A CONSTRUCTIVIST APPROACH TO DISTANCE LEARNING FOR COUNTERTERRORIST INTELLIGENCE ANALYSIS

Tamitha Carpenter, Daniel Fu, Phillip Michalak, Laurie Spencer
Stottler Henke Associates, Inc.
San Mateo, California
tamitha@shai-seattle.com, fu@shai.com, michalak@shai.com, lauries@shai-seattle.com,

Luciano Iorizzo
US Army Intelligence Center & Fort Huachuca
Fort Huachuca, Arizona
iorizzol@huachuca-emh1.army.mil

Topic Outline

The uncertainty of unconventional threats confounds the Military Intelligence community. Changing operational requirements and constraints exacerbate the capability to prepare soldiers for unexpected situations. Thus, complexity of a training requirement increases while conditions truncate the means to provide training. Training must also prepare a soldier to accept the challenge of the unknown rather than become overwhelmed. One way to use limited resources to develop a competency is to combine distance learning (DL) techniques with adaptive learning strategies. This paper will describe one approach to optimizing the use of DL techniques with Constructivist learning theory to train complex domains. We present a courseware system that trains students in the area of "Intelligence for Combatting Terrorism" (ICT), and an associated authoring tool. This authoring tool supports course development both at the subject matter level and at the pedagogical level, allowing the author to create and update the course material and to develop and refine an instructional process that adapts to the learning styles of each individual student. We conclude the paper with a description of our generic programming framework for creating customized tutor authoring tools.

Biographical Sketches

Tamitha Carpenter is a project manager and software engineer with Stottler Henke Associates, Inc. (SHAI). Her specialization is in Artificial Intelligence (AI) and natural language understanding. While at SHAI, Dr. Carpenter has applied AI techniques to a broad range of domains, including computer security, automated document generation, web-based collaboration, and intelligent tutoring systems (ITSs). Dr. Carpenter holds a Ph.D. in computer science from Brandeis University.

Daniel Fu is a project manager and software engineer with SHAI. He divides his time between the management, oversight, and development of distance learning software, and the development of Artificial Intelligence control techniques for simulation entities. Dr. Fu holds a Ph.D. in computer science from the University of Chicago.

Phillip Michalak is a software engineer with SHAI. While at SHAI, he has worked on the development and maintenance of an ITS for naval officers that utilizes intelligent entities in a free-play simulation. He currently splits his time between development of distance learning software and an ITS for literacy enhancement. Mr. Michalak holds a B.S. in Computer Science from Carnegie Mellon University.

Laurie Spencer is a project manager and software engineer with SHAI. While at SHAI, she has worked on the development of a decision support system, and the management, design, and development of a data-mining system that focused on intelligent user interaction and usability, as well as the effort described in this paper. Ms. Spencer is currently working on a M.S. in Computer Science at the University of Washington and holds a B.S. from the Georgia Institute of Technology.

Mr. Iorizzo leads the Distance Learning Office, U.S. Army Military Intelligence Center, Fort Huachuca, AZ. The Office produces products based on Constructivist learning theory delivered via Internet. He served as Technical Advisor for the U.S. Army Armor School, Fort Knox, Ky., for five years. His foundational work in instructional design and prototype development for the Armor Captains Career Course culminated in a U.S. Distance Learning Association award to Fort Knox for best overall program, government category, in 1999. Mr. Iorizzo holds a master's degree in Instructional Design, Development, and Evaluation from the School of Education at Syracuse University.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2005		**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A Constructivist Approach to Distance Learning for Counterterrorist Intelligence Analysis	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **US Army Intelligence Center & Fort Huachuca,Fort Huachuca,AZ,85613**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**10**	

A CONSTRUCTIVIST APPROACH TO DISTANCE LEARNING FOR COUNTERTERRORIST INTELLIGENCE ANALYSIS

Tamitha Carpenter, Daniel Fu, Phillip Michalak, Laurie Spencer
Stottler Henke Associates, Inc.
San Mateo, California
tamitha@shai-seattle.com, fu@shai.com, michalak@shai.com, lauries@shai-seattle.com,

Luciano Iorizzo
US Army Intelligence Center & Fort Huachuca
Fort Huachuca, Arizona
iorizzol@huachuca-emh1.army.mil

INTRODUCTION

This paper postulates that soldiers can adapt to unknown situations when trained in authentic environments based on instructional design strategies derived from *Constructivist* learning theory [Jonassen et al., 1999]. We describe a *distance learning (DL)* instructional system that will train soldiers in counterterrorist intelligence analysis. We provide them with data similar in format and content to information collected in the field, and give them the opportunity to interact with the tools used by counterterrorist intelligence analysts. The system allows the user to construct an analysis directly from the information, and imposes no constraints concerning the order in which the analysis is constructed. The system customizes interaction based on each user's proficiency. In addition, the described system provides *authoring*[1] capability that allows the course designer to update and add course material easily, as well as to develop and refine the system's process of adapting to the learning styles of each individual student.

