Training &
Support
Technology

Issue Date: 29 September 2006
Revision1 Issue Date: October 12, 2006

Performance-Based Advancement Using the SCORM 2004 Data Model

Project Final Report

CONTRACT NUMBER N61339-05-C-0114
DATA ITEM A005

The Boeing Company
Training and Support Technology

PREPARED BY: Dr. Elizabeth Biddle, Dr. Bruce Perrin, & Mr. Brandt Dargue

APPROVED BY: Dr. Elizabeth Biddle
Orlando R&D Manager for Joint Live, Virtual Constructive Technologies

BOEING®

Performance-Based Advancement Using the SCORM 2004 Data Model: Project Final Report

Prepared by:

Dr. Elizabeth Biddle, Dr. Bruce Perrin, & Mr. Brandt Dargue
The Boeing Company

September 29, 2006

Sponsored by:
Joint Advanced Distributed Learning Co-Laboratory
N61339-05-C-0114

Abstract

The Advanced Distributed Learning (ADL) vision encompasses the use of scenario-based simulations to provide a rich environment for training complex tasks. At the same time, it introduces a complex assessment environment, which creates challenges in the accurate and efficient diagnosis of student needs as frequently student behaviors can be interpreted in several ways. Diagnosing student learning needs, consequently, becomes problematic. Unfortunately, there are currently no best practice guidelines for extracting and making use of performance data from a simulation-based training environment. However, methods that address these challenges are required for the successful integration of simulation-based training into the ADL Initiative.

The research conducted under the 2005 Performance-Based Advancement project and described in this report implemented a standards-based method for assessing and diagnosing individual student performance in a simulation-based training session. This method leverages the High Level Architecture (HLA) and Shareable Content Object Reference Model (SCORM) standards. Specifically, this method involves the development of scenario-based metrics related to lower-level learning objectives that underlie the complex skills used in scenario-based training, while using information on trends to isolate individual learning needs.

Specifically, SCORM 2004 specifications enable a single Shareable Content Object (SCO) to be linked to (i.e., to set a value of or the status of) multiple learning objectives. Although the potential impact of this capability for assessment has not been widely recognized to date, it provides a means to interpret relatively complex responses in scenario-based training in terms of all of the learning objectives that may be implicated by a given action. The methods developed under this project support changing the measures that reflect a student's mastery of the underlying learning objectives as a result of study, practice, and forgetting. Further, hypothesis testing methods were employed to resolve ambiguous diagnoses of learner needs.

These methods implemented with the development of an ADL prototype in the context of Marine Air Ground Task Force (MAGTF) command and control training. The MAGTF XXI Tactical Decision-Making Simulation (TDS) was employed as the simulation-based training environment.

Table of Contents

Abstract iii
Table of Contents iv
Table of Figures v
1. Brief Description of Effort 1
1.1. Background 1
1.2. Major Tasks 1
1.3. Prototype Description 2
2. Research Strategy 6
2.1. Identify Research Scenario 6
2.2. Develop Training Content 8
2.3. Define Common Learning Objectives 9
2.4. Integrate Simulation and SCORM 2004 Content 10
3. Major Obstacles 12
3.1. Establishing Behavior – Learning Objective Associations 12
3.2. Accumulating Performance History under SCORM 13
4. Major Accomplishments 15
4.1. SCORM-Based Student Modeling 15
4.2. Trend-Based Performance Assessment 15
4.3. Method for Automatically Identifying Root Cause of Student Error 18
4.4. Developed an Architecture for Simulation-Based Training Management Capability 23
4.5. Implemented Several Hours of Level3/Level4 Courseware 26
5. Shortfalls 29
6. Lessons Learned 31
6.1. Limit Scope of Competing Hypotheses 31
6.2. Selection of Target Population 31
7. Next Steps 32
7.1. Training Effectiveness Evaluation 32
7.2. Extend Prototype 32
7.3. Transition Prototype to the Military Training Community 32
8. Feedback on Process from Start to Finish 34
8.1. Prototype Deliverable Expectation 34
8.2. Final Report Template 34
8.3. Change of Alternate Contracting Officer Representative (ACOR) 34
9. Published papers 35
Acronyms 36
References 37
Appendix A: Enabling Learning Objectives 38
Appendix B: Metrics 40

Table of Figures

Figure 1. Performance-Based Advancement Prototype Contribution to the JADL 2012 IPA 3
Figure 2. MAGTF XXI Interface for In-Stride Breach Mission 4
Figure 3. Prototype ICW 5
Figure 4. MAGTF XXI In-Stride Breach Scenario 7
Figure 5. Performance-Based Advancement Prototype Content Structure Diagram 9
Figure 6. 2003 HLA-SCORM Integration Architecture 11
Figure 7. Enabling Learning Objective (ELO) Association Frequencies 12
Figure 8. Effectiveness of Adaptive Learning Compared to Self-controlled Learning 15
Figure 9. ECMAscript Example 16
Figure 10. Rule Tracking Table 19
Figure 11. Rule Tracking Table after Rule Failure Analysis has been Performed 20
Figure 12. Relevance and Independence Data for Learning Objectives 21
Figure 13. XML Coding Example 22
Figure 14. ECMA Script to Skip Full Mission Scenarios 23
Figure 15. Performance-Based Advancement Prototype Architecture 24
Figure 16. Position Scoring Areas 25
Figure 17. ICW Example – MAGTF XXI Symbology 27
Figure 18. Example of Tactics ICW 28

1. BRIEF DESCRIPTION OF EFFORT

1.1. Background

Incorporating today's significant advancements in computer simulation, visualization, networking, and learning technologies, the Advanced Distributed Learning (ADL) Initiative is enabling the delivery of robust, scenario-based simulations which provide a rich environment for training. Through these simulations, new knowledge can be provided; skills and abilities can be developed, then honed, all in a safe, controlled setting that approximates actual work or combat situations. Within this robust training environment also comes a richer, albeit more complex and confusing assessment environment. The Performance-Based Advancement prototype demonstrates a method and implements an architecture for student assessment and learning-needs hypothesis-testing for use in a simulation-based assessment environment.

Compared to today's traditional paper-and-pencil tests, assessments in online, scenario-based simulations can elicit a wide range of behaviors. Often, these behaviors can have any of several interpretations, resulting in differing diagnoses of learning needs. A simple example, provided below, will illustrate this point.

Suppose that during a scenario-based training exercise, a student, who has received instruction on how to use simple hand tools, tries to use a hammer and wood screws to connect two boards together. Such an action might indicate that the student misunderstands the proper use of screws and a screwdriver. But does the action also imply that the student does not understand the use of a hammer and nails, or only that he/she has over-generalized the function of a hammer to connect two boards? In such a case, studying the use of a hammer is less "to the point" than studying the use of a screwdriver. Or perhaps the error does not reflect misunderstandings of hand tools at all, but only that the student mistakenly thought the screws were nails, i.e., a random, perceptual error. The point of this example is really quite simple – for any given action in a scenario-based training exercise, there are numerous possible learning-need diagnoses.

The Performance-Based Assessment methodologies that are described in this report contribute to the realization of ADL's vision: "to provide access to the highest-quality learning and performance aiding that can be tailored to individual needs and delivered cost-effectively, anytime and anywhere" (http://www.adlnet.org/aboutadl/index.cfm). Specifically, these methodologies advance the current methods for diagnosing student performance thereby improving the ability of ADL courseware to be accurately tailored to an individual's unique needs.

1.2. Major Tasks

The objectives of the Performance-Based Advancement Using the Shareable Content Object Reference Model ® (SCORM) 2004 Data Model project were to design and develop scenario-based assessment and hypothesis-testing methods and demonstrate the implementation of these methods using SCORM 2004. Four primary tasks were performed to meet these objectives.

1.2.1. Task 1: Provide Assessment Methods for Scenario-Based Training Exercises

This task involved the development of methods that changing the estimate of a student's mastery of the underlying learning objectives as a result of learning and forgetting. These methods were based on standard measurement practice and metrics (e.g., moving average). This task is described in more detail in Section 4.2 Trend Based Performance Assessment.

1.2.2. Task 2: Provide Methods to Efficiently Resolve Alternative Diagnoses of Learning Needs

This task involved the development of methods to enable efficient identification of learning needs, based on actively testing competing hypotheses. This task is described in more detail in Section 4.3 Method for Automatically Identifying Root Cause of Student Error.

1.2.3. Document Limitations in SCORM 2004

This task involved assessing the ADL SCORM 2004 for its use in implementing the scenario-based assessment and hypothesis-testing methods. This task is described in more detail in Section 3.2 Accumulating Performance History under SCORM.

1.2.4. Implement Prototype

This task involved the development of a prototype to demonstrate the scenario-based assessment and hypothesis-testing methods. The selection of the application area for this prototype was made in cooperation with Mr. Michael Woodman, Director Tactical Decision-Making Simulation, who at the time of the project start represented the United States Marine Corps' (USMC's) Program Management Training Systems (PMTRASYS) as a transition sponsor. This task is described in more detail in Section 2. Research Strategy.

1.3. Prototype Description

Performance-Based Advancement, which is a standards-based method for automating the assessment and sequencing of individual student learning in a simulation-based training environment, was demonstrated through the implementation of a prototype. The Performance-Based Advancement method was developed to improve the accuracy of learner need diagnosis in the complex environment that is scenario-based training and is the integration of the scenario-based assessment and hypothesis testing methods described above. The Performance-Based Advancement prototype provides the basis for a performance assessment and sequencing architecture for simulation-based training by integrating the "Learning Management System" and "Simulation" elements of the Joint Advanced Distributed Learning Co-Laboratory (JADL) 2012 Integrated Prototype Architecture (IPA) – Baseline Model (Figure 1). Essentially, the Performance-Based Advancement prototype implemented a SCORM course, as depicted in the JADL 2012 Integrated Prototype Architecture (IPA), within a simulation-based training exercise.

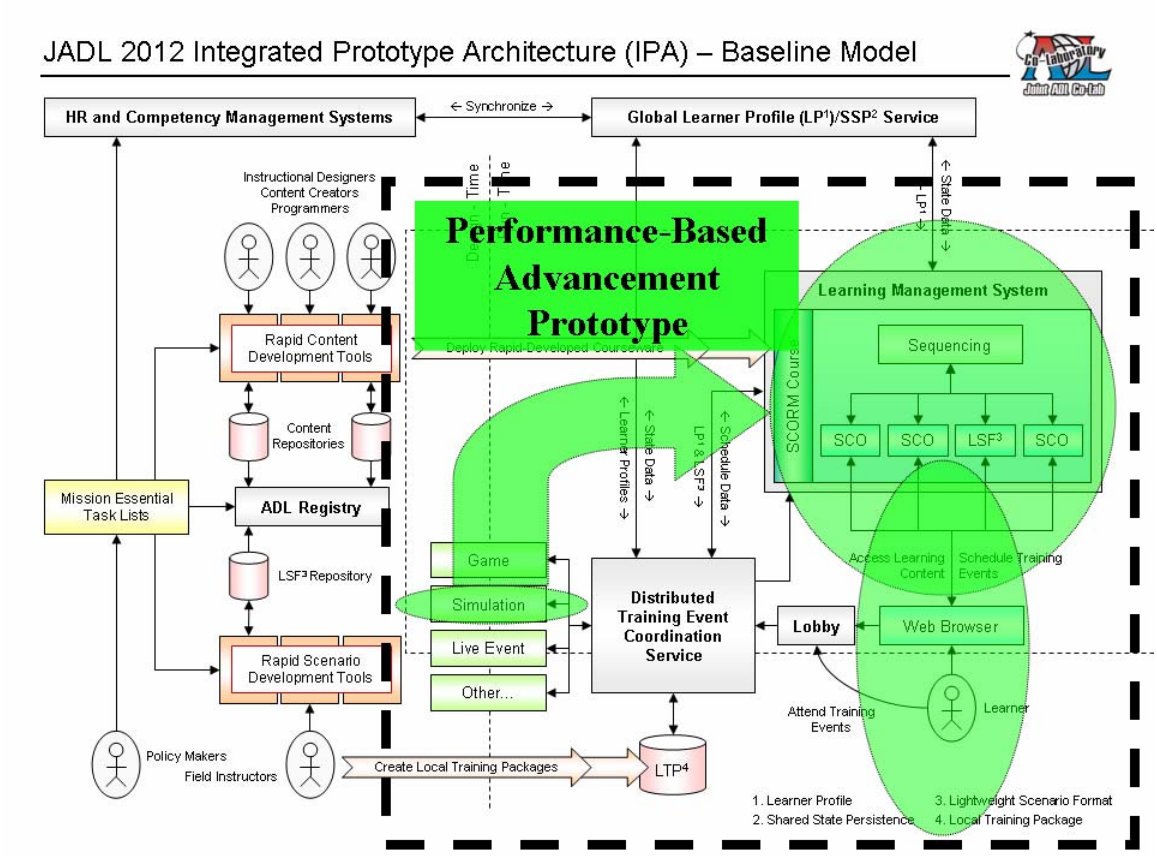

Figure 1. Performance-Based Advancement Prototype Contribution to the JADL 2012 IPA

The prototype uses the Marine Air Ground Task Force XXI (MAGTF XXI) HLA simulation for its skill practice environment (Figure 2), while providing traditional content presentation, which was implemented with Microsoft PowerPoint, for declarative knowledge training. MAGTF XXI is a real-time, tactical simulation developed by MÄK Technologies, Inc. for the United States Marine Corps (USMC) to facilitate expeditionary warfare training under the USMC Program Manager Training Systems (PM TRASYS) Tactical Decision-making Simulation program. The MAGTF XXI simulation was selected for the implementation of this prototype because it provides a complex simulation environment in the context of a realistic military task. Additionally, interface components for collecting and exporting performance data via the HLA from MAGTF XXI to a learning management system (LMS) via SCORM were already under development.

