Supporting the Warfighter Training Community's Use of Game-Based Training

Chris Guin, Virginia Travers, Bruce Roberts, Ray Tomlinson

BBN Technologies
Cambridge, Massachusetts
{cguin, travers, roberts, tomlinso}@bbn.com

LtCol Dean Marvin, USMC (Ret.)

Joint Advanced Distributed Learning CoLab
Orlando, Florida
Dean.Marvin2@us.army.mil

ABSTRACT

The use of games for training is increasing in the Defense community; however, this type of training can ignore the best interests of trainees and trainers. DARPA's DARWARS program sought to address this deficiency by combining competency-driven access to game-based team training with the re-use of training content. For example, DARWARS Ambush!, a game-based training environment developed for DARWARS, allows warfighters to create scenarios that re-create their battlefield experience. Then, using DARWARS graphical tools, warfighters tag scenarios using meaningful search terms, and publish the scenarios to a registry. Instructors search the registry for scenarios that address their unit's training requirements. Once selected, a scenario is automatically integrated with an appropriate Shareable Content Object Reference Model (SCORM) course. Then the DARWARS delivers both the course and the training scenario. Assessments from the training session are reported to the DARWARS Learning Management System (LMS), which uses the assessments to direct trainees to appropriate new or remedial training.

This paper describes how we integrated the following technological advancements in an end-to-end demonstration for the benefit of the warfighter training community: (a) The DARWARS Ambush! convoy trainer, developed under DARPA sponsorship, provides PC-based, simulation-based training for small teams. Ambush! is in use at multiple military installations, and has trained over 30,000 warfighters. (b) The Advanced Distributed Learning Registry (ADL-R) is a searchable index of content metadata that can be resolved to content in distributed repositories. The primary purpose of the ADL-R is to support searching for, discovering, and re-using learning content. (c) A framework for simulation-based training integrated with courseware targeting individuals and small teams. Assessments from the simulation-based training are reported back to a SCORM-conformant LMS, where the results are used to direct each team member to new or remedial training. The Joint Advanced Distributed Learning CoLab (JADL) sponsored this work.

ABOUT THE AUTHORS

Chris Guin is a software engineer at BBN Technologies. He is a developer for both the DARWARS project and the JADL Integrated Prototype Architecture (IPA) project, and has developed a low-cost gaze tracking system. Mr. Guin has a Master of Science in Computer Science from Tufts University, and a Bachelor of Science in Computer Science from Harding University.

Virginia Travers joined BBN Technologies in 1975. She is a Senior Scientist, currently working as the Principal Investigator for the Joint ADL Co-Lab project, focused on integrating simulation-based training with the Sharable Content Object Reference Model (SCORM) Run-Time Environment. She is also the technical lead for the DARWARS project. Ms. Travers has worked on a variety of projects for BBN, both as a technical contributor and project manager, including the Conning Officer Virtual Environment (COVE) project (a simulated training environment for ship handling), and the DARPA sponsored

Advanced Logistics Program (ALP) and Cougaar (Cognitive Agent Architecture) projects for developing distributed agent-based technologies. Ms. Travers holds a Bachelor of Science in Computer Engineering from Case Western Reserve University.

Bruce Roberts is a Division Scientist at BBN Technologies, where he has led the development of a number of simulation-based intelligent tutoring systems. These include an operational tutor for teaching flight-line troubleshooting to Air Force technicians, and a virtual environment based, ship-handling tutor (COVE). His interest in intelligent tutoring systems extends to some of the earliest work incorporating simulation, computer graphics, and expert systems; namely, STEAMER, which taught principles of operation for steam propulsion plants on Navy ships. As a researcher at the MIT Artificial Intelligence Laboratory he implemented one of the first practical knowledge representation systems. He was the Principal Investigator for DARWARS architecture and integration, and led the rapid development and deployment of DARWARS Ambush!, a multi-player game-based training system for road-convoy operations.

