenario segments: Explanatory, Role-playing, Feedback, and Remediation. A graphical user interface is provided to construct the course logic using these segments, together with student response nodes, as shown in Figure 6.

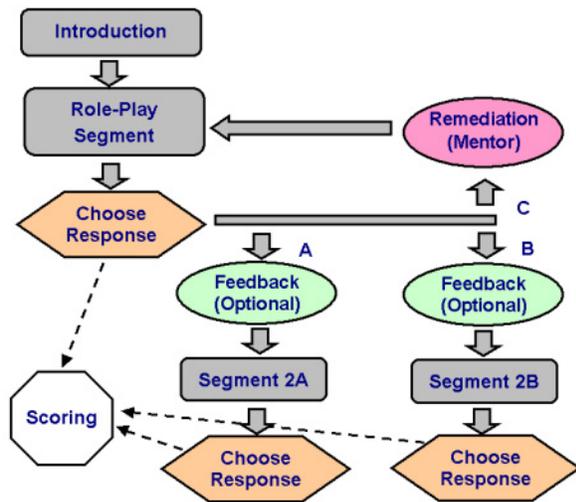


Figure 6. Course Logic Design.

Each segment, in turn, is defined by a Script for each virtual human character. Each script includes a track for:

- Speech (with lip-sync)
- Facial expression
- Focus of Attention
- Gesture and Posture
- Object Manipulation

These segments are authored using Vcom3D’s Vcommunicator® Authoring Tool, shown in Figure 7.

Authoring a scenario segment starts with importing a voice file and corresponding transcription. A lip-sync tool is used to align the transcription with the speech. This transcription is displayed along a time line, shown as the lowest, yellow timeline of a “score” in Figure 7. (If the character speaks a language that is foreign to the author, especially if the language uses a non-Roman alphabet, it may be transcribed into an English approximation, so that the animator can more easily align the animation with the speech.) To date, the system has been tested with English, Spanish, Portuguese, Korean, and Arabic (Egyptian, Levantine, Iraqi, and Modern Standard) with favorable results. It is also possible to use text-to-speech, instead of recorded voice.

The author then adds facial expression, focus of attention, gesture, posture, locomotion, and other animations using the other timelines of the score. A few facial expressions, including questioning, are derived automatically from the speech transcription. These may be accepted or overridden by the author.

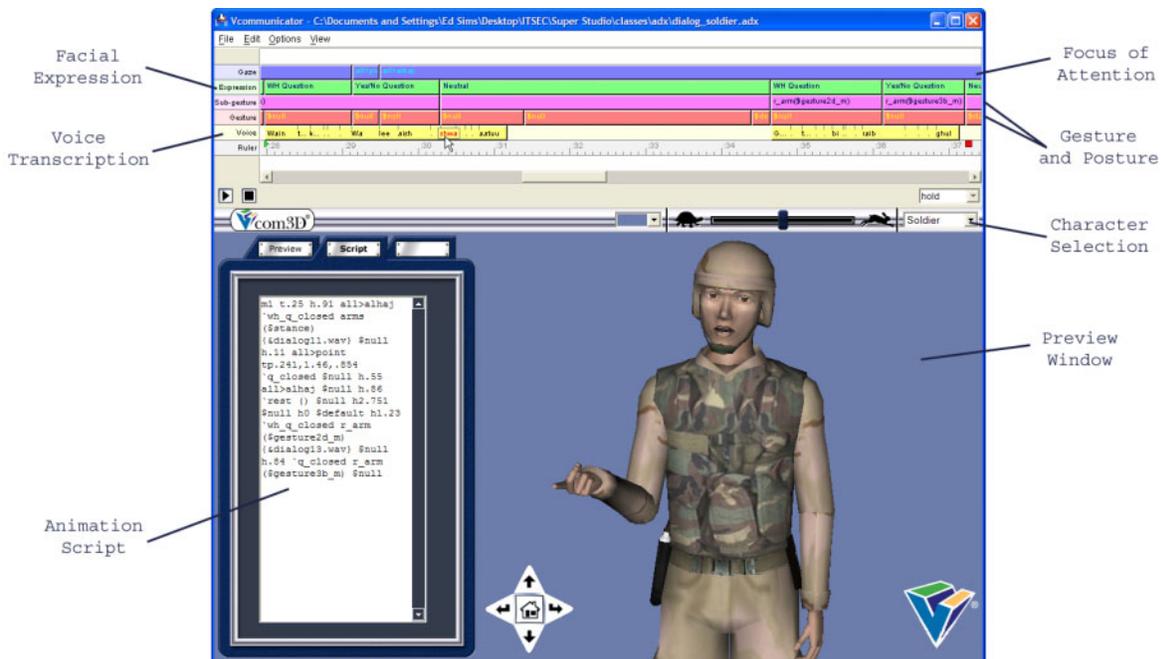


Figure 7. Vcommunicator Authoring Tool

APPLICATION: ONLINE MENTORS FOR FOREIGN LANGUAGE TRAINING AND CULTURAL FAMILIARIZATION

In June 2003, Vcom3D, together with the Naval Postgraduate School and Defense Language Institute, was awarded a contract by the Joint ADL Co-Lab to develop a prototype course to teach tactical language and cultural familiarization. Under this project, entitled Online Mentors for Foreign Language Training and Cultural Familiarization, the authoring tools of the 3D components and virtual humans framework were used to develop an exemplar interactive training simulation.