Several different roles for the student and several different training tasks were considered before we selected the final alternatives for this study. We selected the role of a commander of a mechanized infantry-tank team. The task we identified for the prototype was an in-stride breach of a minefield operation. The scenario selection is discussed in more detail in Section 2.1, Identify Research Scenario.

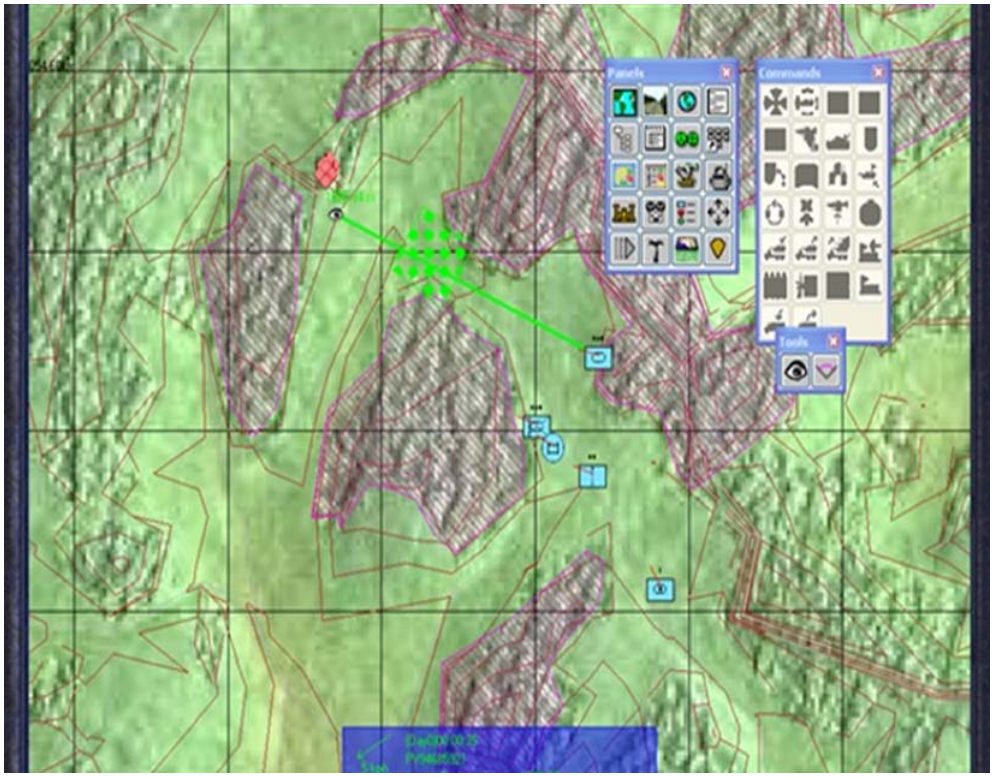

Figure 2. MAGTF XXI Interface for In-Stride Breach Mission

A session with the prototype begins with the student completing approximately 45 minutes of interactive courseware (ICW, Figure 3). This courseware represented 25 learning objectives that encompassed the MAGTF XXI interface and displays and the tactics for performing an in-stride breach of a minefield. The MAGTF XXI simulation was then used to allow the student to apply this declarative knowledge and develop procedural skill in executing and coordinating the actions of a combined mechanized infantry and tank team. Twenty-eight behavioral rules were defined that reflected correct (or incorrect) actions and related those behaviors to the relevant interface, display, and tactical learning objectives. During scenario execution, moving averages were calculated for each of the learning objectives, based on the rules and the behavior exhibited by the student. Following execution of the scenario, the set of learning objectives with relatively low scores could be considered a set of competing hypotheses about the source of a given student's learning needs. The system then selected a subsequent scenario that had behavioral rules that implicated some, but not all of these low-score objectives. In this way, student performance on the subsequent scenario would help to isolate the source of the performance difficulties from among the competing hypotheses. The following section 2. Research Strategy provides more detail on the development of the prototype.

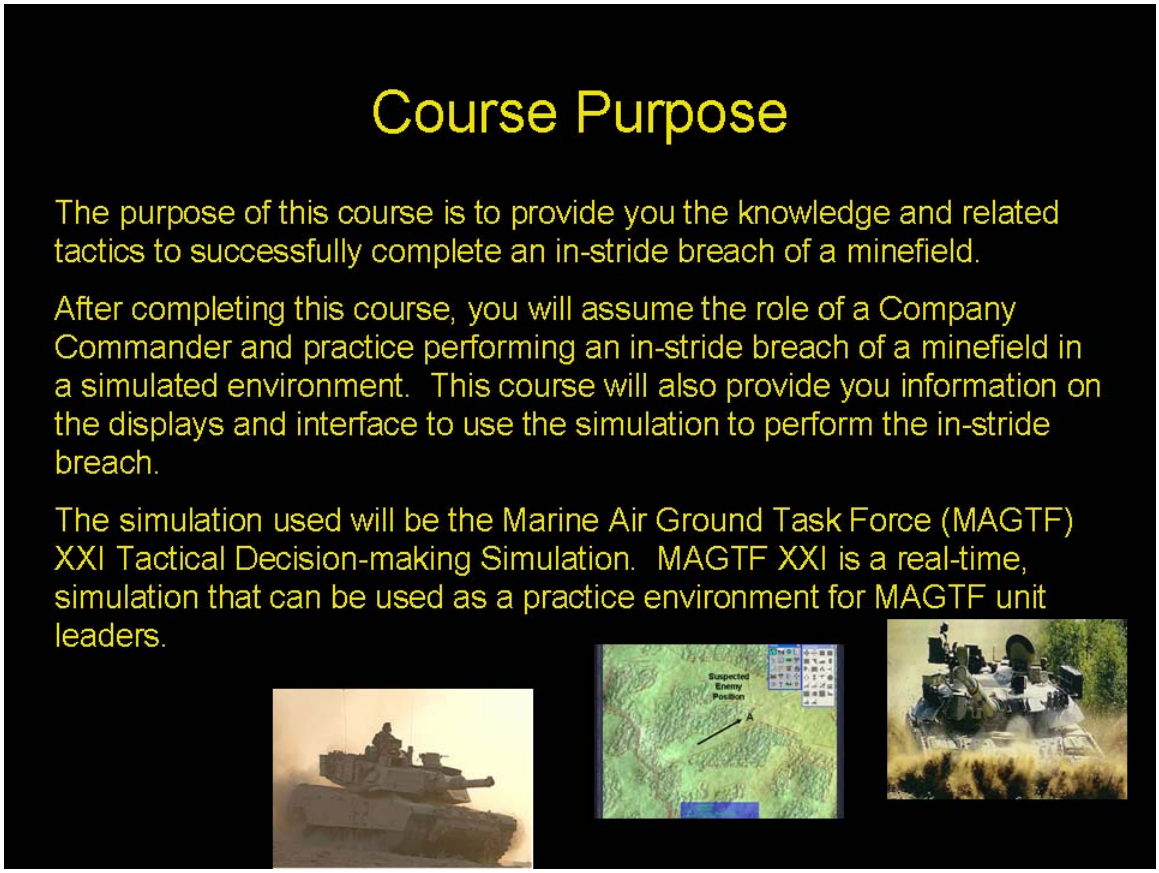

Figure 3. Prototype ICW

2. RESEARCH STRATEGY

The research strategy that we pursued for this project was to establish a training test-bed, in which our scenario-based assessment and hypothesis-testing methods could be implemented and demonstrated. Building this test-bed included the following activities:

- Identifying a research scenario
- Developing training content for this scenario and the simulation controls and displays
- Defining common learning objectives for the content and the simulation exercises
- Integrating the simulation with SCORM conformant content

These activities are described in the sections that follow. With the test-bed in place, we could pursue the design and development of our scenario-based assessment and hypothesis-testing methods. These activities are discussed in Section 4, Major Accomplishments.

2.1. Identify Research Scenario

Working in conjunction with the Joint Advanced Distributed Learning Co-Laboratory (JADL), we identified the MAGTF XXI tactical decision-making simulation, developed by MÄK Technologies, Inc. for use in this project prior to contract award. A first action after award, however, was to identify a scenario to be used for this project.

Several different roles for the student and several different training tasks were considered before we selected the final alternatives for this study. We selected the role of a commander of a mechanized infantry-tank team. The task we identified was to conduct an “in-stride” breach of a minefield (Figure 4).

This task and role selection had several advantages for the purpose of this project. First, the scenario objective and boundaries are relatively clear. The option of bypassing the minefield was excluded from consideration by selecting an area bordered by relatively inaccessible terrain and introducing time constraints. Second, the task duration is relatively brief, permitting iterative practice sessions that would be necessary to demonstrate learning. Third, although the procedure is relatively well defined, it is challenging to execute. It requires both initial planning, as well as continuing, time-critical problem solving as the scenario unfolds. Given the complexity and dynamic nature of the scenario, even experts can fail.

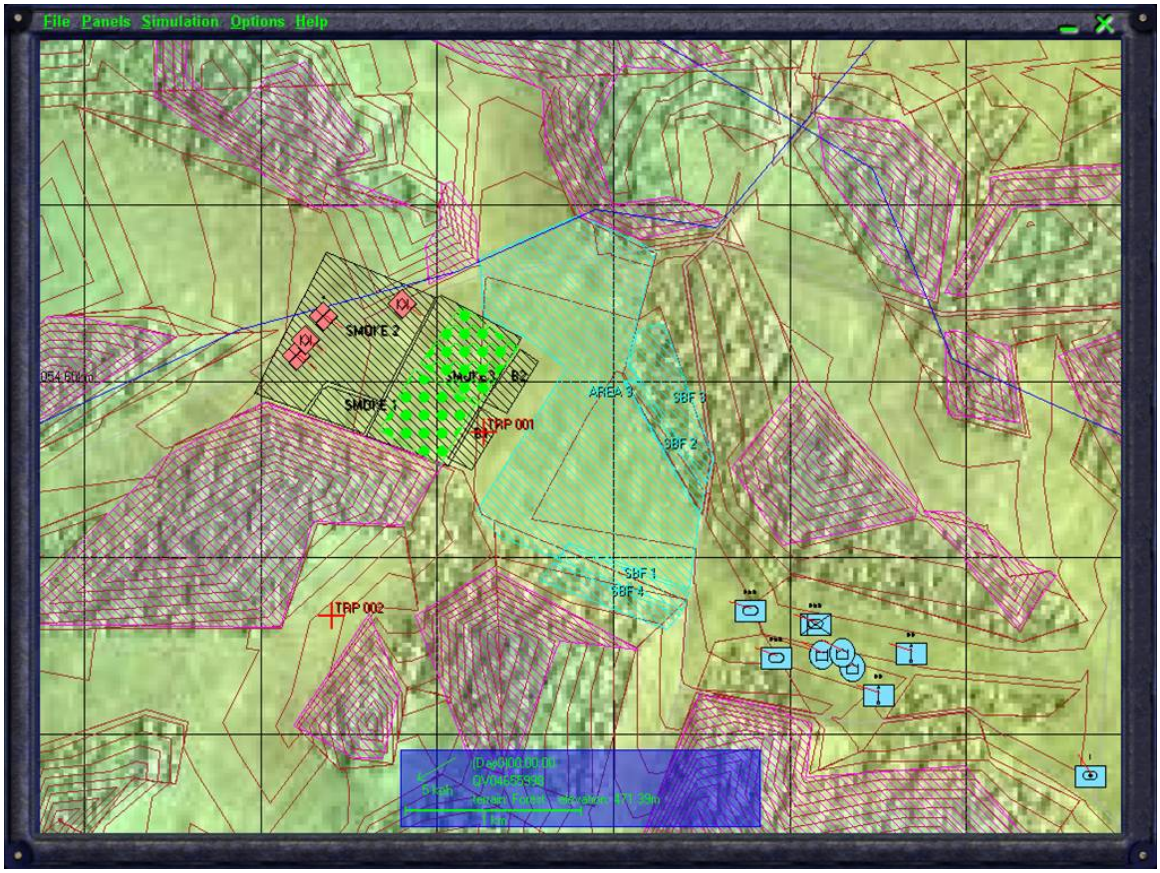

Figure 4. MAGTF XXI In-Stride Breach Scenario

For the student population, we selected novices, both with respect to their experience with MAGTF XXI and as a team commander. This selection created a situation in which most student behaviors during the scenario would implicate at least two different learning hypotheses. Specifically, most responses would have implications for our estimates of mastery of the learning objectives for both the MAGTF XXI interface and for in-stride breach tactical actions.

