INTERACTIVE GRAPHICAL TIMELINES AS COLLABORATIVE SCENARIO MANAGEMENT TOOLS

A Thesis

by

AUSTIN CHRISTOPHER RIDDLE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 2008

Major Subject: Computer Science

INTERACTIVE GRAPHICAL TIMELINES AS COLLABORATIVE SCENARIO MANAGEMENT TOOLS

A Thesis

by

AUSTIN CHRISTOPHER RIDDLE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Frank M. Shipman III
Committee Members,	Richard Furuta
	Lauren D. Cifuentes
Head of Department,	Valerie E. Taylor

May 2008

Major Subject: Computer Science

ABSTRACT

Interactive Graphical Timelines as

Collaborative Scenario Management Tools. (May 2008)

Austin Christopher Riddle, B.S., Texas A&M University

Chair of Advisory Committee: Dr. Frank Shipman

Training emergency response decision makers using live, virtual and/or constructive simulations can be highly complex since certain situations can generate stimulus-response cycles that depend significantly on unpredictable human judgments. In particular, effective training scenarios require a combination of content contributed via pre-authored scripts and content generated dynamically during the training exercise. Large-scale exercises require multiple domain experts contributing oversight and content to the scenario as it proceeds. Such real-time adaptation requires situational and group awareness based on an understanding of pre-scripted materials and the adaptations of others. This thesis describes the evolution and evaluation of a collaborative graphical timeline system, called the Scenario Timeline System (STS), which facilitates asynchronous and synchronous collaborative timeline management, and its application in large-scale, computer-supported emergency response training exercises.

DEDICATION

—To Jesus, my God and my All, *ad maiorem gloriam Tuam*

ACKNOWLEDGEMENTS

I would like to thank my committee chair, Dr. Frank Shipman for his generous guidance and patience, along with my committee members Dr. Richard Furuta and Dr. Lauren Cifuentes.

Special thanks go to Dr. James Wall and the Simulation Group at the TEES Texas Center for Applied Technology for their encouragement, ideas, support and facilitation of this research and other projects that impact lives.

I would also like to thank the members of the TEEX National Emergency Response and Rescue Training Center for their valuable feedback and support in the continued evolution of emergency response training and systems.

My wife and children also deserve special recognition for the sacrifices they have made throughout my studies.

Lastly, I would like to thank my mother and father for their love and teaching me the meaning of honor, integrity, and perseverance.

NOMENCLATURE

STS	Scenario Timeline System
TEES	Texas Engineering Experiment Station
TCAT	Texas Center for Applied Technology
TEEX	Texas Engineering Extension Service
AAR	After Action Review
NIMS	National Incident Management System
ICP	Incident Command Post
EOC	Emergency Operations Center
MACC	Multi-Agency Coordination Center
EOTCS	Emergency Operations Training Center Simulation
EM*ES	Emergency Management Exercise System
AEAS	Automated Exercise and Assessment System
EDMSIM	Emergency and Disaster Management Simulation
WMD	Weapons of Mass Destruction
CBRNE	Chemical, Biological, Radiological, Nuclear, and Explosive
CSCW	Computer Supported Collaborative Work
MSEL	Master Scenario Event Lists
TAF	Training Assessment Facility
EOTC	Emergency Operations Training Center

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Live simulations have been used in the emergency response discipline for decades. These simulations have taken the form of tactical and/or decision-making exercises to help train individuals and groups to better respond to emergencies [Corley and Lejerskar 2003; Frishberg 2005; Jain and McLean 2003]. Being characteristically scenario-based and event-driven, they are managed by a group of individuals called *controllers*. These controllers guide and assess the flow of the exercise based on a stimulus-response cycle with the students. The format could range from a series of question and answer sessions to a very realistic depiction or performance of a scenario. One example might be, "A 911 call was received about a building on fire. What would you do?" The scenario would then unfold as a collaborative storytelling effort. Another example might involve a building that is actually on fire, with real people acting as victims. Irrespective of the format, the students formulate and execute response plans while playing the roles of emergency response decision-makers. Some controllers, who can be considered the *trainers*, contribute pre-scripted and actively-scripted plot elements, or stimuli, to the ongoing scenario. Others play the roles of anyone else with whom the students may need to interact. After the exercise is completed, another group of controllers, who are *evaluators*, analyze the outcome of the exercise to determine how well training objectives were met. This is accomplished using information captured in a log of the exercise in what is called the AAR, or After Action Review [Frishberg 2005].