Ray Tomlinson joined BBN Technologies in June 1967. He is a Principal Engineer, currently working on the JADL IPA Project. He is also working on the Tanker, Airlift, and Special Mission (TASM) project, which will provide testing and data services for mission flight planning, and on the Intelligent Learning project, which is advancing the state of machine intelligence by learning to perform tasks based on observation of an expert demonstration. Previous projects include work on the design of several network protocols (TCP/IP, TELNET, FTP) and their implementations. Other projects include the DARPA sponsored ALP and Cougaar projects for developing distributed agent-based technologies. Mr. Tomlinson wrote the first network email program and sent the first network email message. Mr. Tomlinson holds a Bachelor of Science in Electrical Engineering from Rensselaer Polytechnic Institute, and a Master of Science in Electrical Engineering from the Massachusetts Institute of Technology.

Dean Marvin is a retired Marine Lieutenant Colonel who works as a Senior Military Analyst at the Joint Advanced Distributed Learning CoLab, in Orlando, Florida, where he helps to lead Research and Development efforts. Dean has a BA from Florida Southern College and an MAE from Chapman University.

Supporting the Warfighter Training Community's Use of Game-Based Training

Chris Guin, Virginia Travers, Bruce Roberts, Ray Tomlinson
BBN Technologies
Cambridge, Massachusetts
{cguin, travers, roberts, tomlinso}@bbn.com

LtCol Dean Marvin, USMC (Ret.)
Joint Advanced Distributed Learning CoLab
Orlando, Florida
Dean.Marvin2@us.army.mil

INTRODUCTION

Simulation-based training is recognized as an effective and efficient supplement to classroom and field training. Such training can rapidly disseminate lessons learned from the field, filling the need to address evolving, current threats. The promise of game-based training platforms and tools for authoring new content is creating new opportunities for bringing real-life experiences to the classroom, and this has special import to units preparing to deploy.

Imagine a warfighter returning from the field armed with new lessons learned.[1] Working with a military training specialist he quickly outlines his experiences in a recent convoy mission. Using user friendly authoring tools, within a few hours the trainer crafts a scenario in the game-based training system. That afternoon, at their next training session, trainees "experience" the returning warfighter's scenario.

In the meantime, another unit, in another location, prepares for deployment. How can the warfighter rapidly and effectively disseminate his lessons learned to a broader community? Using currently available tools, including local repositories, learning management systems (LMSs), and the ADL-Registry (ADL-R), we have addressed this problem, demonstrating a framework for efficiently cataloging, searching, and delivering game-based training to the warfighter as part of an integrated training package.

Using this framework, a military trainer publishes the new scenario to a repository. The repository, which is registered with the ADL-R, stores that scenario as part of a training package. With user friendly authoring tools, the instructor composes metadata that describe the training package, then uploads it to the ADL-R. Other instructors can search through this metadata, discover the training package, and with appropriate permissions download a self-installing package from the repository into their training environments. Instructors can then direct trainees to their learning management systems, which present trainees with both web-based materials specific to their roles, and general briefing material, including descriptions of the situation, maps, commander's intent, and the mission—all of which have been authored by warfighters with recent field experience. After reviewing the appropriate materials, trainees are automatically placed in roles in the simulation, where instructors observe and score the performance of trainees. The LMS records trainees' scores, which it uses to deliver appropriate remedial material to each trainee, as well as to assist instructors in conducting after-action reviews.

These training scenarios require a framework designed for both reusability and interoperability. Reusability addresses the requirement that warfighters without programming experience be able to assemble and publish game-based lessons learned that are easily located and disseminated. Likewise, interoperability addresses the requirement that a wide selection of new and existing game-based training systems and more traditional training materials be supported through the use of standardized interfaces.

We have demonstrated the framework described above using the following components:

- DARWARS Ambush!, a game-based convoy training system with a widespread and active user community.
- The ADL Registry, a publicly accessible web-based registry of training content. The Registry allows content providers to register their own content repositories, as well as publish metadata (authoring information, descriptions, relevant training objectives, etc.) describing the available training content. Instructors may then search the Registry using the context provided by the metadata.

[1] The term is used to describe both lessons observed and approved/vetted lessons learned.

- A modified Sharable Content Object Reference Model (SCORM) Runtime Environment (RTE), an example web-based LMS provided by the Joint ADL Co-Lab.