Another benefit to using novices as the student population is that early in training, errors of omission result from inadequate cognitive capacity of the trainees to initiate actions that they know they should execute and that they know how to execute. That is, based on a considerable body of research on the development of expertise (e.g., Anderson, 1993; Newell, 1990), initial performance is likely to be slow and error prone, simply because the trainees are using a declarative form of the problem-solving skills. So, students who know how to launch smoke rounds and understand that they should, may not, simply because they are preoccupied with low level actions such as establishing routes of travel. As training continues, these lower level skills should become more proceduralized, permitting the students to focus attention on more complex decisions.

With the addition of workload or capacity as a root cause of performance issues, many errors will have at least three competing explanations – a lack of mastery of the MAGTF

XXI user interface, a lack of understanding of the proper tactical response in a given situation, or a lack of capacity to perform the action. The content was structured to reflect these three root causes, with aggregations for:

- training MAGTF XXI menus, commands, and displays;
- training strategies and tactics for performing a hasty breach; and
- a final one involving the execution of MAGTF XXI scenarios to practice and proceduralize the skills involved.

The content structure is described in more detail in Section 2.2 Develop Training Content.

In order to demonstrate the scenario selection capability, a pool of 12 MAGTF XXI scenarios was developed. The scenarios differed based on:

- Force Structure – 3 levels (high, medium, and low threat),
- Position – 2 levels (northeast, northwest), and
- Wind – 2 levels (from east, from west).

Table 1. MAGTF XXI Scenario Pool

	Position – northeast Wind - east	Position – northeast Wind - west	Position – northwest Wind - east	Position – northwest Wind - west
High Threat	Full	PT1	PT2	Full
Medium Threat	PT1	Full	Full	PT1
Low Threat	Full	PT1	Full	PT2

Additionally, some of the scenarios were the entire in–stride breach mission (Full), whereas, some of the areas were implemented as part-task, to address only the latter elements of the in-stride breach (PT1, PT2). Table 1 provides the details of the scenario pool.

2.2. Develop Training Content

The situation in which students are novices both with respect to in-stride breach tactics and the MAGTF XXI controls and displays is conducive to hypothesis testing research, because, in most cases, student errors will implicate at least two competing hypotheses. That is, an error may indicate that the student had not mastered the MAGTF XXI interface, or that the student had not mastered the in-stride breach tactics, or both.

It also implies, however, that training content covering all of this required knowledge must be developed in order to provide a basis for the demonstration of this technology. In response, we developed training content covering 25 learning objectives spanning in-stride breach tactics and the MAGTF XXI commands and displays (Figure 5). This content was organized as 25 Shareable Content Objects (SCOs) and involved 73 screens of text, graphics, and animations, the latter generally used to illustrate MAGTF XXI

commands. In a complete implementation of adaptive learning technology, remediation to one or more of the SCOs would be prescribed by our scenario-based assessment and hypothesis-testing methods.

Implementation of remediation to training content, however, was beyond the scope of this project.

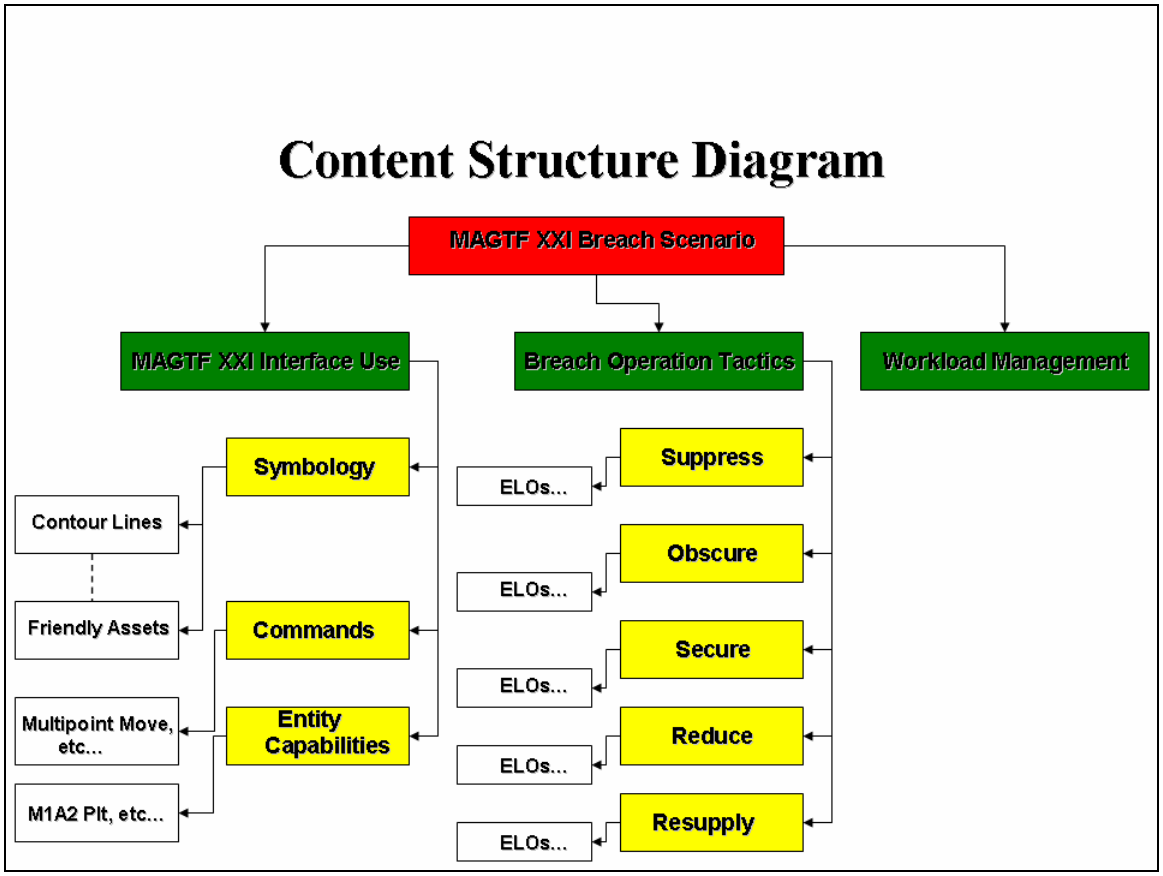

Figure 5. Performance-Based Advancement Prototype Content Structure Diagram

2.3. Define Common Learning Objectives

As a matter of practice, most SCOs are linked to a single, "primary" learning objective. However, SCORM also makes provisions for a SCO to be linked to (i.e., to set a value of or the status of) multiple objectives. The potential impact of this capability for assessment has not been widely recognized to date. It provides a means to interpret relatively complex responses in scenario-based training in terms of all of the learning objectives that may be implicated by performance. To do so, however, a common set of learning objectives, representing all of the content that may be used in performing the simulation exercises, must be defined. Then, student actions in the simulation SCO can be used to implicate the learning objectives for the related content SCOs.

As part of the project, we defined and implemented a set of learning objectives that spanned the in-stride tactics and the MAGTF XXI commands and controls (Figure 5).

Generally, the development of a common set of objectives based on content studied by the student and applied during simulation practice will be straightforward. It is, however, an issue that is rarely addressed directly in the development of the content or the simulation-based training activities that uses the information it provides. Although the research base on this technology is limited, a distinction that has proven useful is that between initial declarative knowledge and the proceduralization of that knowledge in later problem-solving, judgment, or decision-making practice.

Computer-delivered content will typically be used to provide the declarative information, including the facts, rules, procedures, etc. necessary for performance. Standard training texts (e.g., Goldstein & Ford, 2002) recognize the demonstrated utility of computer-delivered content for imparting declarative knowledge – it is, in short, one of the preferred media for this type of information. But to take the declarative forms of these skills, which can be provided in content, and proceduralize and automate them takes practice. That practice may come in the form of simulation.

Consequently, there is generally an inherent relationship between content-provided learning objectives and those pursued in simulation-based training. For scenario-based assessment and hypothesis testing to be successful, that inherent relationship must be recognized and used to interpret behavior. For example, a learner who performs a task promptly and accurately has demonstrated some procedural skill. Under our formulation of adaptive learning, the estimated mastery of the learning objective related to this skill should be incremented. By the same reasoning, the learning objectives related to the declarative knowledge applied during that task execution should also be incremented. While declarative knowledge understanding is often demonstrated by embedded tests within the content, application of this knowledge during simulation indicates that the learner also understands the use of this information in task performance. This application of declarative knowledge is important additional data on the mastery of the related learning objectives.

The inferences from performance are similar, but more complex, especially when a student makes an error. Take, for example, the situation in which controls that define both the skills necessary to implement an action and the situation under which this action should be performed have been defined. Then, suppose that the student does not execute this control when the given situation arises in the course of simulation-based training. Presumably, this error could be the result of forgetting about or misunderstanding of the control that should be used; similarly, it could be the result of forgetting about or misunderstanding of the situation that calls for the use of this control. Development of a common set of learning objectives allows us to entertain hypotheses related to either root cause, establishing the situation necessary for the testing of competing hypotheses.

2.4. Integrate Simulation and SCORM 2004 Content

To assess performance and diagnose learning needs, SCORM 2004 content must have access to information from the simulation. Otherwise, it will not be able to know how the student has performed, and consequently, how to update the learning objective scores and select follow-on activities. While the simulation could have been of many different

types, the preponderance of our simulations conform to the Distributed Interactive Simulation (DIS) and/or HLA standards. Creating a bridge between a SCORM 2004-conformant LMS and HLA conformant simulations would provide ready base of reusable, standards-based simulations for training.

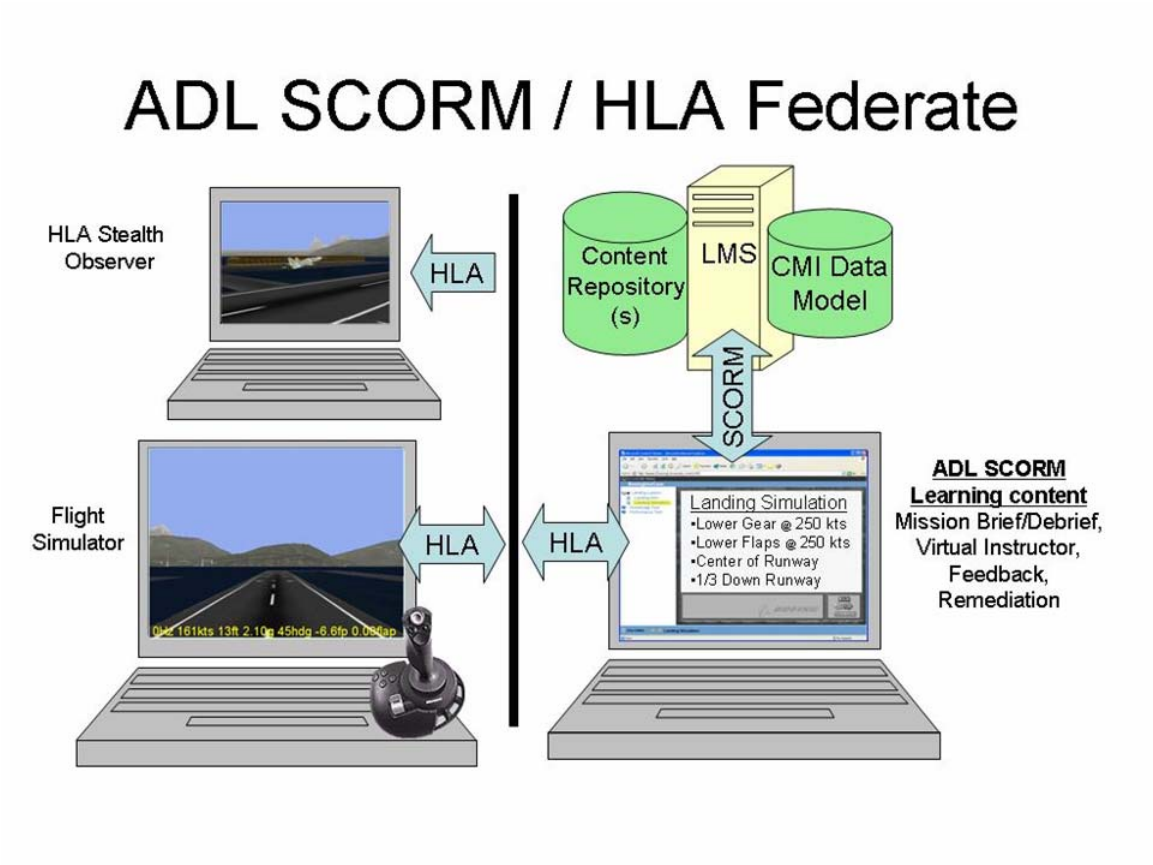

Figure 6. 2003 HLA-SCORM Integration Architecture

We demonstrated the capability to integrate SCORM and HLA simulations in a prototype at the Interservice/Industry Training, Simulation and Education Conference (I/ITSEC) 2003 (Figure 6). In this integration, the simulation SCO sends an HLA data packet to the simulation, which initializes it for the selected exercise. The student performs the task in the simulation, with the simulation sending student performance data back to the SCO using HLA. Assessment of these data is performed by the SCO. In turn, these data are used to update the variables related to the learning objectives implicated by the student's performance. This integration provides for the continuing communications from student performance in the simulation to the student model maintained by SCORM.