This thesis follows the style of *Journal of the ACM.*

In the post-9/11 world, however, a greater emphasis has been placed on virtual and constructive simulations that train emergency response decision-makers for large-scale, mass-casualty incidents [Corley and Lejerskar 2003]. Virtual simulations involve human beings using simulated systems, while constructive simulations are generally model-based and can involve simulated people using simulated systems. These simulations can operate under varying degrees of fidelity. In particular, low fidelity virtual simulations, or those that represent a low degree of "real-world" detail in a computer model, can provide flexibility to trainers in this domain. This is important since controllers need to manage the flow of the exercise and are ultimately in charge of the quality of training. In practicality, the students and controllers simulate a real event or incident and manage virtual resources. In this case, the overall effort of the participants (students and controllers) and their usage of computers and other materials to mitigate the incident constitute the simulation. An *exercise* consists of the management of the simulated incident and an assessment of student performance. The software that assists in the execution of the simulation could more appropriately be considered an exercise management system as opposed to a computer simulation.

Large-scale, mass-casualty incidents are very complex because they require the management of hundreds and even thousands of resources, such as fire trucks and ambulances. Due to the sheer number of resources, this may require collaboration across jurisdictional boundaries [Frishberg 2005]. These incidents are necessarily managed with computers because of the cognitive and clerical challenges involved.

Training exercises related to such incidents require similar complexity in organizational and communication structure in order to provide students with an understanding of the difficulties involved. Thus, even in a low fidelity simulation, an increase in incident scale can equate to an increase in the number of controllers involved, which increases the number of computer software users. The reason for this increase is that since the simulation is low fidelity, there are still details not represented by the computer that must be fabricated by a human being.

In real-life situations, as an incident increases in scale, the number of responders also increases. At a certain point, hierarchical decision-making groups form to more effectively manage the incident. In the United States, the National Incident Management System (NIMS) is the process by which these groups are formed and the incident is managed [DHS 2004]. There are three groups that have been of particular interest to train for these kinds of incidents; the Incident Command Post (ICP), the Emergency Operations Center (EOC), and the Multi-Agency Coordination Center (MACC). The ICP is a group that manages the coordination and synchronization of the incident response effort. The EOC is a strategic group that collects and distributes information, and aids in the acquisition of resources to support one or more ICPs. The MACC is a multi-jurisdictional strategic group that collects and assimilates information from a high-level perspective, and aids in the acquisition of resources to support one or more EOCs.

Until recently, large-scale live, virtual and constructive emergency management simulations have been employed primarily in military venues. Attempts have been made in the past to transition these simulations for use in civilian contexts. However, in many cases distinct systems for training have instead emerged with their own unique strengths and weaknesses. Emergency management training systems like AEAS [ARA 2008], WebEOC [ESI 2008], EDMSIM [C4IC 2008], EM2000 [SDS 2008] and EM*ES from the Texas Center for Applied Technology (TCAT) were specifically designed for the information management needs of the NIMS groups previously mentioned. Training students to operate in these human infrastructures is not new, but since the scale of response increases for large, complex incidents and emergency management systems have become more sophisticated, additional tools are needed to allow controllers to better manage their training exercises.

PROBLEM

Much of the information managed by exercise controllers is logical, hierarchical and temporal in nature. In particular, certain information artifacts called *injects* are introduced into a scenario to stimulate a response from participants. Injects are logical and hierarchical in that they may depend upon previous information, and they are temporal in that they may only be appropriate at a given time and may occur in parallel. For example, a student may request a certain set of resources where some may already be deployed due to a pre-scripted inject. A training point may be that those resources not only cost money, but time, since they take time to drive and some may be delayed. Having a system maintain such time-relative information removes the need for controllers to remember what pre-scripted and actively contributed content is currently underway in the scenario. Existing training practices often manage such information using databases, tables, forms, and human intensive processes.