Background

Events in the Military Intelligence community establish that our soldiers need flexibility to adapt to unfamiliar situations. One way to provide flexibility is through extensive hands-on experience. Unfortunately, traditional methods for providing hands-on experience are prohibitively expensive in terms of time and training resources - especially in the current atmosphere of cost reduction and budgeting. At issue, then, is how to prepare a soldier to adapt to the unknown within the constraints of military budgeting. Distance Learning using a Constructivist paradigm appears to be a good match to attain competency level outcomes while reducing the overall cost of providing comprehensive training.

The decision to use constructivist learning theory is in response to the requirement to prepare the Military Intelligence (MI) community for unknown threats. In the past, the MI community could rely on indicators and templates to predict outcomes when faced with a conventional, known enemy. Disappointingly, traditional techniques are of limited use when the dynamics of a global, information age culture make ill-structured domains the primary target of MI operations. A constructivist approach encourages a learner to develop their own problem-solving model to apply generally rather than to rely on specific procedures for particular events.

The primary challenge in combining DL and Constructivism lies in the apparent incompatibility between the learning environment and strategy. The nature of Constructivism – a method of reflecting and interacting with peers – makes it fit comfortably into the social environment of a classroom, while DL and, indeed, most Internet behavior are intrinsically solitary activities in an artificial environment. However, some research suggests that Virtual Learning Environments (VLEs) can accommodate the individual and managed groups of learners [Britain, Liber, 1999]. Britain and Liber suggest that VLEs are well suited to a resource-based constructivist approach to learning. We attempt to make these viewpoints work together by establishing a VLE using virtual reality, multi-user domains, and automated advice mechanisms, resulting in a tutoring system that is engaging, effective, and motivating.

In this paper, we describe an approach to DL that allows for sophisticated, flexible training within complex domains. By following the Constructivist methodology, our training system supports a student's need to master a variety of competencies and to apply them in unique situations and in different sequences. In addition, our training system is attuned to the trainee's background and needs, motivating the learner to develop an accurate and thorough understanding of the

[1] For a treatment on general ITS authoring, see [Stottler & Ramachandran, 1999].

subject matter, and then effectively verifying the correctness of their understanding and attending to inaccuracies.

Emerging research indicates that VLEs ought to transcend cognitive and individual learning style considerations [Martinez and Bunderson, 2001, in press]. Martinez and Bunderson indicate that conative, affective, and social aspects are a dominate learning influence and suggest a Learning Orientation Model that accounts for an individual's proclivity to take risk and control of their own learning. Our training system is well suited for autonomous learners. A key consideration is how to provide an authentic learning experience for those users who prefer a semi-structured or highly structured learning environment.

The training system we present consists of two innovative products. The first is a general course creation framework that supports the creation of Constructivist DL courseware in a wide variety of areas. The second product is a specific tutoring and scenario authoring system. The Intelligence in Combating Terrorism (ICT) courseware was built using this general course creation framework. This tutoring system gives the students extensive, hands-on training in the analysis of raw intelligence information related to investigating terrorist organizations and installation threat assessments.

THE "INTELLIGENCE IN COMBATING TERRORISM" (ICT) COURSE

The ICT tutor is based on material from the Army's course title "Intelligence in Combating Terrorism". In this course, students learn how to identify and assess terrorist activity through the analysis of information (or "raw intelligence"). Intelligence to combat terrorism is derived by compiling and analyzing open source information, criminal information sources, local information, and government intelligence. The role of intelligence in combating terrorism is to identify the threat and provide timely threat intelligence to the commander. One way to keep the term ***Intelligence*** in perspective is:

Information + Analysis = Intelligence

Within the ICT course, students are taught to analyze information through the use of several analysis tools: association matrices, activity matrices, time event charts, and link diagrams. These tools are designed to aid the intelligence analyst in pinpointing the essential content contained within a large body of unformatted messages containing a wide array of raw intelligence. Gaining proficiency with these tools is essential for any personnel who will be required to analyze raw intelligence in order to reduce the vulnerability of personnel, installations, and units to terrorist attack.

We describe the purpose of each of these tools below:

Association Matrix – The association matrix is where the analyst lists all the people (and their aliases) thought to be involved in or associated with a particular terrorist organization. In addition, the association matrix provides a compact representation for specifying the pairwise association between the people in the matrix. (That is, each person receives notations specifying with which of the other people in the matrix he or she has had or is thought to have had contact.)