- The DARWARS Distributed Training Event Coordination Service (DTECS), a component responsible for matching roles in game-based training systems with material delivered by a traditional LMS. The DTECS ensures that both game-based training and traditional learning materials address the trainee's objectives. It coordinates configuration and delivery of the two modes of training, using the trainee's performance to reinforce and remediate lessons learned.

This framework can be employed with any game-based training system, LMS, DTECS, or registry. In this paper, after further discussing the motivation behind our framework, we provide further detail on the use case that informed our approach. Then we explain how our framework promotes reusability, including in our explanation details about how we employed web services and team training methodologies to allow the ADL Registry to disseminate integrated training material and to enable the interoperability of components. This paper closes with a summary of benefits to the warfighter and with recommendations for next steps.

MOTIVATION

We sought to develop a training framework that facilitates current lessons learned, and one that integrates newer, game-based approaches with traditional learning systems. Of these, the more important is perhaps the need to facilitate the dissemination of up-to-date lessons learned. In that regard, many lessons learned are best assimilated using the immersive, shared environment provided by game-based training systems, especially lessons learned that relate to group interaction or group performance in realistic environments. Further, stand-alone game-based training is greatly enhanced by the context provided by coordinated briefing materials, which enhance learning by providing a familiar and realistic environment. For example, DARWARS Ambush! instructors indicate that the inclusion of materials such as a "Mission Statement," the "Commander's Intent," the "Convoy Commander's OPLAN," as well as situational briefs, overviews, and maps, help to properly set the stage for game-based training by emulating how operational missions are briefed. In addition, simulation tools typically do not offer the remediation capabilities provided by many traditional LMS packages. Consequently, we feel that lessons learned are best shared with the community in integrated form – in a tight yet flexible coupling between traditional briefings or courseware and the game-based immersive environment.

Almost as important is our goal of fostering the growth of blended training instructional methodologies by providing a framework that employs accepted standards to integrate a wide range of game-based and traditional training. Integrating game-based content with material such as a SCORM course, or including briefing materials in a standardized form, such as the SCORM standard provides, can be a daunting task. Therefore, a framework and tools for both producing such integrated content and for making that content readily available to the broader warfighter community, make it easier to both produce new lessons learned and capitalize on those lessons learned through re-use.

USE CASE DETAILS

Figure 1 illustrates the essential components of the framework: showing the ADL-Registry and a local repository, the Learning Management System, the DTECS, and the Games or Simulations. The blue dashed lines indicate information retrieved from the Repository, while the black solid lines indicate the flow of control and information during a training session. Each of the above components may reside on a different computer and in different locations. Typically, each training institution will have a single resident Learning Management System and DTECS to serve their user community. Games and simulations will generally be installed on individual client computer hardware, for reasons explained below. The Registry and Repository are diagrammed as a single component, as that is how it is logically perceived by the users ; whereas these can be separate systems.

Prior to the training sessions system administrators establish a training environment on one or more computers, including server components for the DTECS and LMS, and clients or some combination of servers and clients for the games or simulations. To operate the DTECS and LMS, which typically provide browser-based interfaces, only a standard browser is required on client machines.

Generally the training system software installation itself precedes the search for integrated training content. While instructors may be interested in using the ADL-R to search for new training systems, such as DARWARS Ambush!, we think it is more likely that instructors already employing specific training systems will use the ADL-R to search for new content for those