One way that controllers contribute active content is to play the roles of persons or organizations that are represented in a scenario, such as a mayor or police chief. They respond to the actions of the students, which could be through communications or decisions that alter the current situation. This requires significant competency on the part of controllers to ensure that responses to student actions are appropriate, realistic, and reconciled against what has occurred in the past so as to be non-contradictory. Additionally, transporting such competent controllers to a single location may be difficult, cost prohibitive or even impossible. This results in a need for tools that support a collaborative working environment.

In general, the scale and complexity of Weapons of Mass Destruction (WMD) or Chemical, Biological, Radiological, Nuclear, and Explosive (CBRNE) incidents makes the task of scenario design daunting. Frishberg [2005] mentions a rule of thumb for the amount of work required of exercise developers being 40 hours per 1 hour of exercise time. This process can involve multiple scenario authors working synchronously or asynchronously and is compounded by the fidelity level represented. Some emergency management training systems such as AEAS make scenario specification a software engineering task. Once the scenario is input, there is little flexibility for the controllers at runtime. However, as previously noted, there is a necessity for controllers to manage the flow of an exercise and be able to adjust to unpredictable student actions. WebEOC takes a step further by providing dynamic tables and an inject manager in the form of a table. However, the logical, hierarchical and temporal nature of injects can make a tabular representation difficult to use effectively. EDMSIM uses a timeline to display a summary of created injects, but lacks robust support for the management and creation of injects on a timeline. Given the intricacies of this domain and the deficiencies of existing systems, there is a need for tools that allow scenario developers and controllers to specify and manage robust scenarios that involve logical, hierarchical and temporal information and can stimulate students to accomplish their training objectives.

This thesis explores the evolution and effectiveness of the Scenario Timeline System (STS), a collaborative graphical timeline system, when used to aid scenario developers and exercise controllers in the specification, execution and understanding of large-scale training scenarios and the evaluation of student performance. The practical goal of STS is to facilitate the authoring and understanding of pre-scripted and dynamically contributed information about the past, present and prospective future of a scenario in order to lessen the cognitive load experienced by exercise controllers. The timelines can be integrated into a larger exercise management system, allowing information to be actively pushed to students in a timely and automated manner.

As an example, consider a situation where a controller finds himself being asked for a certain number of assets (e.g. fire trucks) by a student. Somehow, the controller must determine how to answer the request. The information required has much to do with the work of other simulation participants. Since there are a finite number of assets that can be manipulated across all controllers, the controller must keep a view of the past, present and prospective future. Certain resources may have already been allocated, could be tasked in the current moment, or could be pre-planned to be utilized in the future. The controller checks for information about the previous and current usage of certain assets and who enacted the changes. Finding the assets available, he checks the event timeline, which shows a visual depiction of pre-scripted events. Upon finding the assets in use in the future, he alters the pre-scripted events to utilize other assets and responds to the student that the request can be honored. This change of future events subsequently has an effect on the decision-making of other controllers.

As can be seen by this example, the controller needs access to and influence over temporal information that may have been contributed by his co-workers before and during exercise time. The use of timelines may provide an appropriate view and means to manage information, so long as there is not too much to filter through.

Figure 1 depicts a typical collaborative training environment in this domain. As previously indicated, a training effort based on NIMS can involve the management of a high to low degree of incident detail from ICP to MACC. In addition, the scenario itself evolves through a life-cycle that moves through the following stages:

- Design – 5 to 15 scenario designers construct the story of the incident and create stimuli that relate to training goals.
- Specification – 2 to 3 scenario developers (training system experts) map pre-scripted stimuli to training system abilities and specify the scenario to the system.
- Execution – 5 to 15 trainers, depending on scenario size and exercise configuration, contribute pre-scripted and actively-scripted stimuli.
- Assessment – 2 to 3 evaluators assess student performance through collaboration with trainers.