The objective of this course is to teach elements of the Iraqi dialect, along with nonverbal language, culture, and customs that would be most essential to military personnel stationed in Iraq. The course uses an immersive, scenario-based training simulation, in which the student takes the role of a soldier manning a checkpoint on a road into Baghdad. As the scenario progresses, the student interacts with four members of an Iraqi family and with a fellow American soldier.

The Arabic language is characterized by significant variations in dialect among the many countries in which it is spoken. Customs and culture also vary among these countries. The target audience for the course is military personnel who have had at least one semester of Modern Standard Arabic, but who have not yet studied dialectic variations and have not been exposed to interactions with native Iraqis.

In order to be successful in the course, the student must understand and respond correctly in the Iraqi dialect, and must also recognize nonverbal cues, avoid inappropriate gestures, and follow culturally correct protocol in interacting with family members.

At different times, the student is asked to analyze what he or she has observed, and to select a course of action based on these events. If the student does not select the best course of action, the scenario unfolds differently and events can get out of hand.

Virtual humans serve not only as role-playing actors, but also as an instructor/mentor who presents important elements of dialect and culture, who explains best courses of action, and who provides feedback on selections selected by the student. Topics covered in the course include:

- Dialectical variations in the formation of plural verbs.
- Appropriate and inappropriate use of honorific and family names.
- Colloquial terms for policemen, soldiers, and strangers.
- Iraqi gestures that may be misinterpreted by Americans, and American gestures that may be misinterpreted by Iraqis.
- Methods to calm tense situations.
- Proper and improper ways to interact with Iraqi women.
- Showing respect for family relationships.

NPS, DLI/FLC, and Vcom3D personnel first worked together with the Joint ADL Co-Lab to create a storyboard that illustrated the primary topics to be taught. This scenario was first enacted by instructors and students at DLI/FLC, as shown in Figure 8. A video recording of this enactment served to guide the Vcom3D animators in creating the simulated scenario. English and Iraqi dialog were recorded separately in a studio environment.

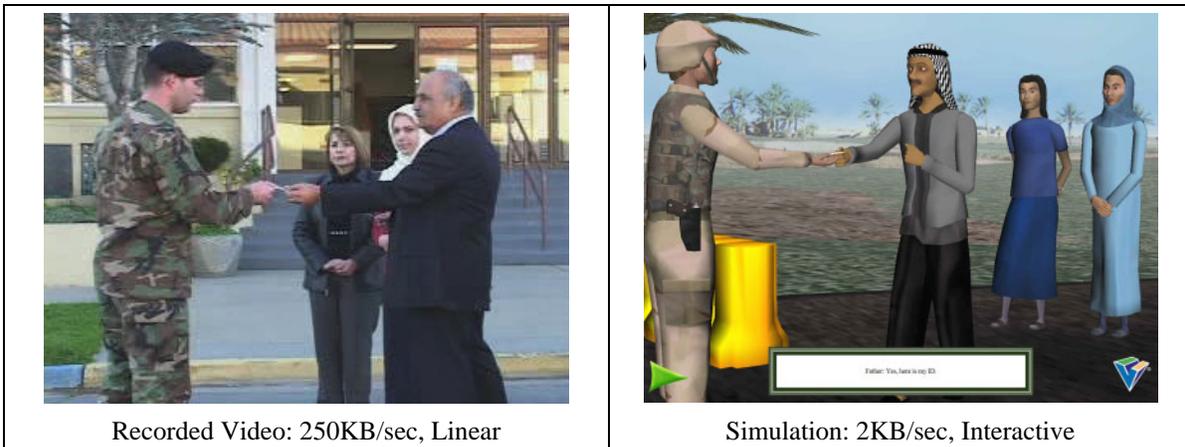


Figure 8. Comparison of Video and Interactive Simulation

RESULTS, CONCLUSIONS, AND PLANS

At the time of publication, we have used the 3D components and virtual humans authoring tools and run-time software to create an initial 7-minute interactive scenario, and we are in the process of creating a longer scenario with 20 minutes of action that is expected to take approximately 30 minutes to be completed by each student.

By developing the language and culture familiarization application in parallel with the development of the 3D Components and Virtual Humans authoring and run-time environments, we have been able to identify needs that were not initially recognized, and to enhance the authoring and run-time software.