3. MAJOR OBSTACLES

3.1. Establishing Behavior – Learning Objective Associations

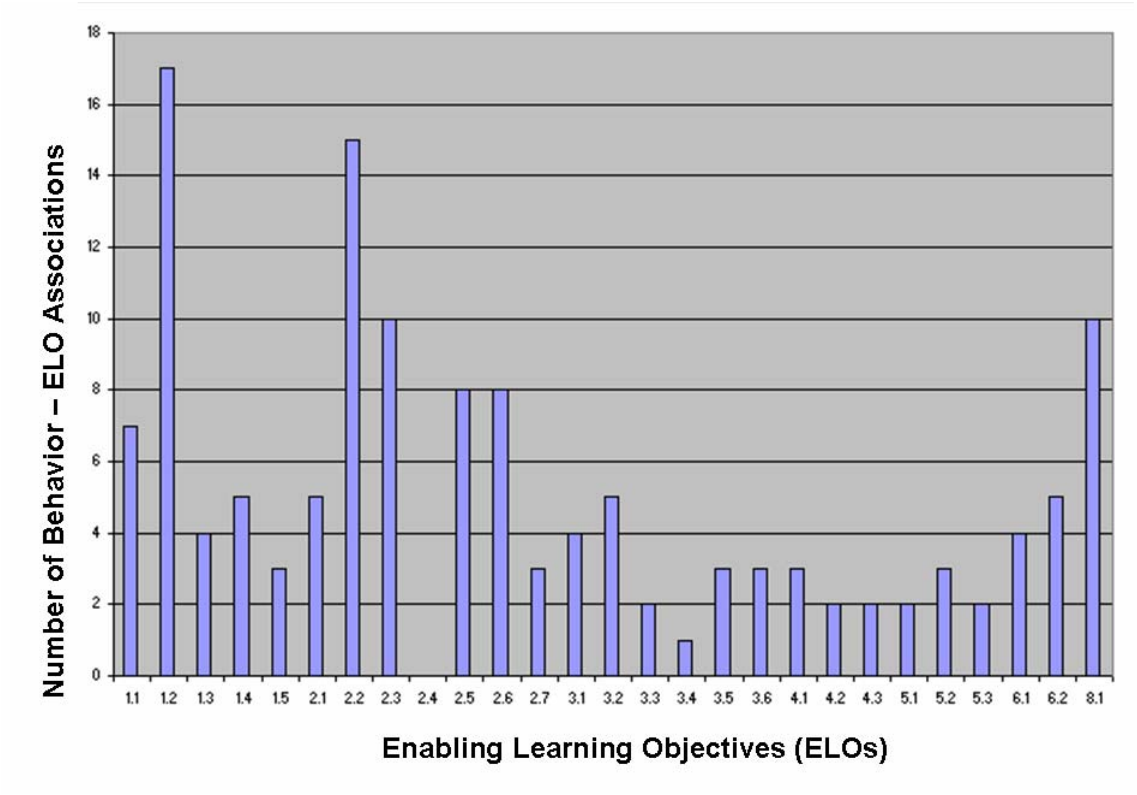

Figure 7. Enabling Learning Objective (ELO) Association Frequencies

The basis for linking specific behaviors and learning objectives proved to be difficult to identify. Generally, for a given behavior, there are one or maybe two user interface commands that must be used, and so, the learning objectives for these commands are linked to the response. Likewise, for a given response, there is usually a single tactic that the student is performing. Consequently, this behavior is also linked to the corresponding tactic. Beyond these two or three linkages, which are relatively straightforward, there may be many other portions of the interface or aspects of tactical knowledge that are implicated less directly. Under this project, a team of four individuals, composed of instructional designers and subject matter experts, developed the rationale for the behavior - learning objective linkages and applied it in a working session covering a period of approximately one day. Using this process, an initial set of 28 behaviors was associated with an average of nearly five learning objectives (4.9), with a range from two to 10 behavior-to-objective linkages (Figure 7).

It seems likely, however, that many of these links will prove to be unnecessary. Consider, for example, a relatively simple case. The repositioning of assets is required in the first minutes of every scenario – holding all assets in place is never an acceptable option. Consequently, data relevant to whether a student has mastered the learning objective related to the MAGTF XXI “move command” will be available quite early.

But, because the repositioning of assets is required throughout the scenario, 14 other links to this learning objective were defined for later elements of the training scenarios. It is likely that these other links will prove to be completely redundant and unnecessary; that is, if the student uses the move command appropriately at the outset of the scenario, it is unlikely that he/she will have forgotten it a minute or two later.

Based on our experience under this project, two actions are recommended. First, we believe it is advisable to restrict the scope of the learning objectives practiced within the simulation exercises. With a reduce scope, student behaviors will implicate a small set of competing hypotheses. The rationale for this recommendation is further discussed in Section 6.1, Limit Scope of Competing Hypotheses.

Second, we recommend that training effectiveness research be conducted on this issue. Specifically, applications of this technology with actual students will indicate the likelihood that the root cause is one of the primary learning objectives, and so, whether there is any additional benefit to including more distantly related topics. With an extremely limited set of data from a formative evaluation we conducted under this project, it appears that learners did tend to see a variety of sources to their learning difficulties, not just the primary learning objectives. If this result holds under more thorough study, adaptive learning systems should include a broad range of hypotheses for test. As a result, the need to develop efficient methods for hypothesis testing will be even greater.

3.2. Accumulating Performance History under SCORM

Once behavior actions are linked to learning objectives, a score for the learning objective can be maintained to show a measure of performance capabilities as described in the following Section, 3.2.1 Intelligent Scenario Delivery. This section also describes how to accumulate a performance history in order to evaluate trends and yet focus on most recent behaviors. The SCORM 2004 provides a mechanism to map the objective scores and success status indicators to global variables that can be accessed by other sessions to develop the history. However, as explained below, to save a history of just the objective scores, 3,780 variables would need to be mapped to global objectives. What makes it difficult is that objective scores are accessed by index, not by ID. This means for each index, the ID must be retrieved until the index with the correct ID is found. A worse case scenario would lead to 1x1033 calls to the LMS using the application interface (API) function getValue(). Instead, it is recommended that a SCO first retrieves all the IDs by calling getValue() once for each index. The SCO then stores the IDs in a local array for quick access. Once Shared State Persistence (SSP) is adopted by LMS vendors, storage and sharing of many variables will be much more efficient.

3.2.1. Intelligent Scenario Delivery

Several of our adaptive content delivery projects have used SCORM Sequencing and Navigation (S&N) to deliver only specific content. S&N rules enable/disable or skip content objects to provide this dynamic adaptive sequencing. Unfortunately, the terminology is not intuitive because, as described below, the SCO indicates that the learner has “passed” content that they will not see or get credit for. A better terminology

would be that specific content activities are “assigned” or “not assigned.” For example, in our Fighter Schoolhouse demonstration of 2004, in order for a novice to skip the expert level activities, the assessment SCO that determined the skill level of the learner marks the expert activities as “passed” so that a “skip if passed” rule would skip that activity.

3.2.2. No Access to SCO or Title ID

SCORM RTE (Run-Time Environment) Data Model, based on IEEE 1484.11.1, does not contain the ID or title of the SCO. So the SCO can not retrieve the title given to it by the content package’s organization. Suppose a SCO was developed to teach proper etiquette for meeting a person. The same SCO could be used in different courses with specific contextual titles. For example, “How to meet and greet an American” “How to greet a customer” “Greeting an Applicant” etc. The first page of the SCO was designed to display the lesson name and average amount of time required. It would be very beneficial if the SCO could call, for example, getValue(“cmi.title”).

4. MAJOR ACCOMPLISHMENTS

4.1. SCORM-Based Student Modeling

Although it is largely a by-product of the project, it is worthy of note that this prototype is one of only a small number of systems that have implemented dynamic student modeling under SCORM. A set of learning objective scores forms a student profile, or to use intelligent tutoring terminology, a student model. Changes to the student model variables are communicated to an LMS under SCORM using the Computer Managed Instruction (CMI) data model. Scores on the student modeling variables, in turn, are used in SCORM sequencing rules to select or re-schedule training activities that are most appropriate, based on learner performance to that point. In addition to a series of technical feasibility prototypes and a system developed for a training effectiveness study, this technology has been applied to a production program. An intelligent tutoring capability has been fielded under the USMC Expeditionary Fighting Vehicle (EFV) program using this SCORM-based student model.

4.2. Trend-Based Performance Assessment

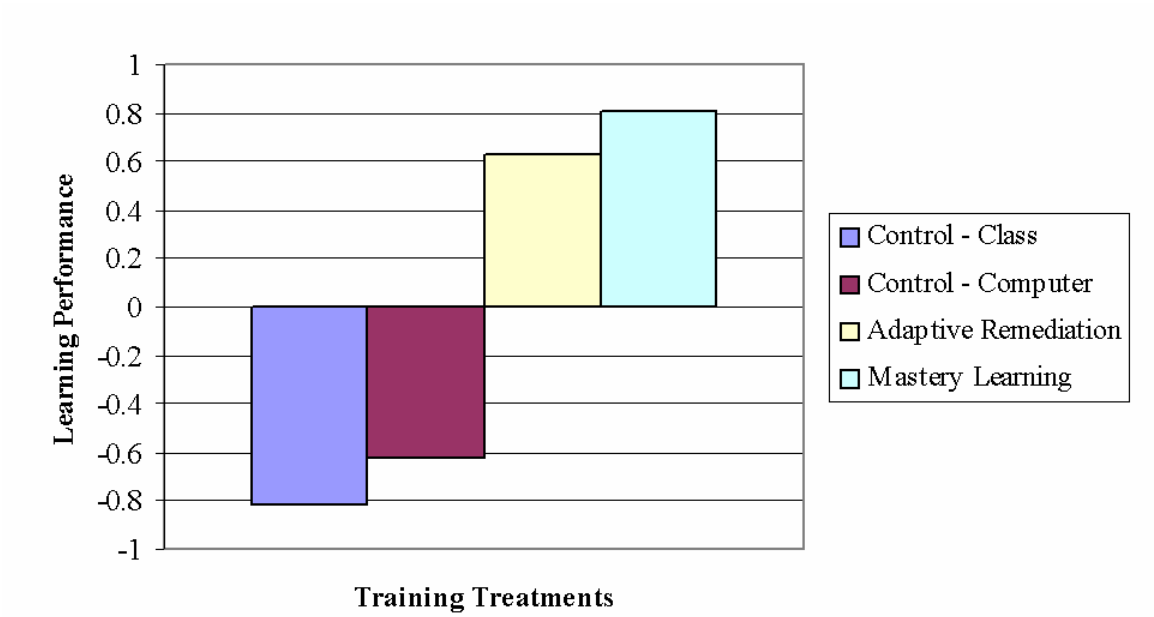

Figure 8. Effectiveness of Adaptive Learning Compared to Self-controlled Learning

Our initial research on scenario-based assessment used a much less complex problem-solving environment (see Perrin, Dargue, & Banks, 2003, or Perrin, Banks, & Dargue, 2004 for a more complete description of this research). Specifically, following the study of the computer-delivered, declarative content, problem-solving exercises were presented in narrative text. Students indicated their response to each problem solving exercise by selecting from a set of four options, one being the best and three being incorrect distracters. Each response was linked to the learning objectives that would be implicated by that response. Because the complexity and length of the course were limited, a single

error was sufficient for the software to schedule a review of the related content, before continuing with the practice problem solving activities. In short, the approach was, commit an error and then review all of the related topics that might be the root cause of the problem. Despite the simplicity of the method, learning performance improved significantly (Figure 8).

One might anticipate, however, that this very simple approach to assessment will not scale well to longer courses or more complex problem-solving contexts. Providing remedial reviews for all of the topics implicated by a single mistake makes sense neither from the perspective of good measurement practice nor the efficient use of training time and resources. A single mistake may reflect misunderstandings or forgetting of the related declarative knowledge studied in the interactive multimedia instruction (IMI), but it might also be the result of a momentary lapse in attention, a simple misperception, workload, or a host of other factors. What is needed is an approach that distinguishes between the root cause of the problem and these non-systematic, noise factors.

A moving average (a mean score calculated over the last set of a given number of opportunities) is one such measure. A moving average can be expected, over the long-term, to yield an unbiased estimate of the "true score" for each of the learning objectives implicated. That is, over sufficient time, averages based on randomly selected scenario-based training exercises can be expected to differentiate among competing root causes, assuming that the same competing hypotheses are not always implicated together as a set for all possible errors. In this case, any error that implicates one of the learning objectives as the root cause will also implicate the other(s).

```
objBaseName = "UsesMoveCommand";
// Get the stack of five values for the objective score
for (stackIndex=1; stackIndex <=5; stackIndex ++)
{
   objID = contentPackageGUID + objBaseName+"_" + stackIndex;
   objIndex = findObjective(objID);
   cmistr = "cmi.objectives." + objIndex + ".score.scaled"
   objScore[stackIndex] = doGetValue(cmistr);
}

 // Push the new score onto the stack
for (stackIndex=1; stackIndex <=4; stackIndex ++)
{
   objScore[stackIndex] = objScore[stackIndex+1];
}
objScore[5] = currentScore;

// Save the stack of five values for the objective score
for (stackIndex=1; stackIndex <=5; stackIndex ++)
{
   objID = contentPackageGUID + objBaseName+"_" + stackIndex;
   objIndex = findObjective(objID);
   cmistr = "cmi.objectives." + objIndex + ".score.scaled"
   doSetValue(cmistr, objScore[stackIndex]);
}

Where:
Constant contentPackageGUID is defined as a globally unique identifier for the content package (or course).
function doGetValue(name) is defined in the "standard" APIwrapper.js
function doSetValue(name, value) is defined in the "standard" APIwrapper.js
function findObjective(objID) is defined elsewhere
```

Figure 9. ECMAscript Example

Under this project, we implemented methods to calculate a moving average for each learning objective score. Thus, unlike previous research, the dynamic student model reflects performance trends, rather than single observations. As such, it has the potential to increase training efficiency, by reducing remedial review when it is not required.