Each stage involves groups of collaborators with distinct roles which contribute to the training effort. These groups also vary in the degree to which they need tools to support the authoring and understanding of the scenario. Typically, some scenario designers and developers are also trainers, and some trainers are evaluators, which provides for continuity throughout the training effort. After each exercise, the cycle continues whereby the scenario design itself is evolved to better support future training. STS provides tools which can be used in conjunction with an exercise management training system throughout the Specification, Execution and Assessment stages.

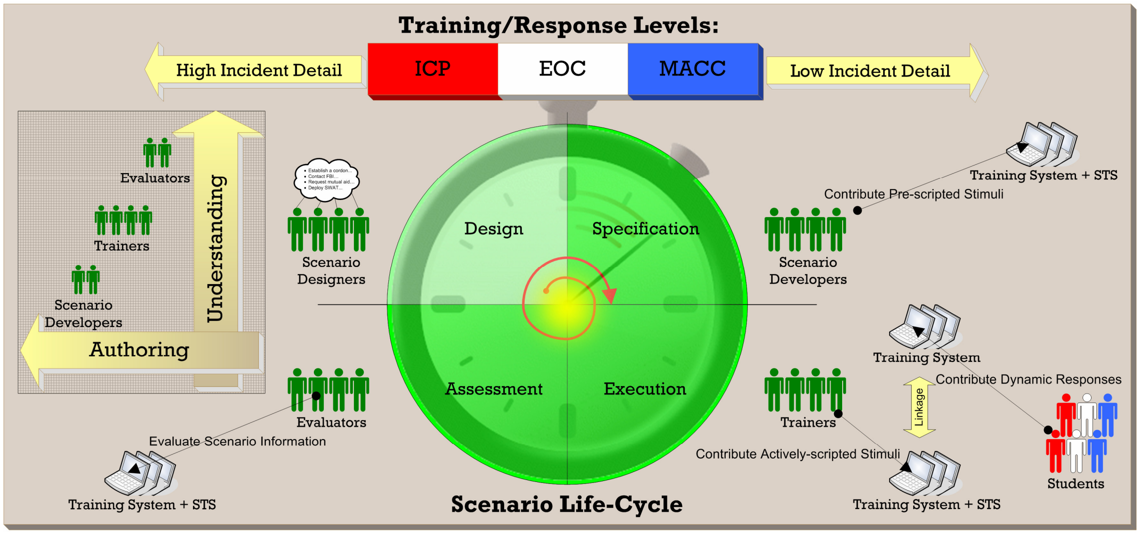

Figure 1: Training Collaboration Environment

RELATED WORK

The necessity for exercise management systems in the context of emergency response training stems from the quantity of information involved and the fact that, historically, exercise management was accomplished through non-digital means. Pen and paper may work well for managing a small incident, but when the scenario scales to hundreds of resources, there is a need for computational tools to effectively manage time-relative, pre-scripted and actively scripted scenario information. In addition, controllers need tools that can aid them in their decision-making processes while an exercise is running. An effective exercise management tool needs to support the integration of pre-scripted and dynamic material in the exercise, capture data generated during the run of the exercise, and facilitate access to this information.

The primary collaborative challenges involve providing shared-awareness about the synchronous work of others from both the *system perspective* (e.g. who is doing what right now) and the *scenario perspective* (e.g. who has planned what to happen at a point in the exercise). Controllers need to know who is concurrently interacting with the same system artifacts and also what is going on in the scenario that affects their current decision making process. From the system perspective, Gutwin and Greenberg [1998] identify certain tensions between what is advantageous in groupware to the single user and to the group. Particularly, their proposed balance of concerns for the notion of artifact manipulation and view representation is an appropriate approach to the information problem found in exercise management. The concept of *activity awareness* identified in [Carroll et al. 2003] describes the challenge of providing awareness from a project or group goal perspective. These conventional activity awareness techniques from Computer Supported Collaborative Work (CSCW) can be effective for the system perspective and can be situated to support the scenario perspective.