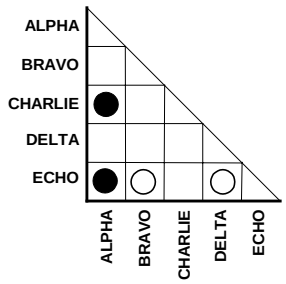

Figure 1. Association Matrix

Activity Matrix – The activity matrix is where the analyst lists all activities in which people from the association matrix have been involved. Activities include events, groups, places, governments, and, indeed, anything significant that is not a person. In addition, the activity matrix provides a compact representation for specifying which people were involved or are thought to have been involved in which activities.

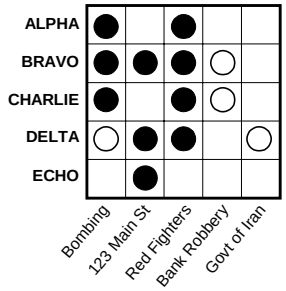

Figure 2. Activities Matrix

Time Event Chart – The time event chart allows the analyst to construct a timeline of significant events associated with a particular terrorist organization. Each entry in the time event chart includes a summary of the

event and the date on which it occurred. Time events are drawn in ascending chronological order.

Link Diagram – The analyst constructs the link diagram in order to pull together information contained in the association and activity matrices and produce a pictorial representation of the likely structure of a terrorist organization. People and their associations are represented as linked circles, and each activity is then drawn as a container surrounding all people involved. The resulting diagram makes it possible to form hypotheses about which people form the leadership and core membership of the organization. Inclusion or exclusion of particular activities and people reveal patterns at a finer granularity and allows the analyst to focus on smaller groups of people and/or activities.

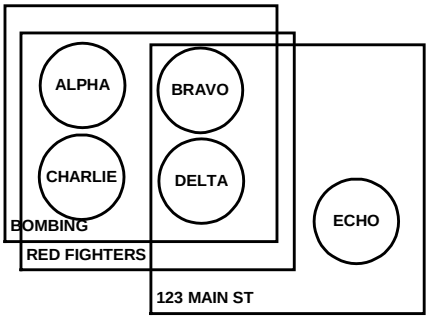

Figure 3. Link Diagram

THE ICT TUTOR

The purpose of the ICT tutor is to train the student in the analysis of raw intelligence leading to a compact summary of a terrorist operation and an assessment of the current level of threat. The tutor uses Constructivist learning theory by supporting adaptive learning, modeling, intentional activity, and rich scenario contexts. The tutor immediately places the learner in a 'real-world' environment that allows the student to learn in context and apply what they have learned. It is this contextual experience of knowledge acquisition in an authentic environment that facilitates the learner to create their own constructs that can be applied to new, unfamiliar situations.

The Tutor presents the student with a problem, provides the student with the tools to solve the problem, and offers customized suggestions as a resource to solve the problem. The student produces a solution, receives consequences based on their solution, and is guided to suggested areas for review before beginning another scenario. The student even accesses baseline knowledge within the context of the scenario.

The ICT tutor has the following characteristics:

Students learn at their own pace, and interaction with the tutor is completely unconstrained – This characteristic is a direct result of the Constructivist approach used by the ICT tutor. In Constructivist learning, the role of the instructor is to provide guidance while the student learns the material by exploring and manipulating the environment. In the case of the ICT tutor, the environment consists of a set of "messages" containing the available raw intelligence, and empty copies of the analysis tools (i.e., association matrix, activity matrix, time event chart, and link diagram).

The student is free to read (or not read) the messages and access the available help resources, including the textbook and standard system help. The student manipulates the environment by linking parts of a particular message ("message fragments") to the analysis tools. For example, if the student discovers a reference to "Fred" in a message that links Fred to the terrorist organization under investigation, the student can choose to add Fred to the association and activity matrix. The ICT tutor stores a history of the links the student made between messages and analysis elements. At a later time, the student can revisit what they have previously discovered, thus possibly gaining a greater understanding of material explored earlier. These links are an explicit representation of the thought process a skilled analyst would use. The intentional act of creating these links compels the student to enumerate the steps that become internalized.

The ability to interact freely with the environment allows students to evaluate sequences and strategies as they engage in the learning process. Through this interaction, students determine the best use of the tools in a way that is meaningful to them. The system's ability to cope with the Constructivist paradigm of experiential learning heightens its value as a Distance Learning tool.