Using the SCORM API and RTE Data Model, a SCO can store and retrieve a score of each particular action in the LMS. The assessment of each action that influences the performance measure of each ELO can increment or decrement the cumulative score of that performance measure. With SCORM 2004, the performance measurement score can be mapped to "global objectives" that other SCOs can map to for access. This means that the score can more accurately reflect the student performance over several scenarios delivered by different SCOs by assigning read and write privileges to that score for each SCO. To calculate an average of the measures affecting that score, the total score itself and a count of the measures both need to be stored. Any SCO that has access to those two numbers then can calculate the average. However, to more accurately reflect the current knowledge/performance of the learner using a moving average, each SCO needs access to each measurement to be used in the average. That is, a list of measurements needs to be stored and accessed.

For this project, a moving average of five measurements was believed to be adequate. Each measurement is stored in its own objective score. Therefore, the performance score for each learning objective is stored in five shared global objective scores. The five global objective scores serve as a cache or stack where each new measurement is "pushed" onto the stack and the oldest measurement is "popped" off the stack. Each SCO has to perform the manipulation of the stack by retrieving all the values, shifting the index (throwing away the oldest value), and storing the new stack of values. One method to code this using ECMAscript is shown in Figure 9.

The SCORM 2004 Content Aggregation Model (CAM) version 1.3 defines a mechanism for the "organization" of content to indicate the scope of sharing for the global objectives.

> `adlseq:objectivesGlobalToSystem` (optional, default = true) – This attribute indicates that any mapped global shared objectives defined in sequencing information (Refer to Section 5.1.1: *<sequencing> Element*) are either global to the learner and the content organization (`false`) or global for the lifetime of the learner within the LMS (`true`) across all content organizations.

Unfortunately, this defaults shared objectives to be "global to the system", which means they must be maintained by the LMS for as long as the student is registered in the LMS. So it is very important for the content package to indicate that these global objectives are only global locally to the content package unless absolutely sure that these objective scores will be used by another content package. This still can lead to a lot of storage space. For this project, we used 26 objectives requiring 130 global variables to store the history stack of five measures. Once LMSs support SSP, using a moving average across SCOs becomes more practical.

4.3. Method for Automatically Identifying Root Cause of Student Error

As noted previously, a given performance error may have numerous interpretations. Misunderstandings or the forgetting of the facts, rules, and procedures will produce learning performance errors, or the root cause may be due to factors beyond the scope of training (e.g., momentary lapses in attention during scenario execution). As also noted above, a moving average can be expected, over the long-term, to yield an unbiased estimate of the true score for each of the learning objectives implicated. Such an approach, however, may still unnecessarily extend training time. By random selection of scenarios, it may be some time before sufficient data are available to distinguish among different root causes of an observed error. A more efficient method for diagnosing learning-needs may be to actively identify and test competing hypotheses as to the source of the learning difficulty.

4.3.1. Identifying Competing Hypotheses

The identification of potential competing hypotheses begins when the score of a learning objective indicates that performance is poorer than expected, e.g., it falls below a threshold value. In this case, methods are needed to indicate whether a lack of mastery of the knowledge connected to this objective is the root cause. Alternatively, a lack of mastery of the content associated with other learning objectives may be the source of the performance difficulty.

We considered the use of several different metrics in the identification of competing hypotheses, including correlation coefficients. For this initial work, we selected a relative simple, straightforward measure. To identify competing root causes, we isolate the set of behaviors that implicated the learning objective under study. Then, we calculate the proportion of times other learning objectives were implicated in the same set of behaviors.

Consider, for example, the situation in which a repeated error observed during a simulation-based training exercise implicates learning objective 1 (LO1) and learning objective 2 (LO2). In other words, LO1 and LO2 are competing explanations for the performance errors observed. To clarify the source of the problem, a follow-on scenario is selected that only addresses the learning objective that has the highest probability of being the source of the student's error. To do this, two factors, relevance and independence, are used to calculate for each scenario in the pool of activities.

Relevance of a scenario simply reflects the fact that rules in the scenario implicate LO1 or LO2 or both. Relevance can be measured as the sum of the number of rules implicating LO1 and LO2.

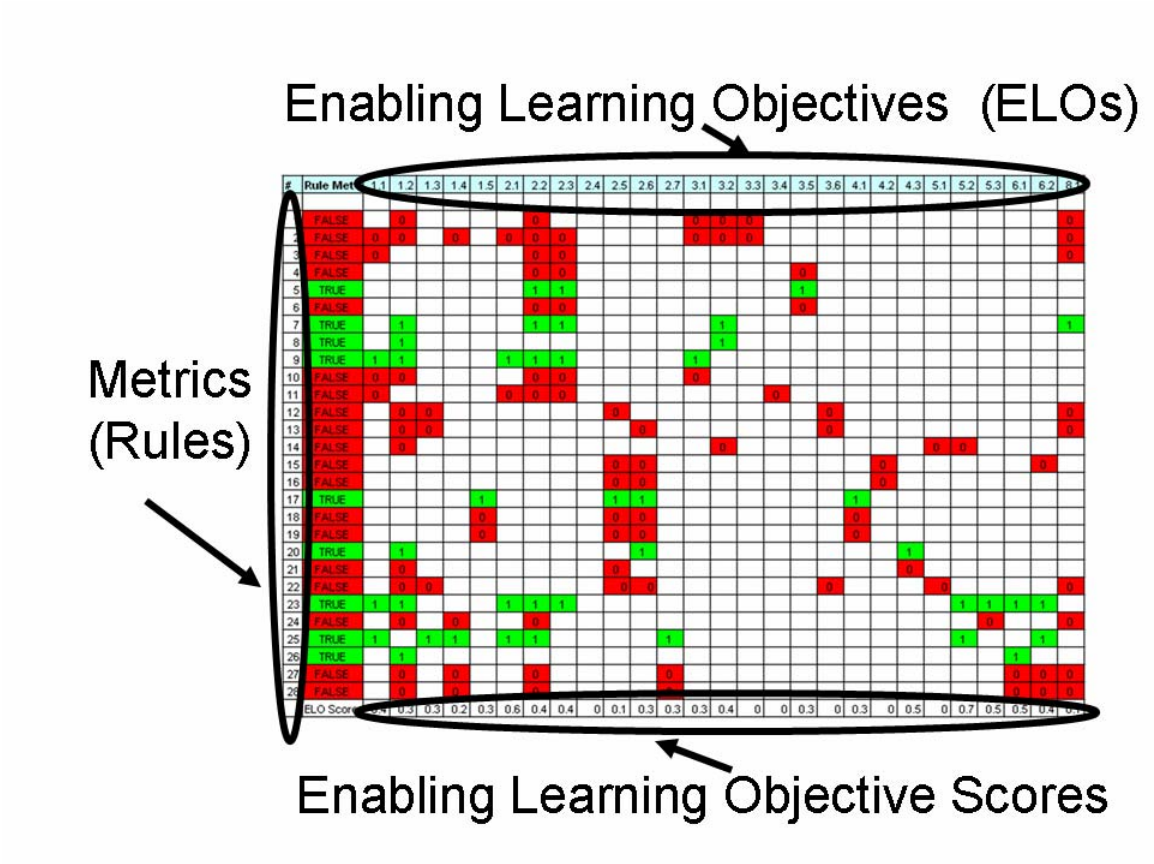

Figure 10. Rule Tracking Table

Independence of a scenario reflects the degree to which the scenario's rules implicate one of the two competing hypotheses, but not the other. Independence can be measured as the proportion of times LO is implicated in rules that implicate LO 1 – in this case, smaller is better.

In order to determine which behaviors caused the low score in the objective being studied, the performance assessment module needs to track the results of each rule. With respect to the development of the Performance-Based Advancement prototype, we chose to implement this in a table where each row represents a rule or behavior and each column indicates the influence on a particular objective score. Since we used 28 rules and 27 objectives, this table has 28 rows and 27 objectives. Figure 10 shows this table with each cell colored and populated based on whether each rule passed or failed. The row across the top lists each of the ELOs (see Appendix A for a complete list of the ELOs). The left hand column lists each of the metrics (rules) that were developed for the in-stride breach mission (see Appendix B for a complete list of the metrics developed). The row at the bottom shows the scores for the ELOs, which is calculated using a moving average for each metric evaluated for the specific ELO, at the end of the in-stride breach mission.

For a single SCO/scenario, this is simply maintained in the performance assessment module and does not need to be sent to the LMS to be shared with other SCOs. If we

were to use a moving average history of each metric, we would need to store 28 x 27 x 5 = 3,780 variables that would have to be sent to and retrieved from the LMS. If the specific array index assigned by the LMS for each objective was known, a minimum of eight lines of code and one call to the LMS using getValue() for each variable would be required. However, if we were to use the searching routine used in the function findObjectives, a worse case would require 304,888,344,611,713,860,501,504,000,000 calls to LMS using getValue() for each one of the 3,780 values. This would not be practical without using Shared State Persistence. In the case of this project, each scenario used the same set of performance evaluation rules with different parameters so a more efficient, yet still impractical, coding is possible requiring only 140 variables rather than 3,780 times the number calls required.

For this project, only the objective score needed to be averaged between scenarios because only the behaviors observed in the current scenario were used for hypothesis identification. The cumulative moving average of the objective scores then could be used for trend analysis and helping to weigh the competing hypotheses.

#	Rule Met	1.1	1.2	1.3	1.4	1.5	2.1	2.2	2.3	2.4	2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
1	FALSE		0					0						0	0	0												0
2	FALSE	0	0		0		0	0	0					0	0	0												0
3	FALSE	0						0	0																			0
4	FALSE							0	0									0										
5	TRUE							1	1									1										
6	FALSE							0	0									0										
7	TRUE		1					1	1						1													1
8	TRUE		1												1													
9	TRUE	1	1				1	1	1					1														
10	FALSE	0	0					0	0					0														
11	FALSE	0					0	0	0								0											
12	FALSE		0	0							0								0									0
13	FALSE		0	0								0							0									0
14	FALSE		0												0								0	0				
15	FALSE										0	0									0						0	
16	FALSE										0	0									0							
17	TRUE					1					1	1								1								
18	FALSE					0					0	0								0								
19	FALSE					0					0	0								0								
20	TRUE		1									1										1						
21	FALSE		0								0											0						
22	FALSE		0	0							0	0							0				0					0
23	TRUE	1	1				1	1	1															1	1	1	1	
24	FALSE		0		0			0																	0			0
25	TRUE	1		1	1		1	1					1											1			1	
26	TRUE		1																							1		
27	FALSE		0		0			0					0													0	0	0
28	FALSE		0		0			0					0													0	0	0
	ELO Score	0.4	0.3	0.3	0.2	0.3	0.6	0.4	0.4	0	0.1	0.3	0.3	0.3	0.4	0	0	0.3	0	0.3	0	0.5	0	0.7	0.5	0.5	0.4	0.1

Figure 11. Rule Tracking Table after Rule Failure Analysis has been Performed

Figure 11 shows the same set of rules and objective scores after rule failure analysis has been performed. In the example shown, objective 3.1 has been identified as the first objective that is below the failure threshold. The column representing this objective is

highlighted in orange. The row representing each rule that caused a zero to be added to the score of that objective is identified by a grey highlight.

For each of the objectives that were also affected by the failed rules, we calculated the proportion of times other objective scores were implicated in the same set of errors. So for the example shown, objectives 1.4 and 2.1 were only implicated in one out of three times (33%) Whereas objectives 1.1, 2.3, 3.2, 3.3, and 8.1 were implicated in two of the three errors (67%), and objectives 1.2 and 2.2 were implicated in all three errors (100%). This can be seen by counting the number of red boxes in the grey rows for each of the columns. The columns representing each of the competing hypotheses that have a correlation greater than 50% are highlighted in yellow.