Timelines are one technique for providing a view of a scenario or exercise. Numerous systems include interactive graphical timelines to support users' access to and comprehension of temporal information [Ahlberg and Shneiderman 1994; Karam 1994; Kumar et al. 1998; Monroy et al. 2003; Owen et al. 1994; Plaisant et al. 1996; Shiaw et al. 2004; Swan and Allan 2000]. Some systems, such as *Xtg* [Karam 1994] and *Timelines* [Owen et al. 1994] offer limited capabilities to annotate and edit timeline data. Such systems share some of the goals of scenario co-authoring and management since their goal is to create an analytically rich and human readable timeline with controls for information visualization.

Flow diagramming, visual programming and simulation products are very common and powerful for articulating logical relationships and processes. Some such as *Simulink* [MathWorks 2008] and *LabVIEW* [Lipovszki 2004] even allow time-based execution of designs. In terms of running an exercise, however, there is a necessary balance between customization and automation. Simulink is exemplar in providing capabilities that allow users to assess and manually control the temporal process of execution. In this training domain, the controllers need scenario tools that are visual and collaborative and that closely integrate the concepts of the training domain. For instance, a controller may need to pause, disable, or immediately execute injects or vignettes at will during runtime, where others can see those actions from a common operating picture. In Simulink there is an added complexity of having to specify complex logical elements and deal with system technicalities that can distract a controller from the domain. Additionally, it is desirable to enable them to maintain an understanding of how the components executed over time and how well students completed their training objectives.

Multimedia production is the most prevalent and common domain for graphical timeline editing. Commercial tools such as Adobe *Flash* and *Premiere* abound for the authoring of audio and video works using timelines. In [2000], Allen and Acheson describe a multimedia story authoring tool that allows tagging of plot elements, and the synchronization and categorization of them according to a certain narratology. Systems such as *Pavlov* [Wolber 1998] extend multimedia editing to programming by demonstration contexts. Outside of multimedia, there are timeline authoring tools such as *TimeSpace* [Krishnan and Jones 2005] which allows a user to author timelines based on file system elements. These systems support the authoring of graphical timelines, but they lack support for synchronous collaboration.

From the perspective of visualization, Kurihara et al. [2005] describe specific challenges with the use of timelines in multimedia authorship which are also very appropriate to scenario management. First is the fact that a timeline generally is linear, and certain effects such as looping may not be realizable. Secondly, static timelines usually depict a global or high-level perspective of a range of time, which can make project scalability a problem. Lastly, timelines that enforce a rigid specification of time for elements can prohibit users from "working rough", or working in an informal, less rigid, context [Kurihara et al. 2005], which is important when scenarios are worked out in multiple drafts, similarly to any writing project. A timeline should really augment the abilities of the controller, not take over. Some other challenges are similar to those found in the area of group writing, such as alerting collaborative users when another author is editing the same elements [Ellis et al. 1991], along with other activity awareness intricacies.

Timelines have been used in an educational context by Carroll et al. [2003] in an asynchronous and synchronous collaborative system called *Virtual School*. This system includes active timelines that help users to organize their work based on shared-awareness provided by the system. Their system allows users to collaborate by publishing their progress on particular portions of a project. This in turn is indicated on a timeline which shows a group perspective of work accomplished, in-progress, and still to be done. The Virtual School timeline is updated based on the work of others, but there is no information being pushed to the students from the timeline. The timeline can be modified by, but not modify other parts of the system. This is an important distinction, since an inject timeline situated in an emergency management system would be a source of active stimulus to the students, not just a means for collaboration between controllers.

Livnat et al. [2005] describe the *VisAlert* system, which depicts correlated artifacts in a radial timeline visualization for enhancing situational awareness in varied contexts, such as network intrusion detection and emergency response. The radial timeline allows a viewer to sense correlations between types of alerts that occur and to burrow down further to obtain more detailed information.

The described work shows that timelines can be effective at representing dynamic temporal data, and that interactive timelines can be used to raise awareness of their users. Providing support for direct timeline co-authoring and group awareness about the work of synchronous information contributors from the system and scenario perspective are the primary needs. For a small number of users, certain solutions like informal communication, browsing, and simple visual cues suffice, but for a larger group the issue becomes more complicated, since such methods of information access can be distracting and time-consuming. For large simulations, an additional challenge is getting the appropriate information to users, based on their current work in the simulation. The system described in this thesis combines dynamic visualizations, information retrieval and group awareness features to address some of the challenges of synchronous co-authors found in this domain.