Students can train with or without a human instructor – The ICT tutor provides all of the information that a student needs in order to learn how to analyze counterterrorist information. Help sources include an online version of the textbook used in the resident ICT course, standard application help, and "hints" motivated by the tutor's model of the student and their activity. These customized hints also account for learners who prefer a semi-structured learning experience. In addition, when an instructor is available, even remotely, the ICT system provides access to tools that allow the instructor to evaluate the student's progress and, when necessary, provide feedback. This is

especially important to learners who suffer anxiety when highly structured experiences are absent.

A persistent, adaptive student model ensures appropriate help – As the student uses the ICT tutor, a model of progress and exhibited abilities is constantly updated. While the student is working on the course, the student model is used to stimulate the automated help system. Further, it also provides progress snapshots of individual students or an aggregate view of student trouble areas for an instructor.

The student model is persistent across all of a student's sessions with the ICT tutor, allowing the tutor to recognize the level of the student's skill at all times. Thus, the ICT tutor provides novice students with more help compared to an advanced student during the training process. This customized feedback is attuned to the student's needs and background, contributing to the value of the ICT tutor as a distance learning tool. This type of feedback is consistent with the role of the instructor in the Constructivist paradigm.

An integrated *principle hierarchy* – The ICT tutor's help system and student model predicate from a "principle hierarchy" that enumerates the competencies a student needs to master. The student model is updated as the student demonstrates mastery of a particular competency. The ICT Tutor intrinsically ties automated help to the principle hierarchy. Therefore, the Tutor will stop generating help related to mastered competencies, thus customizing the tutor as the student progresses. In addition, principles located beneath the mastered competency in the hierarchy can be assumed, and the help system will adapt accordingly.

The author of the scenario considers each principle carefully and quantifies the actions and analyses that demonstrate comprehension and mastery of that principle. These assessments can take the form of a percentage of possible actions completed, or the successful analysis of particular parts of the data. In a similar fashion the author specifies the point at which the system should generate hints for the student. The hint conditions take the form of a set of actions and/or analyses that have been completed and a set of actions and/or analyses that have not yet occurred. The hint will offer the student one or more resources pertaining to the items that have not occurred.

This principle hierarchy enumerates competencies in two areas. The first area contains representations of competencies related to the use of the actual ICT Tutor. By modeling the student's mastery of the ICT Tutor, automated help can be provided to minimize the learning curve associated with using a new piece of software, and maximize the student's ability to focus on learning the ICT material.

The second area contains representations of competency associated with the analysis of raw intelligence. The goal of the ICT Tutor is for the student to achieve proficiency and understanding of the principles enumerated in the second area.

Local execution, centralized student model – The ICT tutor is designed to maximize the benefits of distance learning, while avoiding the pitfalls of over-dependence on network connectivity. Therefore, the student installs the ICT tutor as a local application on their own PC. The Tutor uses network connectivity when the student: 1) begins a session, 2) explicitly saves the session to the server, 3) ends the session, 4) submits results for evaluation, or 5) collaborates with peers or an instructor.

In addition to maintaining a minimal reliance on network connectivity, having the ICT tutor run locally significantly improves application startup speed. Storing the student model centrally allows students to complete their work regardless of their location, supporting the use of home computers, computers at the office, and computer labs shared by multiple students.

The ICT Tutor Interface

The student interface for the ICT tutor is designed to maximize the student's ability to explore, manipulate, and understand the process of developing an analysis. The interface is easily configurable, allowing the student to see all portions of the tutor in a single, easy to understand window (see Figure 4). This window is divided into several independent panes, each of which may be hidden with a single mouse click when it is not needed, and resized with a single mouse operation to suit an individual student's preferences. The screen snapshot in Figure 4 shows the ICT tutor window with the panes for the Message List, the high-level Scenario Outline, and the Analysis Tools visible. The two panes hidden in this snapshot are the Link Diagram pane and the Hints pane.

Message List – When a student begins working with a new scenario, the only scenario-specific information that is available is the set of messages containing all of the raw intelligence that has been "collected" on a particular terrorist situation. The Message List is split into two resizable sub-panes. The first is simply a list of the messages (i.e., raw intelligence) contained in the scenario. Clicking on one of the messages in the list causes the text of the message to appear in the second sub-pane.