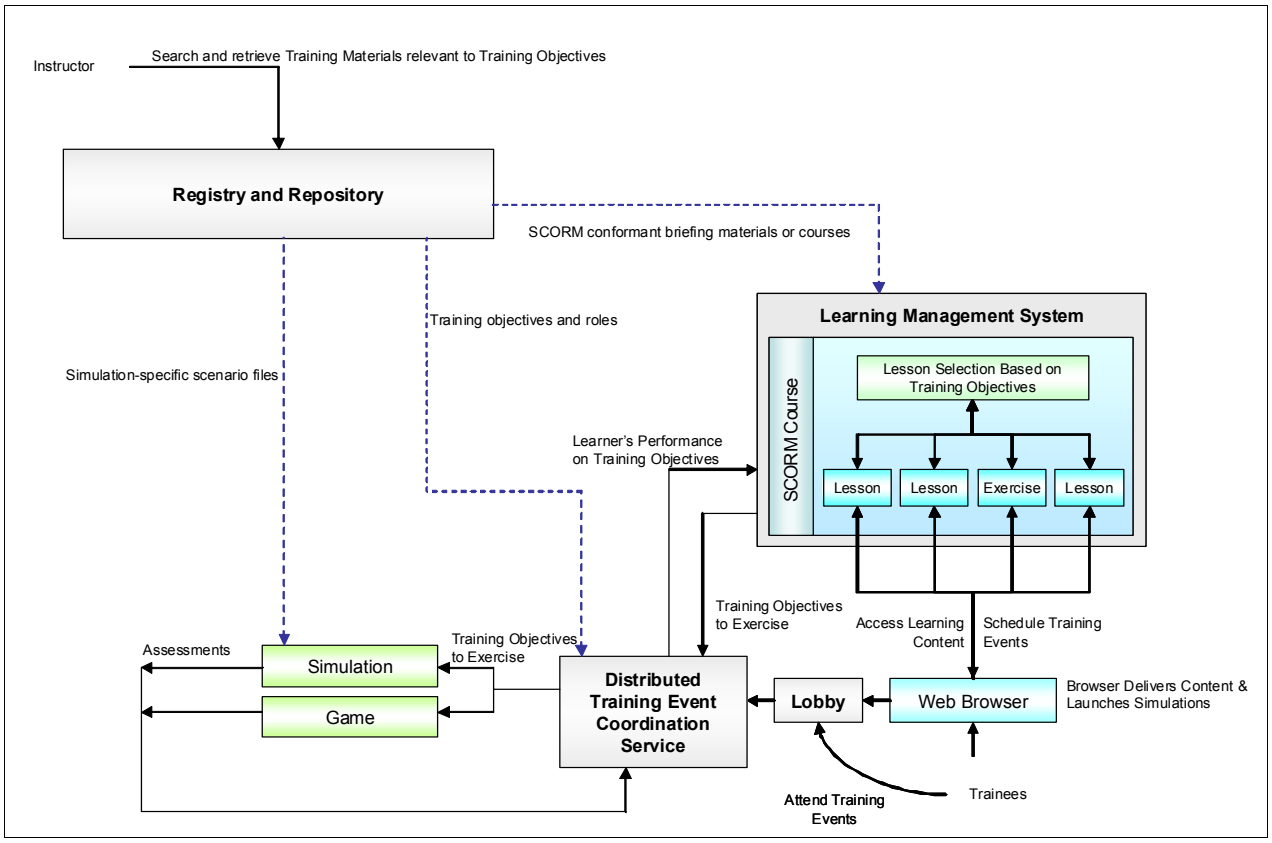

FIGURE 1: COMPONENTS OF THE INTEGRATED TRAINING FRAMEWORK

systems, rather than searching for new training systems themselves. Proposals to employ the ADL-R in support of new, game-based training have provoked much discussion about what information should be stored in the registry. With traditional content, i.e., web pages containing text and images, the actual content is stored in the repository. With simulation-based training systems we need to consider what is reasonable to store in the repository in order to promote reuse. Although some people have advocated storing the entire training system in a repository, we think this is the wrong level of granularity for several reasons:

- Game-based training systems can be very large.
- Game-based training often has complex system requirements; i.e., there can be various versions for different operating systems, varying requirements for specific graphics hardware and/or software, etc.
- The installation of training systems can require different system privileges (i.e., system administrator access) than are typically accorded to instructors.
- The installation of training systems generally requires local IT approvals.

For all of the above reasons, we advocate storing only mission or scenario-specific information in the repository, and we assume that all required software is already installed on local client or server machines. Once the software tools are installed, the instructor searches for training materials to address a specific set of training objectives; then an appropriate training package is retrieved from the repository and installed onto the server computer.