SCENARIO TIMELINE SYSTEM (STS)

At the Texas Center for Applied Technology (TCAT), located at Texas A&M University, there are two ongoing projects that use the interactive timelines in STS to help emergency response simulation controllers conduct large-scale decision-making exercises. These systems, named EOTCS (Emergency Operations Training Center Simulation) and EM*ES (Emergency Management Exercise System), are intended to help controllers co-author and execute scenario-based exercises. Specifically, EOTCS is used to train members of the Incident Command Post (ICP), whereas EM*ES is for members of the ICP, Emergency Operations Center (EOC), and Multi-Agency Coordination Center (MACC). Although these groups operate at different levels of concern and decision making, their need to actively manage logical, hierarchical and temporal information is very similar. Both use timelines as a means for managing and presenting dynamic time-relative stimuli and increasing awareness of the current state of a running scenario.

STS is a collaborative, graphical timeline system that is a major component of both EOTCS and EM*ES. The system supports two primary activities: *scenario editing* and *scenario understanding*. Scenario editing is accomplished by the co-authoring and management of event-based timelines used to automatically present stimuli to students. The system allows scenario developers to create events that can be affiliated with some effect in the system such as a message being sent to the students or a video being played. Events can also enable or disable other events in the future. The constructed timeline can then be executed to automate the presentation of stimuli. Authoring occurs both at scenario design time and at exercise run time to allow customizations based on student behavior. Scenario understanding involves the retrieval and presentation of information about the past, present and prospective future of the scenario. This activity is useful for controllers during exercise run and for the development of an After Action Review (AAR), which is a presentation of student performance with regard to training objectives.

STS has undergone an iterative design process since its first development for EOTCS in 2003. The following sections describe the primary features and limitations identified by internal assessments and informal user feedback at each stage of its evolution.

EOTCS – Design Iteration 1

In the EOTCS project, there were several challenges that existed from the onset of the requirements gathering process. Namely, as is the case with many software development projects, the needs of the users were not easily identified. The scenario developers had difficulty deciding how to balance between the automation and control they desired, and mainly conveyed examples of features from their gaming experiences. In this case, providing tools that supported scenario editing and understanding went well beyond the scope of their experience.

The overall goal of the scenario authors was to dynamically inject stimuli into the simulation to facilitate learning via an information push/pull cycle. These injects were traditionally manifested in the form of MSELs (Master Scenario Event Lists), which included simple line item descriptions with relative times that could possibly be associated with more complex descriptions. Example high-level MSEL entries are shown in Table 1. They can also be accompanied by annotations as to what training objectives are correlated. These lists allowed controllers to run tabletop exercises which engaged certain training objectives. During an exercise, controllers would use a MSEL to direct the flow of events, while annotating actively scripted additions and responses to the scenario as time progressed.

Table 1: Example MSEL Items

Time	Inject Description	Training Objective/Desired Outcome
+5 min.	911 call received.	Conscientious initial response accomplished.
+10 min.	Message: "I see a gas plume rising from the complex!"	Cautious response: Hazmat Team Activated, Fire, Police, EMS services sent to the scene.
+20 min.	Message: "Hundreds of people all over are writhing and vomiting!"	Remote Triage Center establishment planned. Increased medical personnel requested.
+25 min.	Chemical makeup of plume identified.	Incident Action Plan is established. Evacuation mandated and plan is established and executed. Mayor issues State of Emergency declaration.
+30 min.	News report about incident is broadcast.	Chief Information Officer holds press conference.
+35 min.	Casualties located.	Family information center established.
+60 min.	Request issued for more resources/equipment.	EOC contacted and requests for mutual aid formalized.
+70 min.	FBI arrives and proceeds with criminal investigation.	Plan for cooperation with FBI and coordination with neighboring jurisdictions formalized.