4.3.2. Intelligent Scenario Selection

ELO being diagnosed		ELOs that are potentially independent from the ELO being diagnosed																										
	Relevant scenario	1.1	1.2	1.3	1.4	1.5	2.1	2.2	2.3	2.4	2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
3.1	Full			1.3	1.4	1.5					2.5	2.6	2.7				3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	
	Obscure & Secure			1.3	1.4	1.5					2.5	2.6	2.7		3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
3.2	Full			1.3	1.4	1.5					2.5	2.6	2.7				3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	
	Obscure & Secure	1.1		1.3	1.4	1.5	2.1	2.2	2.3	2.5	2.6	2.7	3.1			3.3	3.4	3.5	3.6	4.1	4.2	4.3			5.3	6.1	6.2	8.1
3.3	Full	1.3		1.5							2.5	2.6	2.7				3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	
3.4	Full		1.2	1.3	1.4	1.5					2.5	2.6	2.7	3.1	3.2	3.3		3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
3.5	Full	1.1	1.2	1.3	1.4	1.5	2.1				2.5	2.6	2.7	3.1	3.2	3.3	3.4		3.6	4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
3.6	Full	1.1			1.4	1.5	2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5		4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	
	Obscure & Secure	1.1			1.4	1.5	2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5		4.1	4.2	4.3	5.1	5.2	5.3	6.1	6.2	
4.1	Full	1.1	1.2	1.3	1.4		2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5	3.6		4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
	Obscure & Secure	1.1	1.2	1.3	1.4		2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5	3.6		4.2	4.3	5.1	5.2	5.3	6.1	6.2	8.1
4.2	Full	1.1	1.2	1.3	1.4	1.5	2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1		4.3	5.1	5.2	5.3	6.1		8.1
	Obscure & Secure	1.1	1.2	1.3	1.4	1.5	2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1		4.3	5.1	5.2	5.3	6.1		8.1
4.3	Full	1.1		1.3	1.4	1.5	2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2		5.1	5.2	5.3	6.1	6.2	8.1
	Obscure & Secure	1.1		1.3	1.4	1.5	2.1	2.2	2.3				2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2		5.1	5.2	5.3	6.1	6.2	8.1
5.1	Full	1.1			1.4	1.5	2.1	2.2	2.3				2.7	3.1		3.3	3.4	3.5		4.1	4.2	4.3			5.3	6.1	6.2	
	Obscure & Secure	1.1			1.4	1.5	2.1	2.2	2.3				2.7	3.1		3.3	3.4	3.5		4.1	4.2	4.3			5.3	6.1	6.2	
5.2	Full	1.3			1.4	1.5			2.3		2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1		5.3	6.1		8.1
	Obscure & Secure	1.3			1.4	1.5			2.3		2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1		5.3	6.1		8.1
5.3	Full	1.3				1.5					2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1					
	Obscure & Secure	1.3				1.5					2.5	2.6	2.7	3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1					
6.1	Full	1.1		1.3		1.5	2.1		2.3		2.5	2.6		3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3			
	Obscure & Secure	1.1		1.3		1.5	2.1		2.3		2.5	2.6		3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3			
6.2	Full	1.1		1.3		1.5	2.1		2.3		2.5	2.6		3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3			8.1
	Obscure & Secure	1.1		1.3		1.5	2.1		2.3		2.5	2.6		3.1	3.2	3.3	3.4	3.5	3.6	4.1	4.2	4.3	5.1	5.2	5.3			8.1

Figure 12. Relevance and Independence Data for Learning Objectives

Figure 12 shows a table of the relevance and independence data for each objective (3.1 to 6.2) in the Full, Obscure (part-task), and Secure (part-task) scenarios. At the current time, there is no difference between the Obscure and Secure scenario rule sets, so their independence data is exactly the same. In this example, the objective we are diagnosing is 3.1. The rows for that objective are highlighted in orange. The columns representing each of the competing hypotheses are highlighted in yellow as in the previous table. As shown here, objective 3.1 can be assessed independently from the competing objectives 3.2, 3.3, and 8.1 only in either the Obscure or Secure scenarios but not the Full Mission scenarios as identified by the green highlights. Therefore, in this case, we should deliver

an Obscure or Secure scenario to the learner next to determine if the root cause is objective 3.1, 3.2, or 3.3. If we look ahead, the next pair of rows representing objective 3.2 indicate that there is also in independence between 2.3 and 3.3 in the scenario, so this should be successful in determining which objective is misunderstood and is the cause of the mistakes.

4.3.3. Using Simple Sequencing to Deliver the Selected Scenario

Simple Sequencing, as implemented in SCORM 2004, provides rules that can be used to select the next learning activity (SCO) to launch. These rules can use a set of factors such as scores completion status and success status. However, the current behavior of the RTE defined in the SCORM 2004 Sequencing and Navigation (S&N) document resets completion status for each activity node upon each attempt of the aggregation. Therefore, we only recommend using score and success status if there might be any remediation looping involved. In the case of this prototype, twelve scenarios were authored. Since we only want to delivery any scenario only once, we need to map the completion status of each scenario to a "global objective." Since we also need to dynamically assign scenarios, we also need an indication of whether each scenario is assigned. Therefore, each scenario, in addition to mapping the five measures of each objective, must have two unique "global objectives." These objectives are named for the scenario and the purpose. So, for example, scenario 1 has unique objectives "scenario_1_completed" and "scenario_1_not_assigned." The node(s) to launch and assess scenario 1, then have rules to "skip" if "scenario_1_completed" or if "scenario_1_not_assigned." The extensible mark-up language (XML) coding for this is shown in Figure 13.

```
<imsss:sequencingRules>
  <imsss:preConditionRule>
    <imsss:ruleConditions
conditionCombination="any">
      <imsss:ruleCondition

referencedObjective="scenario_1_completed"
condition="satisfied" />
      <imsss:ruleCondition

referencedObjective="scenario_1_not_assigned"
condition="satisfied" />
    </imsss:ruleConditions>
    <imsss:ruleAction action="skip" />
  </imsss:preConditionRule>
</imsss:sequencingRules>
```

Figure 13. XML Coding Example

The assessment SCO sets the success_status of “scenario_1_completed” to “passed” to ensure that the scenario is not delivered to the student if the aggregation is retried. The assessment SCO also sets the success_status of “scenario_*x*_not_assigned” to “passed” for each of the scenarios that we do not want the student to experience next. In the example shown in Figure 12 and Figure 13, the selection algorithms identified Obscure and Secure scenarios, so the “scenario_*x*_not_assigned” for all of the full mission scenarios should be set to “passed.” Example ECMAscript code to skip the full mission scenarios is shown in Figure 14. In this example, all obscure missions have an ID that ends in “_2” and secure missions have IDs that end in “_3.” For example, “scenario_2” is a full mission, whereas “scenario_2_2” is an obscure mission and “scenario_2_3” is a secure mission.

```
Function unassignMissions(scenarioType)
{
  switch (scenarioType)
  {
    case “Full”:
      scenarioString = “”;
      break;
    case “Obscure”:
      scenarioString = “_1”;
      break;
    case “Secure”:
      scenarioString = “_2”;
  }
  for (scenario=1; scenario <=6; scenario ++)
  {
    objID = scenario +"_" + scenario + scenarioString;
    objIndex = findObjective(objID);
    cmistr = "cmi.objectives." + objIndex + ". success_status"
    doSetValue(cmistr, “passed”);
  }
}
```

Figure 14. ECMA Script to Skip Full Mission Scenarios

4.4. Developed an Architecture for Simulation-Based Training Management Capability

Under a previous effort for DARWARS, MÄK developed a mechanism for SCORM-Conformant SCOs to install, launch, and monitor an HLA-based simulation. For this project, we needed the ability to choose scenario files and to be able to continuously monitor and assess the learner. The modifications performed under this project included adding:

1. Ability to launch using specific scenario
2. Detonation target and entity in area calculations performed by Collector
3. Event-driven interface between SCO ECMAScript and Collector Applet
4. Performance Assessment Rule Engine

5. Sending scores to SCORM API messages
6. Sending global variables to LMS for sequencing rules

Figure 15 shows the final architecture. The following sections describe the architecture in more detail.

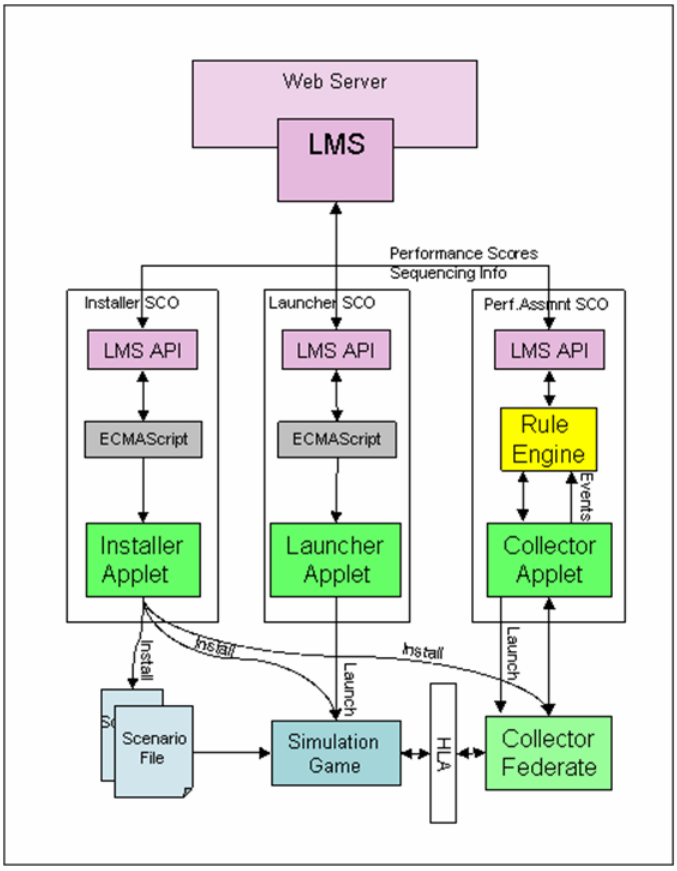

Figure 15. Performance-Based Advancement Prototype Architecture

4.4.1. Performance Assessment Mechanisms

MAGTF XXI uses the Real-Time Platform Reference Federation Object Model (RPR FOM, (pronounced "reaper" Fom) which was based on the Distributed Interactive Simulation (DIS) data packets. Most of the information required to assess the learner's performance is included in the HLA data. However, some calculations have to be performed to derive intended performance. For example, although the HLA data includes where a detonation occurs, it is more meaningful to know if that is within an acceptable distance from a target or if it is within a certain area. Similarly, it is more meaningful to calculate whether an entity is within a particular area rather than just knowing it's latitude and longitude position. The RPR FOM includes an "Environmental processes" data section that the game uses to contain graphical objects that the instructor can see but the student cannot. This "layer" was used to provide an innovative way for the subject matter expert (SME) to indicate these particular areas. A naming convention was used as metadata to indicate the purpose of the area and assign a scoring weight to the area. For

example, area “B1” is the optimum area for a breach and “B2” is less optimal. Similarly, “Smoke 1” is the optimum are for smoke detonations, “Smoke 2” is less optimum, and “Smoke 3” is acceptable. Figure 16 shows the game with the instructor layer visible. Areas in cyan are entity position scoring areas. Areas in black are the breach areas and smoke detonation areas. Areas in magenta are inaccessible areas.

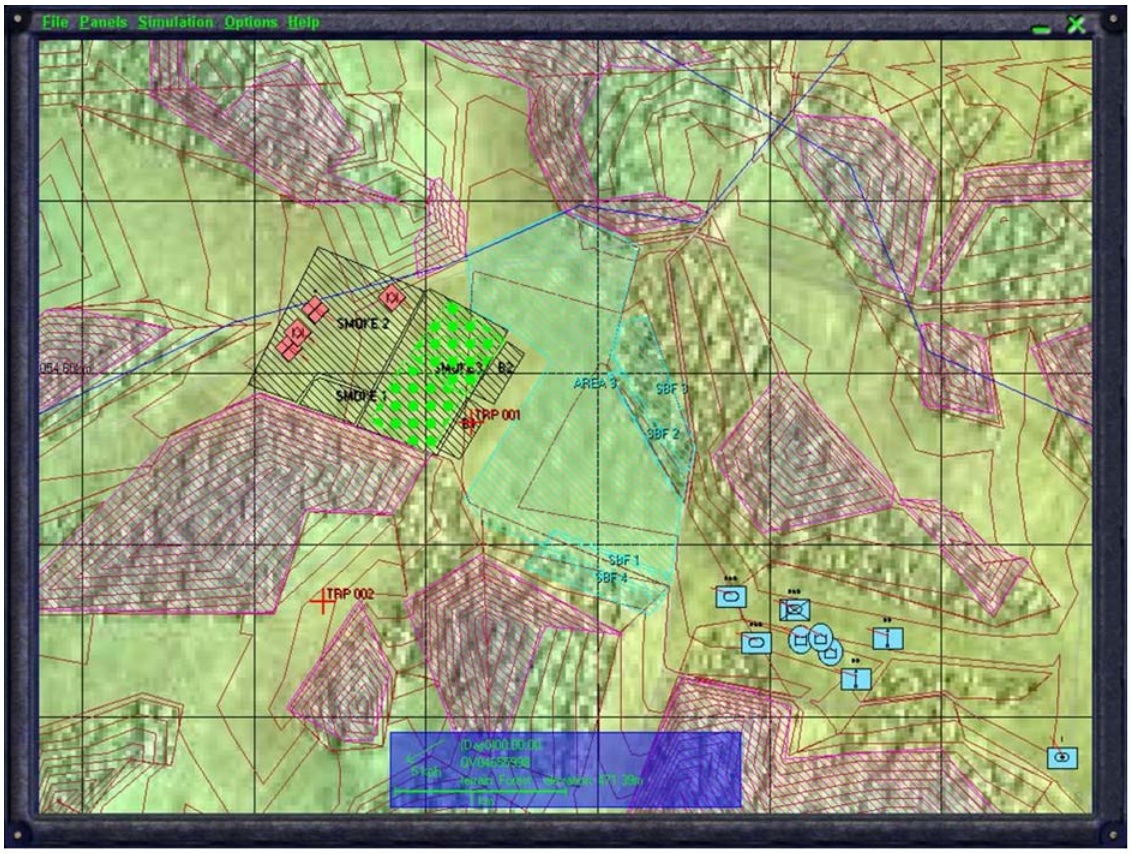

Figure 16. Position Scoring Areas

In this manner, the SME was able to create graphically draw a majority of the metrics used by the rules. The areas were different shapes and positions in different scenarios based on, for example, threat positions, threat capabilities, and wind direction. The collector federate/applet pair performs “entity in polygon area” calculations and sends an event when an entity enters or leaves an area. For detonations, the collector calculates which area the detonation is within and determines the intended target by fining the nearest threat within a specified distance from the detonation. The detonation event from the collector to the rule engine contains this added information. Other events included entity started movement, entity stopped movement, and entity fired round.