The package contains several parts within a single self-installing executable: a SCORM course or other collateral containing web-based training material, a DTECS training package containing data needed to match trainees to appropriate training roles by specified objectives, and the DARWARS Ambush! (or other simulation-based training system) scenario file. Because DARWARS Ambush! distributes scenario files from a designated server machine to the clients at runtime, the scenario files need only be installed on a single designated server computer. The installer distributes the information to the SCORM RTE, DTECS, and training system as illustrated in Figure 2.

The training package specifies how to distribute trainees to appropriate roles in the scenario. In the case

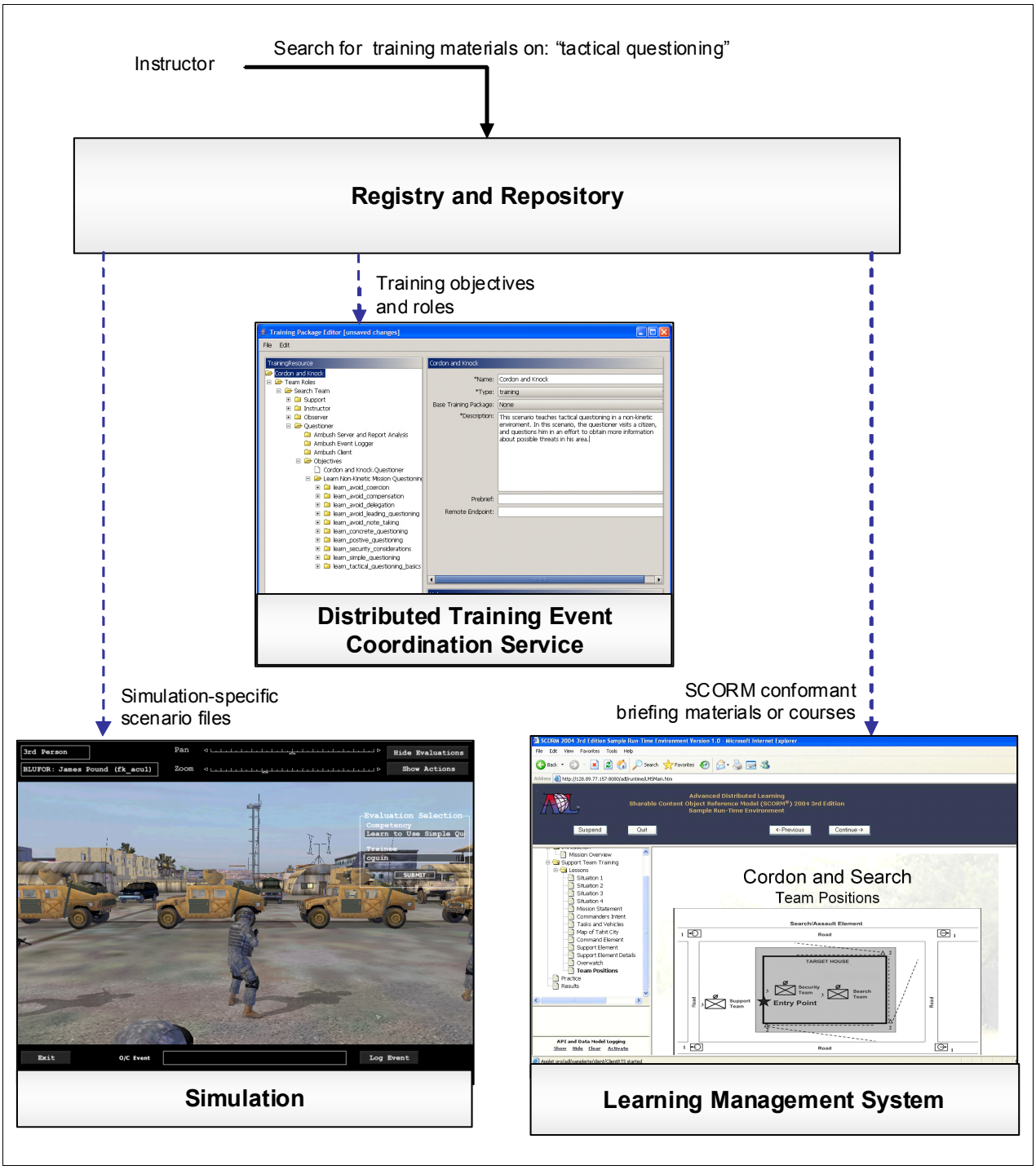

FIGURE 2: TRAINING MATERIALS SHARED THROUGH THE REPOSITORY

of a Cordon-and-Knock scenario, there is a role for an instructor, for one or more observers (to evaluate the trainees), and several trainee roles (interrogators, drivers, gunners, convoy commander, etc.). The training package also specifies what training objectives are associated with each role. When the LMS redirects each trainee to the DTECS, it submits relevant training objectives to the DTECS that it will then use to match the trainee to the best role in the training package.

The training package also contains the information needed to launch the training system with the appropriate scenario and roles. In our example Cordon-and-Knock scenario the training package provides information to DARWARS Ambush! to ensure that

each user begins the mission in the appropriate place with the appropriate role. Different training packages may be used to couple traditional training materials with game-based scenarios in a variety of ways.

Continuing our description of the use case, each trainee sits at a client machine and receives instructions as to which SCORM course or SCORM-conformant materials to review. A different SCORM course may be available for each role in the simulation. For example, while both a convoy-driver course and a gunner course might include the same general briefing information for the mission, the material covering specific roles would differ. The SCORM courses are also tied to different sets of training objectives, which are reported to the DTECS when the trainee enters the lobby to participate in game-based training. The DTECS then uses the training objectives, as well as the trainee's self-identified team membership (if relevant to the scenario), to match the trainee to an appropriate role in the convoy simulation. When all trainees are ready, the DTECS launches the simulation, which in return reports scores via web service calls to the DTECS server, where the instructor may review them later. The DTECS server also consolidates and returns the scores to the SCORM RTE, where the SCORM sequencer uses them to redirect the trainee to remedial material as appropriate.