One of the difficulties in using MSELs for large scale, long running scenarios is the increased cognitive and clerical complexity required to manage variants that arise from student and controller responses to stimuli from each other. In such a situation, the complex dependency relationships that can exist between injects can shift the controller's focus from precise timing to the rough shifting of more immediate injects as they approach the current time. Therefore, the primary challenge addressed in this iteration was how to provide the scenario developers and controllers with a tool that allowed them to quickly adjust the relative spacing of injects in a rough or imprecise manner, while maintaining a semantic relation to their traditional use of MSELs.

Early in the design process it was argued whether providing a simple tabular timeline was most appropriate considering that the MSELs were tabular and the users would be already familiar with such an interface. However, due to the observed need to enable rough and quick adjustments of injects, it was decided that a tabular view would not give the user enough flexibility for inject management or feedback for quick understanding of the relative inject times. Based on this assessment of the MSELs and the scenario developers' desire for flexibility, design continued with an approach to scenario editing that allowed authors to create macros that could be dragged onto a scalable, graphical timeline upon demand. A macro is simply a container for various actions that can be accomplished in the EOTCS. Actions in this case could be any activity that can be pre-scripted for a scenario. Examples include sending a message, showing a shape on the map, moving a resource, and adding another macro for execution. The timeline supports the viewing and addition of macros by synchronous users. As shown in a screenshot of the original Event Timeline in Figure 2, macros are visualized by a vertical line that can be dragged to adjust their time of execution. The line is assigned a color based on whether the macro was pending execution, completed, or in error. The timeline scale can be adjusted to show larger or smaller periods of time, and can be automatically scrolled to follow the yellow current time indicator.

In this design iteration, the only additional tool given for understanding scenarios was a tabular view of macros that identified the number of times a macro executed.