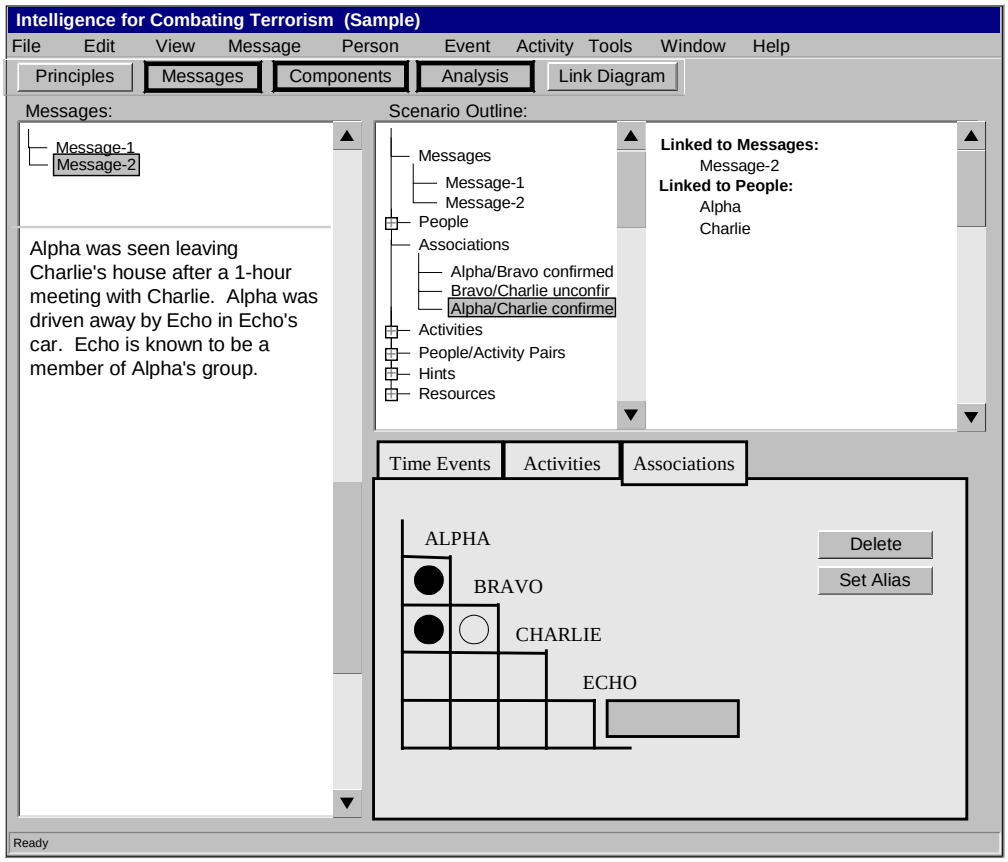

Figure 4. A screen snapshot of the ICT tutor (training system).

The text sub-pane is where a majority of the student's efforts are focused. In order to build an analysis of the scenario, the student selects pertinent text fragments from this sub-pane, and links them to the analysis tools.

Scenario Outline – The Scenario Outline provides the student with a birds-eye view of the scenario. Similar to the Message List, the Scenario Outline is split into two resizable sub-panes. The left-hand pane displays a hierarchical list of all elements of the scenario, including messages, people, associations, activities, time events, hints, and resources (i.e., the textbook, web pages, etc.). Clicking on any of these elements causes the right-hand sub-pane to display all links associated with the selected element. For example, in Figure 4, the association "Alpha/Charlie confirmed" is linked to Message-2, since that is where the student found the evidence for this association, and to the people Alpha and Charlie.

In addition to providing the student with a summary of their activity so far, the Scenario Outline provides the student with an expedited means for navigating the scenario. Double clicking on any element in either sub-pane of the Scenario Outline causes that element to be brought into focus within the appropriate pane. For example, if the student double-clicks on the name of a message, that message becomes selected within the Message List (causing the Message List pane to become unhidden if necessary). One major benefit of this navigation system is that it allows the student to revisit past decisions, which can lead to additional discoveries and refinements.

Analysis Tools – The Analysis Tools pane provides the student with tabbed sub-panes containing the Association Matrix, the Activities Matrix, and the Time Event Chart. Students make entries in each of these tools in two ways. First, as mentioned earlier, the student can select a text fragment from a particular message and link it to a new entry in one of these tools. Second, the student can interact with the tools manually (i.e., enter names of people, activities, or time events; mark associations and activity/person pairs as confirmed, unconfirmed, or unknown.)

Students are encouraged to associate message fragments throughout the tools which automates a point

of reference for the student, allowing them to revisit their decisions at a later date. However, following the Constructivist paradigm, students are free to interact with the tools as they see fit. In this instance, the student is queried to identify which fragment they believe salient.

The **Link Diagram** is created from the entries that the student has made in the Association and Activities Matrices. It serves the purpose of allowing the student analyst to summarize the information discovered and obtain a complete picture of a particular terrorist organization's structure. Although the student may construct the Link Diagram at any time, the most benefit is achieved when the other analysis tools are close to complete, since incomplete information will yield an imperfect view of the organization.