Because the DTECS training package, SCORM courses, and game-based scenario files were all authored and bundled together with the intent to provide an integrated training experience, and because the DTECS uses a single-sign-on mechanism, each trainee receives a tightly integrated, coherent, and fluid training experience.

PROMOTING REUSABILITY

The key element in our approach to promoting reusability is the use of a searchable, shared repository of training content. The web-based ADL Registry provides this functionality. The Registry allows developers of training content to register their repositories in a unified, publicly-accessible registry, which is similar to a library card catalog system. Developers can then enumerate the individual content packages in their repositories by uploading XML-formatted metadata. The metadata must conform to a strict standard, called the ADL Learning Objective Metadata (LOM) Schema. Metadata files contain pre-defined data fields that include keywords, a general description, and a security level designator. Metadata also includes a list of additional user-defined data of interest such as information about the author, the date of authorship, and the organization to which the author belongs.

Our example metadata for the Cordon-and-Knock scenario includes a list of specific training objectives that the scenario offers ("tactical questioning," "use of negotiation skills," etc.), as well as a set of keywords that describe the mission objectives and scenario ("cordon-and-knock").

The DTECS provides the ability to create training packages defined using XML, and a SCORM Runtime Environment allows the importation of SCORM-conformant material. Individual game-based training systems will differ in their representation of simulation scenarios, but many, including DARWARS Ambush!, provide the capability to author and import scenario files. An instructor bundles these elements – the DTECS training package, the SCORM course, the game-based scenario files –to produce an integrated training package, which he then registers with the ADL Registry.

Although bundling these into a single compressed file would be enough to allow an instructor to manually import each element into the appropriate tool, we further facilitate reuse by modifying the tools as follows:

- SCORM RTE We added an importation facility to the SCORM RTE enabling clients to import a course without having to manually navigate through an administrator interface. This web-based facility allows an automated installer to import one or more SCORM courses as part of a single, intuitive, un-packaging process.

- DTECS Server We enhanced the DTECS server to expose a web service that enables clients to import one or more training packages automatically.

With such modifications, a single self-installing executable can import any number of SCORM courses, DTECS training packages, and game-based scenario files.