4.4.2. Scenario Assessment Pairing

To do this using the SCORM, the game launcher applet uses a parameter in the hyper-text mark-up language (HTML) page. In this manner, each scenario has its own HTML page. The assessment module was written in a different way so that all scenario assessments were performed by the same HTML page launched in different ways. The LMS launches

the assessment module with a parameter indicating which scenario is loaded passed in the universal resource locator (URL), similar to the command line method of passing parameters. In this way, one single HTML page is called using different parameters at different nodes in the activity tree. It is important to note here that SCOs are not the only reusable items in a SCORM environment. Different SCOs can reuse the same HTML pages. Therefore, after launching the game, the launcher SCO could load the assessment module web page. This method would lower the risk of launching a scenario without an assessment module. However, the launcher SCO itself would become less usable as it could not be used without modifications in a practice or demonstration aggregation without assessment. Possibly, the most adaptable, reusable approach is to pass two parameters to the launcher/assessment SCO. One parameter to indicate the scenario to load and another to indicate the assessment mode. A more hacker-resistant method is to put the parameter(s) in cmi.launch_data rather than on the command line. There are several benefits and drawbacks to each approach that might be better addressed in a "best practices" document. What is important to this discussion and all performance assessment modules, is that specific performance assessment rules must be used for each unique scenario – that is, the assessment engine needs to know which scenario the learner is performing.

4.5. Implemented Several Hours of Level3/Level4 Courseware

In order to provide a scenario pool to demonstrate the scenario selection capability, in-stride breach, MAGTF XXI missions were developed for the prototype. The scenarios differed based on the capabilities of the enemy, wind direction, and length (either full mission or part-task) as discussed earlier in Section 2.1 Identify Research Scenario. For each mission, metrics were tailored to account for the factors that would affect student actions, such as optimal locations for support by fires (SBF) positions and optimal placement of smoke mortars. These missions all work with the automated performance assessment and scenario selection capability, and therefore, provide simulation-based training.

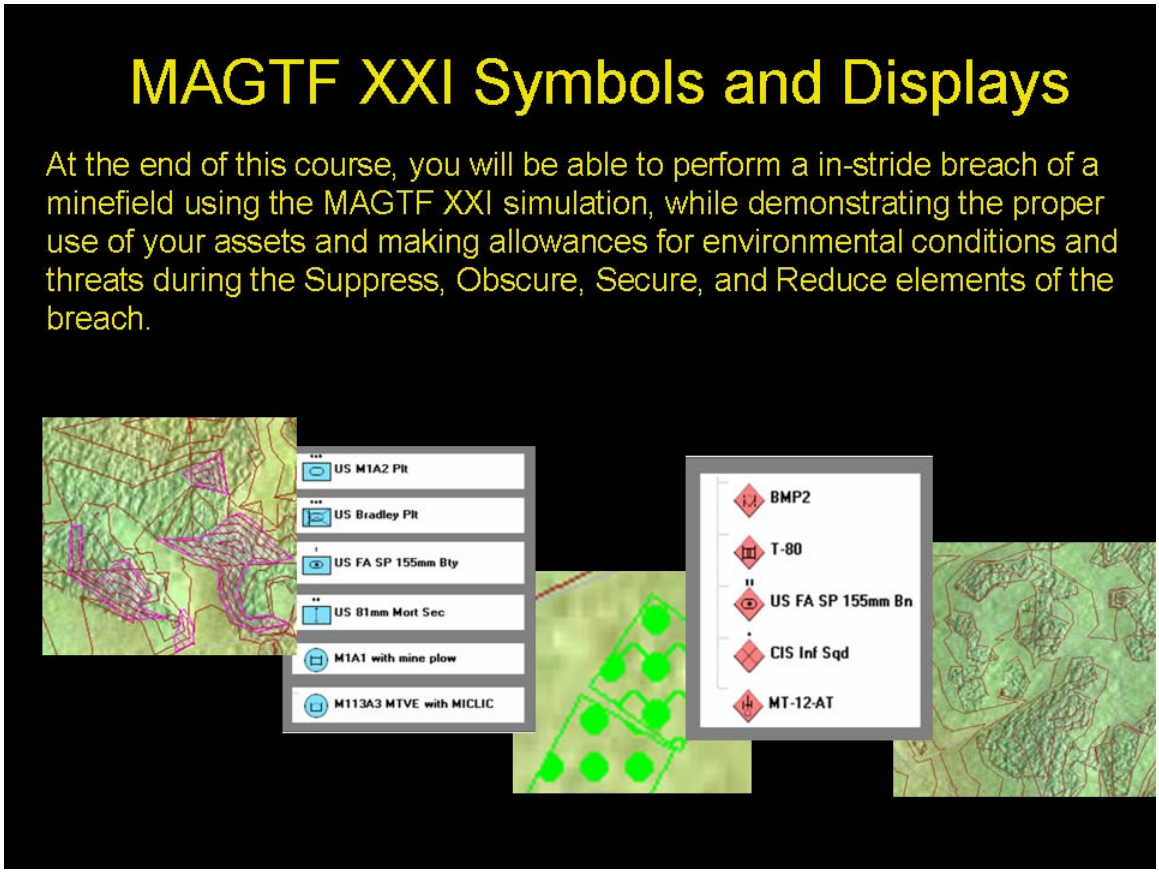

Figure 17. ICW Example – MAGTF XXI Symbology

Additionally, approximately 45 minutes of level 1/level 2 ICW was developed to provide the pre-requisite declarative knowledge requirements for performing the in-stride breach using MAGTF XXI. The ICW was primarily implemented with Microsoft PowerPoint, however, there were several Flash movies embedded to demonstrate the use of the MAGTF XXI commands. Figure 17 provides an example of the ICW developed to provide instruction on MAGTF XXI symbology, and Figure 18 is an example of the in-stride breach tactics ICW.

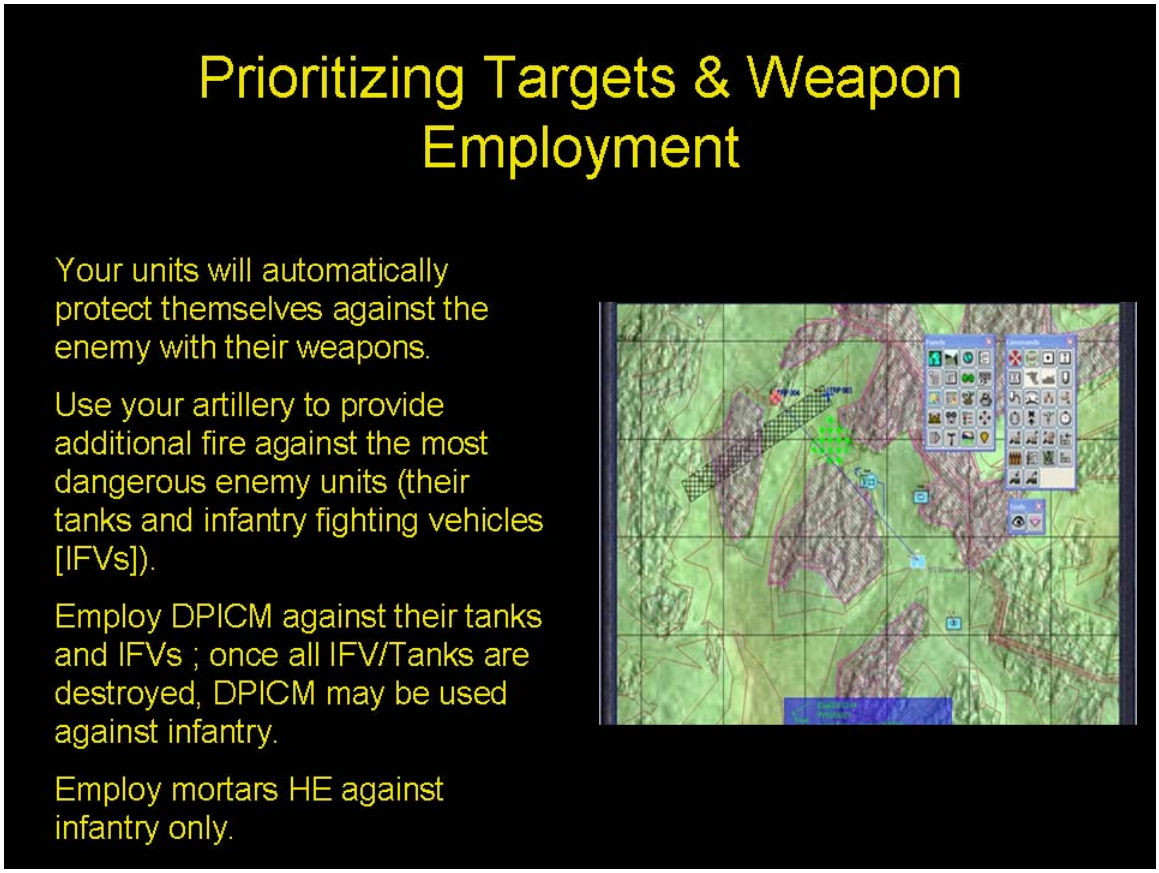

Figure 18. Example of Tactics ICW

5. SHORTFALLS

The only shortfall to report is that the scenario pool for the prototype is limited to a single tactical mission, in-stride breach. A broader range of tasks would be desirable in order to fully evaluate our learning needs hypothesis testing methods. The current demonstration includes several scenarios, but each scenario involves the performance of the same basic task – breaching a minefield. Consequently, the expected student behaviors are largely the same for each scenario and these behaviors are linked to the same learning objectives. Consequently, the selection of follow-on scenarios that elicit behaviors that implicate some, but not all of the same learning objectives as the previous scenarios is limited. In short, our ability to distinguish among competing learning needs hypotheses is restricted as alternative tasks that elicit behaviors with different linkages to learning objectives have not been developed.

Nonetheless, the current technology feasibility demonstration provides data relevant to our learning needs hypothesis testing methods. The learning objectives in this demonstration system are generally confirmed with differing numbers of opportunities for performance or under differing timeframes. Consider again the learning objective for the move command, which must be performed early within the first scenario, as noted above. Failure to issue a move command for a period of time could be the result of a lack of understanding of the interface or of the tactic, or it could be due to workload. However, within this timeframe, lack of mastery of the interface seems to be the most likely cause. Since little else is occurring in the scenario at the outset, workload is an unlikely cause. Ascribing the inaction to a lack of tactical knowledge would imply that the student has understood few of the tactics, as most require unit movements. On the other hand, any movement of assets largely excludes the interface from consideration, and the root cause is most likely tactical knowledge or workload.

With the breadth of scenarios currently under development for this demonstration system, our learning hypothesis testing methods are generally driven by the number of opportunities required to infer mastery. Learning objectives related to the MAGTF XXI interface will generally be implicated if they not elicited within the first few opportunities. Similarly, these interface objectives will be considered to be mastered with relatively few performances. The learning objectives related to tactical knowledge, on the other hand, are more clearly implicated when the student has shown that he/she knows the interface, but fails to use it to implement a tactic. Finally, we can expect workload to follow a power function of practice, a relationship backed by an extensive body of research (see for example, Anderson, 2000). Simply put, workload is less likely to be the root cause as the number of opportunities to perform the behavior increases.

The mechanism to account for performance opportunities in inferring root causation is built into our student modeling approach. Specifically, by setting the number of observations to be used within the calculation of the moving average, these differences in performance opportunities can be reflected directly within the student model. So, for example, the average for the move command might use three observations and have a threshold for mastery set at 0.66 (or in other words, it was used in two out of three

opportunities). Assuming that the student issued the move command at least twice at the outset of the scenario, the search for a root cause in performance errors would then shift to workload or tactics. By a similar logic, the moving average for a tactic that is performed early within the scenario could be set to four out of six opportunities. While the proportion of success is the same for this learning objective, the tactic would not be considered the root cause until the move command was already mastered or the learning need was remediated.

6. LESSONS LEARNED

6.1. Limit Scope of Competing Hypotheses

We intentionally included training on both in-stride breach tactics and MAGTF XXI controls and displays. The rationale, discussed with MÄK and the JADL at I/ITSEC 2005, was that this presented a situation where competing hypotheses were nearly guaranteed. Almost all errors might be a result difficulties remembering tactical or interface information.