Hints – Throughout the student's interaction with the ICT tutor, a model of the student's activity is maintained, and automated hints are provided as necessary. These automated hints provide the user with guidance, much as a human instructor would, while encouraging the student to set their own pace and organize their own activity so it is most suitable to their own style of learning.

The Hints pane provides the student with a current list of the automatic hints that the Tutor decides are appropriate for the student's level of progress. If a new hint is generated while the Hints pane is hidden, an indicator appears in the status bar at the bottom of the window.

The hints provide the student with small pieces of context-relevant information designed to help the user make progress with the session, but without "giving away" the solution. The types of hints provided fall into three categories:

1. Application hints are provided to help the user (both student and author) with the mechanics of interacting with the ICT application. These hints include hyperlinks into the Getting Started tutorial and the online help.
2. ICT general hints are provided when the student is exhibiting unfamiliarity with the ICT material. For example, if the student has not been identifying all types of activities, a hint clarifying the definition of "activity" will be displayed: "An activity can be a place or an organization, as well as an action." These hints include hyperlinks into the ICT textbook, so the student can get more detailed information.
3. Scenario specific hints are provided to prevent the student from getting frustrated. These include hints created automatically by the ICT tutor by comparing the student's progress with hint conditions in the principle hierarcy, and hints designed explicitly by the instructor. As a courseware author, the instructor is able to create a substantial level of detail within the hints hierarchy. Thus, individuals reluctant to take risk structuring their learning experience are more likely to find a comfort level while still immersed in an authentic environment.

Note that the generation of hints is contingent on the student model. Thus, hints do not appear on topics where the student has exhibited proficiency.

Student Assessment

When the student decides their analysis is complete, they move on to the "assessment" part of the course, which consists of two parts. This first part requires the student to write an "Installation Threat Assessment", in which the student summarizes their analysis of the scenario. The second part consists of multiple choice questions pertaining to the student's Installation Threat Assessment. These questions are written based on the student's "end product", not on general course material. Thus, questions will be of the form "What were your findings on perimeter security?" as opposed to "What constitutes a secure perimeter?"

During this assessment, the students have the opportunity to reflect on the "big picture" of the scenario. The Threat Assessment that each student completes is based on the student's investigation of the intelligence reports in the scenario. In using their investigative results as the basis for their analysis, students are compelled to reflect on the investigative job they performed. Was the membership of the relevant terrorist organizations sufficiently well defined? Was the data collected enough to indicate trends or preferences among terrorist groups? Such reflective questions help impress the importance of each stage in intelligence analysis.

AUTHORING

When referring to courseware authoring, we distinguish between subject matter expertise and pedagogy. The authoring system described here allows a domain expert to author materials like courseware and scenarios. Then, a pedagogical expert can devise instructional methodologies that allow the system to adapt to the