PROMOTING INTEROPERABILITY

Our framework was designed to support a wide range of integrated training systems. Towards that end, our approach uses web services as a modular, well-established, and extensible means of communication among components.

Currently, the ADL SCORM standard restricts content to browser-based material capable of interacting with the SCORM RTE via ECMAScript. By extending the standard to expose similar functionality via web service calls, non-browser-based applications (including stand-alone simulations and the DTECS) can communicate directly with the SCORM runtime during training.

In our approach, the DTECS communicates with the SCORM RTE via web service calls, mediating between the simulation software and the RTE. This requires exposure of several web service methods that correspond to the functionality currently provided by the ECMAScript interface.

We describe the web service calls below:

- Initialize() – Retrieves the relevant SCORM CMI data model and stores it in the session, allowing a non web-based program to function as a Sharable Content Object (SCO).

- GetValue(), SetValue() – Parallel the similar GetValue() and SetValue() methods of the ECMAScript interface, using a string representation of a SCORM value to get or set that value.

- Commit() – Commits gets and sets to the CMI data model in permanent storage.

- Terminate() – Releases the session and cleans up.

We are currently proposing this web service as an IEEE standard under the sponsorship of the IEEE Learning Technology Standards Committee (LTSC). The adoption of this standard will promote interoperability among the components of our framework, as well as generally support integration of any non-web-based content into the SCORM Runtime Environment.

TEAM-BASED TRAINING

Our framework allows team-based simulations such as DARWARS Ambush! to interact with individual centered, web-based training tools. The opportunity to interact as a member of a team with other trainees in a realistic shared environment is one of the primary advantages provided by game-based training. In our framework negotiating between LMS packages and game-based systems is the responsibility of the DTECS and the DTECS training package.

Team-based training adds a layer of complexity to the process of matching individuals to training roles. In a convoy training system, for example, it may not be enough simply to match individuals to training roles that match their training objectives. The instructor may also desire that trainees train with the specific teammates with whom they will be working in the field. A trainee who wishes to train as a driver for a particular team, therefore, needs to be placed not in an arbitrary driver role, but in the driver role for the vehicle carrying the trainee's teammates.

We have enhanced the specification of the DTECS training package to include team roles as well as individual roles. This allows our framework to support a classroom divided into multiple teams, assigning team members to their proper team roles in the game-based training system, while still ensuring that each individual trains to his or her appropriate objectives. A DTECS team role is defined as a collection of one or more individual roles. At simulation time, a team role is assigned a team in the same way that an individual role is assigned an individual. A team can be predefined by the instructor using the DTECS web-based instructor interface, or created ad hoc at simulation time as trainees log into the DTECS waiting area ("lobby") to begin training. If an individual is already a member of a pre-defined team, the DTECS prompts the trainee to specify the appropriate team name as the trainee enters the lobby, thus allowing trainees to participate in different teams during different training sessions.

Support for team training resides entirely within the DTECS, while the SCORM RTE focuses on delivering content that meets the training objectives of an individual. Restricting team membership issues to the DTECS simplifies both the content creation and content delivery process.

BENEFITS TO THE WARFIGHTER

Game-based training allows warfighters to practice individual and collective skills in a safe, supportive environment. Furthering the reusability and interoperability of integrated training allows lessons learned in the field to be disseminated to other warfighters more effectively and more rapidly. And in a game-based environment lessons learned can be experienced as well as studied.

Simulations by themselves do not offer training. The context provided by the associated briefing materials is critical for allowing trainees to assimilate the information in a useful and directed way. The

integrated training package provides both the shared experience and the larger context required to communicate lessons learned.

Facilitating the ease of reuse further allows a community of interest to form around creating and discussing game-based scenarios and their accompanying collateral. In such a community, content authors find a wider audience for their training material, and this can motivate the development and refinement of additional scenarios and training packages. The growing community then provides a place to not only publish and find new lessons learned, but to adapt existing training material to new circumstances.