Effectiveness of the Authoring Environment

One real payoff in using virtual humans for the language and culture familiarization application has been the ease of modifying and correcting the content. For the duration of the project, the development team has remained in Orlando, while the Subject Matter Experts (SMEs) have remained in Monterey, California. All voice and video recordings have been transmitted electronically between the two locations. Although a significant number of character behaviors and appearances needed to be revised, based on feedback from the DLI SMEs, the changes were accommodated easily by recording individual voice files, modifying one aspect of a character's clothing, or changing a specific gesture or expression.

Also of interest has been the effectiveness and reusability of the behavior library from which the animations have been created. Using the base library of about 2,000 behaviors, we were able to create over 95% of the total scenario animation. Many of the animations that were created specifically for the remaining 5% of this scenario, including such actions as "get out of car", "point rifle", and some gestures specific to the region, are potentially reusable for other scenarios, and have been added to the behavior library.

Bandwidth considerations

A second objective of the 3D components run-time environment has been to provide high quality simulations over low-bandwidth connections, such as may be found in remote areas to which the training will be delivered. As indicated in Figure 6, once the simulation environment and character library has been installed (approximately 6MB), the course

content can be delivered over low-bandwidth connections. The primary component of this required bandwidth is the recorded voice, which requires 11kbits/sec. The simulation commands require only about 5kbits/sec.

Plans

During late summer and early fall 2004, we will be evaluating the training effectiveness of the language and culture familiarization scenario. These will be published as part of the ADL Prototype documentation.

On a separate project funded by DARPA, we will be exploring methods for rapidly modifying courses for use with other languages. For example, while maintaining the overall course outline and storyboard for the Iraqi checkpoint scenario, we will change the target language and culture to Farsi or Kurdish, which are minority languages in Iraq.

ACKNOWLEDGEMENTS

The 3D Components Framework and X3D Viewer/Simulation Engine were developed in part under funding from US Department of Defense SBIR contracts N61339-00-C-0006 and N61339-02-C-0156.

The facial expression and lip-synchronization capabilities were developed in part under funding from NSF Grant 9901855, and were extended to include multiple language capability under US Department of Defense SBIR contract N61339-02-C-0156.

The Online Mentors for Language Training and Cultural Familiarization is being funded as an ADL Prototype by the Joint ADL Co-Lab. Collaborators on this project include Dr. Don Brutzman, Associate Professor, and Jeffrey Weekley, Research Associate, NPS Moves Institute; Luba Grant, Dean, Asian School I, DLI/FLC; and Dr. Renée Stout, Joint ADL Co-Lab. Subject Matter Expertise has been provided by the Arabic Faculty at DLI/FLC. The Joint ADL Co-Lab funds have been supplemented by SBIR matching funds from Department of Defense contract N61339-02-C-0156.

Opinions expressed in this paper are those of the authors and do not necessarily represent the views of these sponsoring organizations.

REFERENCES

- Arguello, L., et al. (2001). On Integration of Simulations into ADL Systems for International Space Programmes. *European Simulation Interoperability Workshop*.
- Blais, C., D. Brutzman, J. Weekley, and J. Harney. (2002). Emerging Web-based 3D Graphics for Education and Experimentation. *Proceedings of IITSEC 2002*.
- Brutzman, D., M. Zyda, M. Pullen, and K. Morse. (2003). *Extensible Modeling and Simulation Framework (XMSF) Opportunities for Web-based Modeling*.
- Humanoid Animation Working Group (2004). *ISO/IEC FCD 19774:200x, Humanoid animation (H-Anim)*. Retrieved June 24, 2004, from <http://www.h-anim.org>.
- Koneya, M., and A. Barbour. *Louder Than Words...Nonverbal Communication*. (1976). Charles E. Merrill Publishing Company.
- Lester, J. C., Converse, S. A., Kahler, S. E., Barlow, S. T., Stone, B. A., & Bhogal, R. S. (1997). *The persona effect: Affective impact of animated pedagogical agents*. Proceedings of CHI '97.
- Macchi, W., and E. Sims. (2002). Creating Interchangeable Human Models Using the SEDRIS DRM and H-Anim, *Fall 2002 Simulation Interoperability Workshop*.
- Shareable Content Object Reference Model (SCORM) Version 1.2, *Advanced Distributed Learning Initiative*, October 1, 2001.
- Marsella, S., W. Johnson, and C. Labore. (2000). Interactive Pedagogical Drama. *Proceedings of the 4th International Conference on Autonomous Agents*, 2000, 301-308.
- Web3D Consortium. (2004). *X3D Specification*. Retrieved June 24, 2004, from <http://www.web3d.org>.
- Zimmerman, P., and P. Dodds. (2002). Possibilities with Simulation Assisted Learning. *Fall 2002 Simulation Interoperability Workshop*.