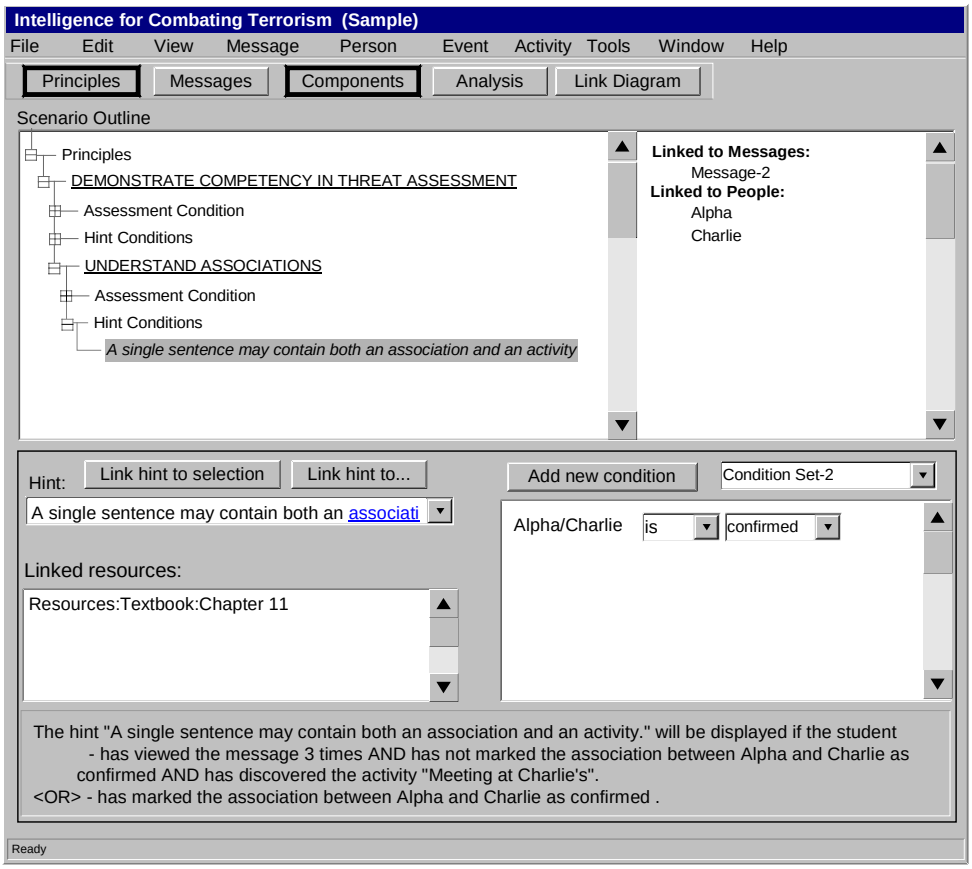

Figure 5. A screen snapshot of the ICT scenario authoring system, showing the hint authoring mechanism. Note that this hint has two condition sets (i.e., disjunctive conditions), the second of which is displayed. The full meaning of the conditions is given at the bottom of the Hint Authoring pane.

individual student, as well as provide criteria to determine when a student has learned a concept. With these methods, an authoring tool can produce a tutoring system that customizes instruction based on the student's background and performance over the duration of the course.

The ICT Scenario Authoring System

The ICT tutor contains, at its core, a robust functionality for allowing students to interact with messages and analysis tools, to access help, and to automatically receive hints when necessary. This functionality is completely independent from the content of any given ICT scenario. Thus, we have been able to provide the course author a powerful authoring system for creating new scenarios and editing currently available scenarios. This flexibility allows the author to define the varied situations necessary for successful mastery of concepts, thus contributing to the value of the ICT tutor as a distance learning tool.

- Scenarios can be created for different parts of the world (e.g., Middle East, South America), making the ICT tutor appropriate as a tool for acquainting an intelligence analyst with the dynamics of their new assignments.
- A particular scenario can easily be updated so that it reflects real world activities.
- Scenarios that students find particularly confusing can easily be clarified, or more comprehensive automated help can be added.

The ICT scenario authoring tool provides authoring capability for all aspects of the ICT tutor. Messages can be added, edited, or imported. The scenario solution (i.e., the correct content of the analysis tools) can be edited to allow the tutor to evaluate the student's results. Help resources can be provided. Suites of threat assessment questions can be created.

The most complex aspect of the ICT scenario authoring tool is its provision for the author to create the Principle Hierarchy and custom Hint Conditions (see Figure 5).

The Principle Hierarchy Editor allows the author to create an unrestricted hierarchy, which includes at each node a condition to determine whether the student has mastered the competency, and conditions for displaying hints. The assessment conditions, taken as a whole, can be regarded as the evaluation of the student model. For each competency in a scenario, the principles enumerate the concepts that the student is expected to learn, and the assessment conditions are an explicit measurement of what the student has learned. When new scenarios are created, the author will customize the scenario level hints and assessment conditions, using the higher level concepts as a guide. The ICT principle authoring mechanism provides a convenient interface for the author to create the text of the principles and hints, embed hyperlinks from the hint to a message or resource, and create conjunctive and disjunctive conditions based on the student's activity, such as:

- Whether the student has displayed proficiency at a particular competency (as specified in the principle hierarchy). The assessment of this competency can be quantified by the performance of certain activities or by a set number of completed activities. For example, the skill of identifying an activity may be considered learned when a student has integrated any three activities, or when a student has created a particular activity within the scenario.
- How many times the student has viewed a particular message.
- The status (i.e., confirmed, unconfirmed, unknown) of an association or an activity/person pair.
- The discovery (or lack thereof) of a particular person or activity.

Generic Course Creation Framework

One of the known difficulties with building robust tutoring systems is the problem that a great deal of the programming is domain specific. For example, in the ICT Tutor and Authoring System, the interaction between messages and the analysis tools, as well as the content and construction of the analysis tools, is specific to ICT. This is because the user interface is tailored to assist a specific type of analysis. In general, support for graphical interaction that is specific to a particular topic must be provided at the programming level.

However, much of the framework used by the ICT tutor is generic across topics. As part of the construction of the ICT tutor, we have developed a "Generic Course Creation Framework." This framework is not a course authoring tool, but instead provides a *programming* framework for creating customized tutor authoring tools.

The programming framework consists of the following:

Principle Hierarchy support – Much of the difficulty in creating a good tutoring system is in hooking the principle hierarchy into the tutoring system. We have developed a design for the principle hierarchy that supports attachment between the hierarchy and the customized course content. In fact, upon completion of the first scenario, the general hierarchy will be finished. Most additions by subsequent scenario authors will be performed at the lower levels.