COST SAVINGS

Our framework is designed to save time, effort, and money by increasing the efficiency of both the training itself and the dissemination and reuse of training materials. Training time is made more efficient by guaranteeing that all trainees are training to a role that meets their specific objectives. That is, fewer trainees are merely playing a supporting role to other trainees, or participating in a simulation in an undirected role. This means less valuable training time is required to achieve the same results. Interspersing remedial material with game-based training makes training more immediate and more relevant, as the trainee is assimilating information in multiple modes. Making training more immediately effective decreases the need for extended periods of remediation or reinforcement, decreasing the cost.

Improving reusability also creates savings by reducing training costs. Training cost can be measured as the cost to produce and distribute training materials vs. the number of hours of training provided. For example, initial development of Ambush! took approximately six months. The initial release had 25 scenarios grouped into five "missions" which shared a storyline and some assets (scripts, models, etc.). Rough man-power estimates produce a cost per scenario of about 300 person-hours. As an example of the kind of reductions one can expect when the development pipeline is in place, a recent set of enhancements to Ambush! produced 10 new scenarios, new models for three new vehicles and other objects and additional scripting. These were produced for slightly less than 150 person-hours per scenario. Finally, there is another portion of the development-hour/training-hour scale that is exemplified by Ambush! As mentioned earlier, Ambush! showcases the value of authoring tools that allow non-developers to modify and create additional content. One sees novices developing scenarios within 2-4 hours. Ratios of less than 10-to-1 (10 hours of development for 1 hour of training) are typical of this "end-user authoring". In our experience, similar reductions in development effort apply to the production of traditional briefing materials and other collateral associated with an integrated training experience.

While experience with authoring tools, and the ability to build on previous scenarios and training materials drives down the initial cost of producing training, a potentially much larger cost reduction is attainable through re-use. We are seeing these cost benefits in the Ambush! community where over 30,000 warfighters have been trained on the initial 25 scenarios developed with the training system. Once training materials have been produced, the additional cost of registering the materials using the ADL-R is minimal, often requiring less than an hour to package and distribute new content. Hence, reuse dramatically decreases the development cost per hour of training.

NEXT STEPS

The ultimate goal of this work is to provide improved military training. There are several steps that should be taken in order to reach this goal:

- Adoption of a web service API into the SCORM standard. This will facilitate the integration of non-browser-based simulation training into SCORM-compliant learning management systems.

- Integration with LMS systems currently used by distance learning centers and schoolhouses. The SCORM Sample RTE used in our demonstration is only a sample LMS. Promoting widespread usage will require integrating with tools already widely used.

- Promotion of the sharing and reuse of training materials through the ADL Registry. Such promotion might be accomplished by inviting content authors from already active communities (such as the DARWARS Ambush! community) to submit their simulations to the Registry for distribution. The effectiveness of the Registry could be measured by usage statistics.

- Support a pilot operation with a specific user community. The effort documented here produced a demonstration of the framework. Further work on this approach would benefit from information gained by testing within an established training community,

such as the existing widespread DARWARS Ambush! community.

- Formal testing of outcomes to demonstrate the effectiveness of integrated simulation-based and traditional training. Such testing could compare results on competency tests or in live exercises after either participating in either an integrated training experience or in an alternative (e.g., traditional classroom training interspersed informally with simulation-based training).

In order to encourage the adoption of this approach, we are working with several technical standards' groups including the SCORM Technical Working Group and the Learning-Education-Training Systems Interoperability (LETSI) Technical Working Group. In addition, we have posted the Web Services Definition Language (WSDL) and accompanying technical documentation for review on the ADL community web site , and we maintain a "sandbox" for experimentation on externally available servers at BBN.

ACKNOWLEDGEMENTS

BBN acknowledges the support of the Joint ADL Co-Lab in sponsoring this work under contract number N61339-06-C-0084.

REFERENCES

The ADL Registry: http://www.jointadlcolab.org/support/adlr/index.aspx

ADL Co-Lab, (2004). *SCORM 2004 3rd Edition Sharable Content Object Reference Model*. Available from: http://www.adlnet.gov/scorm/index.aspx

Travers, V., Tomlinson, R., Guin, C., Roberts, R. B., Marks, J. (2007). A Web Service Architecture for Integrating Didactic and Experiential Learning. *I/ITSEC 2007*.