However, to adequately address all of the user interface, command, and tactic knowledge required to perform the in-stride breach mission using MAGTF XXI, approximately 45 minutes of IMI was required and produced as part of the prototype. Given the breadth of this training prior to any simulation practice there were more competing hypotheses than anticipated ("28 behaviors was associated with an average of nearly 5 learning objectives (4.9), with a range from 2 to 10 behavior-to-objective linkages" as noted in section 3.1 Establishing Behavior - Learning Objective Associations.) The training could have easily been broken into interface and tactics components, with separate simulation exercises and scenario selections for each, i.e., there would still have been numerous competing hypotheses.

6.2. Selection of Target Population

The selection of a novice population was driven by the accessibility of the population for formative evaluation during the prototype development. Due to time and budget constraints, the conduct of formative evaluation during the prototype development effort was not feasible. The use of in-house personnel as participants in the limited formative evaluation was a more viable option.

However, the prototype content is so complex that the target user should be USMC personnel with prior knowledge of MAGTF operations. As mentioned previously, extensive up-front instruction is required to perform the supposed simple in-stride mission that was developed for the prototype.

7. NEXT STEPS

7.1. Training Effectiveness Evaluation

The Performance-Based Advancement prototype successfully demonstrated the feasibility for leveraging HLA and SCORM standards to implement an individual training assessment, modeling, and scenario selection capabilities. Specifically, we demonstrated the use of:

- HLA to SCORM bridges,
- Performance assessment engines that "fire" the behavioral rules and update learning objective scores, and
- Logic that uses these objective scores to select subsequent scenarios.

However, the training effectiveness of this method, which had been proposed as a Phase II (Option) effort, has yet to be evaluated. There is little reason to field methods that do not benefit the learner, and the training literature is replete with claims for the benefit of a technology that have not been born out in practice. Thus, demonstrating the training effectiveness of the Performance-Based Advancement methods is integral to positioning this approach for implementation and operational use.

7.2. Extend Prototype

The prototype successfully met our goals to demonstrate automated performance assessment of individual training in a simulation-based environment and intelligent scenario selection to efficiently diagnose the root cause of student error. These methods can be extended to provide a means of providing tailored feedback, based on the diagnosis of student error, and/or remediation to appropriate sections of pre-requisite declarative knowledge instruction.

The current version of the Performance-Based Advancement prototype has the framework in place to achieve these goals. To do this, would primarily involve the development of feedback statements associated with individual learning objectives and conversion of the existing ICW into SCORM 2004 courseware, which can be done with minimal effort.

7.3. Transition Prototype to the Military Training Community

As MAGTF XXI is already being used by the USMC for various training activities, the prototype demonstrates applicability to USMC training in particular. In order for the warfighter to benefit from this research, we need to identify ways to work with the military training community to refine the prototype for implementation in operational training.

There already appears to be interest by the Department of Defense (DoD) training community in pursuing the transition of this technology. Specifically, Mr. Dan Gardner (Director, Readiness and Training, Policy and Programs, ODUSD) recommended that we pursue funding to transition this technology to the training community when he viewed a demonstration of the Performance-Based Advancement prototype during the 2006 JADL

Implementation Fest. Additionally, J.P. McDonough (Modeling & Simulation Analyst, USMC Training and Education Command) and J.M. "ATIS" Lozano (Marine Net Officer at the College of Continuing Education) expressed interest in the Performance-Based Advancement prototype during the 2006 JADL Implementation Fest.

8. FEEDBACK ON PROCESS FROM START TO FINISH

8.1. Prototype Deliverable Expectation

There is no Contractor Data Requirements List (CDRL) in our contract for a software delivery of the prototype. However, delivery of the prototype has been an expectation communicated from the JADL staff since the Kick-Off Meeting. We are going to provide the JADL with a copy of the prototype, as this mutually benefits the customer and Boeing.

8.2. Final Report Template

The final report instructions identified in the CDRL differ from the template that the JADL briefed during the kick-off meeting. In order to satisfy the CDRL and the JADL expectations, we used both sets of instructions to prepare the final report.

8.3. Change of Alternate Contracting Officer Representative (ACOR)

The change of ACOR was confusing as it was unclear whom we should be communicating with after we were notified of the change. This is because our primary contact since the change has not been the newly-appointed ACOR.

9. PUBLISHED PAPERS

Biddle, E., Perrin, B., Dargue, B., Pike, W.Y., Marvin, D., & Lunsford, J. (2006). Performance- based advancement using SCORM 2004. To appear in, *Proceedings to the Interservice/Industry Training, Simulation, and Education Conference* (Orlando, FL; Dec. 4-7).

Perrin, B., Biddle, E., Dargue, B., Pike, W.Y., & Marvin, D. (2006). SCORM as a coordination backbone for dynamically blended learning. *Proceedings to the Society for Applied Learning Technology (SALT) Conference* (August 23-25, Arlington, VA).

Acronyms

Acronym	Definition
ACOR	Alternate Contracting Officer Representative
ADL	Advanced Distributed Learning
API	Application Interface
CAM	Content Aggregation Model
CDRL	Contractor Data Requirements List
CMI	Computer Managed Instruction
DIS	Distributed Interactive Simulation
DoD	Department of Defense
EFV	Expeditionary Fighting Vehicle
ELO	Enabling Learning Objective
HLA	High Level Architecture
HTML	Hyper-text Mark-up Language
I/ITSEC	Interservice/Industry Training, Simulation and Education Conference
ICW	Interactive Courseware
IMI	Interactive Multimedia Instruction
IPA	Integrated Prototype Architecture
JADL	Joint Advanced Distributed Learning Co-Laboratory
LMS	Learning Management System
LO	Learning Objective
MAGTF	Marine Air Ground Task Force
PMTRASYS	Program Management Training Systems
RPR FOM	Real-Time Platform Reference Federation Object Model
RTE	Run-Time Environment
S&N	Sequencing and Navigation
SALT	Society for Applied Learning Technology
SBF	Support By Fires
SCO	Shareable Content Object
SCORM	Shareable Content Object Reference Model
SME	Subject Matter Expert
SSP	Shared State Persistence
TDS	Tactical Decision-Making Simulation
URL	Universal Resource Locator
USMC	United States Marine Corps
XML	Extensible Mark-up Language

References

Anderson, J.R. (2000). *Learning and Memory: An Integrated Approach*. New York: Wiley.

Anderson, J.R. (1993). Rules of the Mind. Hillsdale, N.J.: Erlbaum.

Goldstein, I.L., & Ford, J.K. (2002). *Training in Organizations*. Belmont, CA: Wadsworth.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press.

Perrin, B., Dargue, B., & Banks, F. (2003). Dynamically adapting content delivery: An effectiveness study and lessons learned. Paper presented at the *Interservice/Industry Training, Simulation, and Education Conference*, Orlando, FL, December 2003.

Perrin, B., Banks, F., & Dargue, B. (2004). Student vs. software pacing of instruction: An empirical comparison of effectiveness. Paper presented at the *Interservice/Industry Training, Simulation, and Education Conference*, Orlando, FL, December 2004.

Appendix A: Enabling Learning Objectives

Learning Objective Hierarchy

Terminal Learning Objective: Perform a hasty breach using the MAGTF XXI simulation demonstrating proper use of assets and allowances for environmental and threat conditions during the Suppress, Obscure, Secure, and Reduce phases of the breach.

LO1: **(Learning Objective)**: Interpret symbols and displays used in MAGTF XXI simulation for this training

ELO1.1: Interpret symbols for terrain (contour lines, vegetation, inaccessible areas).
ELO1.2: Interpret symbology and capabilities for assets.
ELO1.3: Interpret symbology and capabilities for threats.
ELO1.4: Interpret symbology for obstacles.
ELO1.5: Interpret symbology for wind direction.

LO2: Indicate MAGTF XXI commands necessary to perform a hasty breach.

NOTE: Assumes that trainee will begin with MAGTF XXI screen displaying simulation training area, so that navigation buttons are unnecessary.

ELO2.1: Indicate use of the terrain analysis tool.
ELO2.2: Indicate use of the multipoint move.
ELO2.3: Indicate route following on multipoint move.
ELO2.4: Indicate use of the target command for direct fires.
ELO2.5: Indicate use of the fire/mission command for Smoke or DPICM.
ELO2.6: Indicate use of the Mortar Attack command for Smoke or HE.
ELO2.7: Indicate use of the Engineering panel and breach button.

LO3: Indicate proper use of assets and allowances for environmental and threat conditions during Suppress phase of a hasty breach.

ELO3.1: Indicate selection of support by fire locations.
ELO3.2: Indicate asset selection.
ELO3.3: Indicate asset ordering.
ELO3.4: Indicate movement to use terrain masking.
ELO3.5: Indicate coordinated arrival and correct orientation.
ELO3.6: Indicate target prioritization and weapon employment.

LO4: Indicate proper use of assets and allowances for environmental and threat conditions during Obscure phase of a hasty breach.

ELO4.1: Indicate placement of smoke rounds.

ELO4.2: Indicate timing of smoke rounds.

ELO4.3: Indicate use of mortar before artillery for smoke rounds.

LO5: Indicate proper use of assets and allowances for environmental and threat conditions during Secure phase of a hasty breach.

ELO5.1: Indicate continuing weapon employment against threats.
ELO5.2: Indicate engineering unit movements to position.
ELO5.3: Indicate movement of tank platoon to secure position.

LO6: Indicate proper use of assets and allowances for environmental and threat conditions during Reduce phase of a hasty breach.

ELO6.1: Indicate engineering unit selection of breaching asset.
ELO6.2: Indicate breaching location and creation of breach.

LO7: Demonstrate using the MAGTF XXI simulation the proper implementation of tactics and assets during the Suppress, Obscure, Secure, and Reduce phases of a hasty breach

L08: Manage workload during performance of MAGTF XXI scenario.

ELO8.1: Manage workload during performance of MAGTF XXI scenario.

Appendix B: Metrics

Element	Rule #	Behaviors
Suppress	1	Tank Platoon 1 OR Tank Platoon 2 OR Mech Platoon starts movement within 15 seconds
Suppress	2	(Tank Platoon 1 OR Tank Platoon 2) AND Mech Platoon are moving within 45 seconds
Suppress	3	Limit time for all assets within inaccessible areas** to < 20 cumulative seconds per asset
Suppress	4	(Tank Platoon 1 OR Tank Platoon 2) AND Mech1 complete movement within 360 seconds
Suppress	5	(Tank Platoon 1 OR Tank Platoon 2) AND Mech Platoon complete movement within 90 seconds of each other
Suppress	6	(Tank Platoon 1 OR Tank Platoon 2) AND Mech Platoon achieve proper orientation within +/- 30 degrees
Suppress	7	Issue at least 1 movement command for each of the three manuever units (Tank Platoon 1, Tank Platoon 2, and Mech) and the three engineering units
Suppress	8	Do not issue any movement command for any of the fire support units (mortar1, mortar2 and Artillery Battery)
Suppress	9	(Tank Platoon 1 OR Tank Platoon 2) and Mech Platoon arrive and remain for at least 5 minutes in different optimal SBF areas: See "Additional Metrics.xls" file for details
Suppress	10	(Tank Platoon 1 OR Tank Platoon 2) OR Mech Platoon arrive and remain for at least 5 minutes in different less optimal SBF areas: See "Additional Metrics.xls" file for details
Suppress	11	Limit (Tank Platoon 1 OR Tank Platoon 2) AND Mech Platoon time outside of correct SBF route* to <10 seconds
Suppress	12	Employ DPICM HE against tanks AND IFVs AND against infantry once all IFV/tanks are destroyed
Suppress	13	Employ MORTAR HE against infantry only
Suppress	14	(Tank Platoon 1 OR Tank Platoon 2) and Mech Platoon stay in SBF position through creation of breach
Obscure	15	Fire smoke when closest asset is within 2 min of entering BREACH 1 or BREACH 2
Obscure	16	Fire smoke at intervals not less than 3 minutes and not more than 5 minutes
Obscure	17	Place rounds within optimal area Smoke 1 for the scenario*
Obscure	18	Place rounds within less optimal area Smoke 2 for the scenario*
Obscure	19	Place rounds within least optimal area Smoke 3 for the scenario*
Obscure	20	Fire smoke from mortar units FIRST - (before using artillery units)
Obscure	21	Fire smoke from artillery battery
Secure	22	Employ (DPICM AND/OR HE) at a rate of 3 rounds every 2 minutes as long as there is a live enemy unit
Secure	23	Position Securing Tank and Engineering unit (one of 3) in Breach1 until breach is complete
Secure	24	Securing tank (Tank Platoon 1 OR Tank Platoon 2) reach the breach area (Breach1 OR Breach2) within 1 minute prior to breaching engineering units (any one of 3 units)
Reduce	25	Conduct breach within BREACH 1 (best). (Breaching nit is in breach 1 area)
Reduce	26	Breach with "M1A1 with mine plow" first
Reduce	27	One breach completed
Reduce	28	Reduce phase (one breach) completed within 15 min (Scenarios 1, 2W, 3, 4W, 5, 6W), 8 min (Scenarios 2W_2, 3_2, 4W_2, 5_2) and 5 min (Scenarios 2W_3 and 5_3) elapsed scenario time