Automated Hint support – Another area of course creation that can be particularly difficult is the creation of customizable automated hints. We created a framework supporting the functionality of the hint authoring system because students vary in their desire to structure their own learning. This necessitated the capability to link hints directly into the custom principle hierarchy, develop hints with conditions predicated on specific course content, and embed hyperlinks to a variety of resources and course content into individual hints.

The Scenario Outline – The programming framework includes the functionality of the Scenario Outline accessed in the ICT Tutor interface. As with the other portions of the programming framework, the Scenario Outline is customizable to any topic that can be represented as a hierarchy of concepts.

Centralized Student Model support – The framework includes functionality for connectivity with a centralized JDBC[2] compliant database, and stores state information for each student model. This allows a student to work from any PC at any location. In addition, it provides the core functionality that allows the author to view the aggregate student model. This in turn allows the author to evaluate the success of a given scenario and determine whether the scenario or the principle hierarchy needs modifications.

Assessment Questions – The framework includes functionality for authoring arbitrary multiple-choice questions. This allows our self-assessment technique for Installation Threat Assessment solutions to be re-purposed for procedural and general knowledge testing

[2] "Java DataBase Connectivity" - a Java API that enables Java programs to execute SQL statements, thus allowing Java programs to interact with any SQL-compliant database.

if one ever had a need to measure comprehension independent of context.

These five framework elements represent a significant portion of courseware development effort. Using this framework will allow future courseware developers to create complex Constructivist DL learning systems and authoring tools in a substantially shorter timeframe.

SUMMARY

When developing Distance Learning (DL) systems, it is crucial to provide students with the correct balance of interactivity, self-direction, and guidance. By marrying the Constructivist paradigm to our Distance Learning system, we have created a learning environment that does not just present material to the student, but actually invites the student to develop their own learning experience. This leads to a deeper understanding that can be applied to unfamiliar situations.

This paper has described two products that support the combination of DL and Constructivism. The first is a general course creation framework that supports the formation of Constructivist DL courseware in a wide variety of areas. This framework has the potential to substantially impact the future of DL tools by supporting development of Constructivist courseware in a wide variety of domains.

The second product is a specific tutoring system built using the general course creation framework to teach the Intelligence in Combating Terrorism (ICT) course. This tutoring system gives the students extensive, hands-on training in the analysis of raw intelligence related to investigating terrorist organizations and conducting Installation Threat Assessments. The ICT Tutor immerses each student in an authentic environment, allowing the student to control the pace and the order in which they learn the material. The ICT Tutor provides a pedagogy that customizes interaction based on each user's level of competency and learning preferences. This is a substantial improvement over standard multimedia courseware, which takes a "one size fits all" approach to student interaction and feedback.

The ICT tutor contains, at its core, robust functionality allowing students to interact with messages and analysis tools, to access help, and to receive hints automatically based on deliberate student attributes. This functionality is completely independent from the content of any given ICT scenario. Thus, we have been able to provide the course authors a powerful authoring system for creating new scenarios and editing currently available scenarios. In addition, the authoring system gives the author the ability to edit and augment the principle hierarchy, thus giving the author control over the courseware's adaptation to individual students' needs and learning styles. This flexibility allows the author to define the varied situations necessary for successful mastery of competencies, thus contributing to the value of the ICT Tutor as a distance learning tool.

ACKNOWLEDGEMENTS

We want to recognize Ms. Helen Remily for her contributions in technical discussions and project management and Ms. Teri Jackson for orchestrating significant financial support for this project from the Air Force Research Laboratory.

REFERENCES

Britain, Sandy & Liber, Oleg (1999). A Framework for Pedagogical Evaluation of Virtual Learning Environments. *Joint Information Systems Committee of the Higher Education Funding Councils; JISC Technology Applications Programme,* Report: 41.

Jonassen, D.H., Peck, K.L. & Wilson, B. G. (1999). *Learning with technology: A Constructivist perspective.* New York, NY: MacMillan.

Martinez, M. & Bunderson, C. (January, 2001, in press). Building Interactive Web Learning Environments to Match and Support Individual Learning Differences. *Journal of Interactive Learning Research*, 11(2).

Stottler, R. & Ramachandran, S. (1999). A Case-Based Reasoning Approach to Internet Intelligent Tutoring Systems (ITS) and ITS Authoring. In *Proceedings of the Twelfth International Florida Artificial Intelligence Research Symposium (FLAIRS) Conference Special Track on Intelligent Tutoring Systems.*