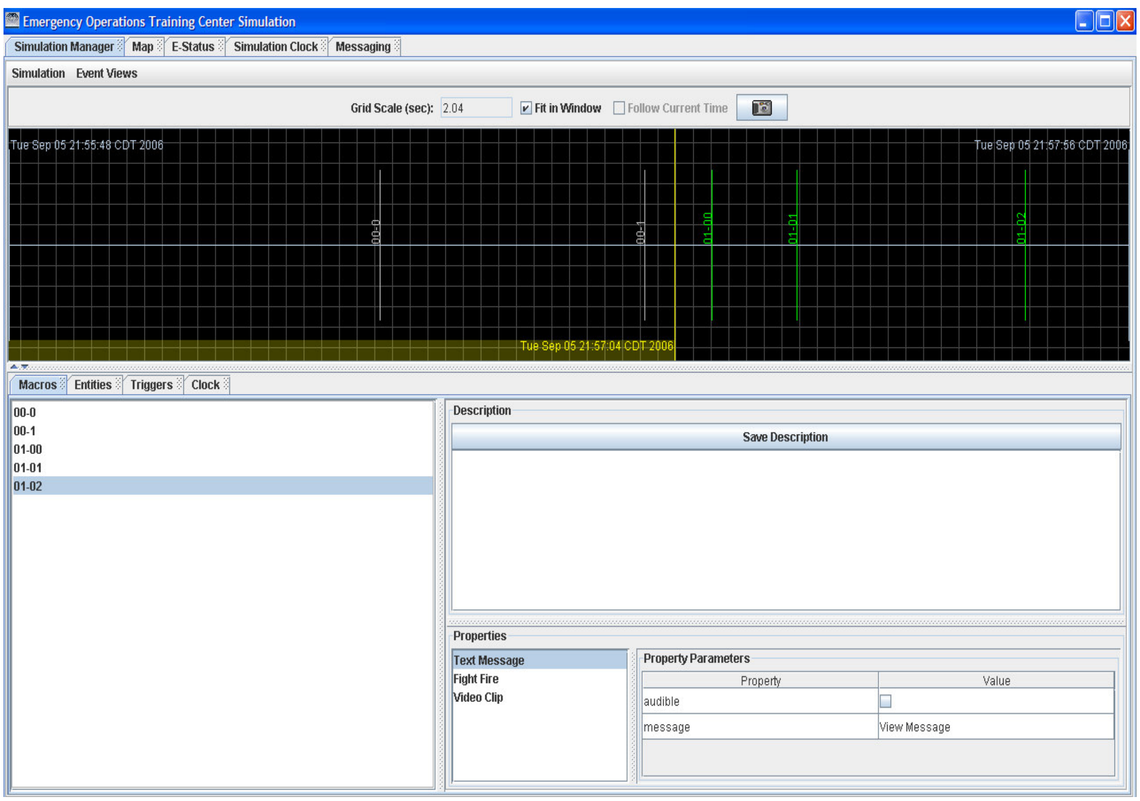

Figure 2: EOTCS Event Timeline

STS was deployed in EOTCS and is currently used in monthly training courses that involve up to 40 students and 20 controllers. Over 2000 students have been trained using the system to date. Six different scenarios are used in each course run. This use in real exercises uncovered limitations of the original synchronous timeline features and interface:

- Macro visual elements were difficult to manipulate due to their thin profile, and the timeline was difficult to read because of the text orientation.
- Dynamic timeline construction allowed for too much flexibility in structure of stimuli, which caused consistency issues between different exercises with the same scenario.
- Parallelism (concurrent macro execution) was difficult to understand from the visualization.
- Actions did not support multiple targets (e.g. move multiple resources to the same location)
- Invalidly set action properties were difficult to detect (i.e. the user would not find out that an action was invalid until it was executed).
- The timeline did not give enough feedback to allow controllers to understand how the scenario changed over time.
- Simple movement of individual macros on the timeline did not give enough control over sets of related macros.
- Macro dependencies and preconditions were difficult or impossible to specify.

During the design phase of this first version of STS, there were serious concerns about the fact that it would not fully address some fundamental domain problems. However, more features were not added because it was seemed simple enough for the scenario designers to understand and use and yet provided a basic flexibility for scenario specification and execution. The scenario developers already had to design the scenario while accounting for the features available in EOTCS. Based on informal assessments of a prototype of STS, it was clear that, since the developers were already consumed by the complexity of EOTCS, they were not prepared to introduce a more rigid structure on the timeline. They appreciated that this version offered a vehicle for user adaptation to a more robust graphical timeline environment since scenario development at the ICP level is very complex and detailed. It enabled them to develop and specify these complex scenarios without a heavier burden of transition from existing practices.

After an analysis of the work done by the scenario authors over time, it became clear that the necessity to closely relate to MSELs was at an end. As scenarios became increasingly complex, they more closely modeled decision trees. Decision trees are flow charts that model the potential paths that a scenario might take based on student behavior. Since there are many possible outcomes to a situation, the scenario developers would prune the possibilities by classifying student responses as good, bad, or acceptable. One example is the assessment of a structure that is about to collapse. Depending on the actions of the students, the paths of the decision tree could be for the structure to collapse, degrade further without collapse, or remain standing. The appropriate path would then be executed based on that assessment. Since the users had been transitioned into a graphical environment and were less critical and intimidated by the computer, it was clear that they were ready to move on to more complex interfaces. In order to support future scenario development using decision trees, the timeline should enable the management and visualization of these trees.

EM*ES – Design Iteration 2

For the EM*ES project, scenarios were initially represented in less detail than in EOTCS. This is because the target students trained as an EOC, which supports the operations of multiple ICPs. Stimuli were less detailed and less frequent because the training of the particular EOC supported in this iteration enforced an information push/pull cycle that injected stimuli in bursts. This meant that an event timeline for their scenarios was dense at places and sparse in others. The applicability of decision trees still held in this case, but the user-base was different. The scenario authors in this iteration were much more comfortable with the combination of graphical environments and the lower fidelity scenario development involved in an EOC training environment.

Scenario Editing

Recalling the observed limitations of STS in EOTCS, the system was redesigned in order to better facilitate the editing needs of the more adapted scenario authors in this project. The graphical timeline, shown in Figure 3, depicts time on the X-Axis and parallelism on the Y-Axis. The concept of macros was replaced with sequences and actions with events. This allowed for more structured content since events could be grouped together and arranged in time relative to the sequence. Sequences themselves could be arranged in parallel and dependencies could be described between them. A dependency is simply an execution pre-condition that is built on a logic operation (AND/OR).

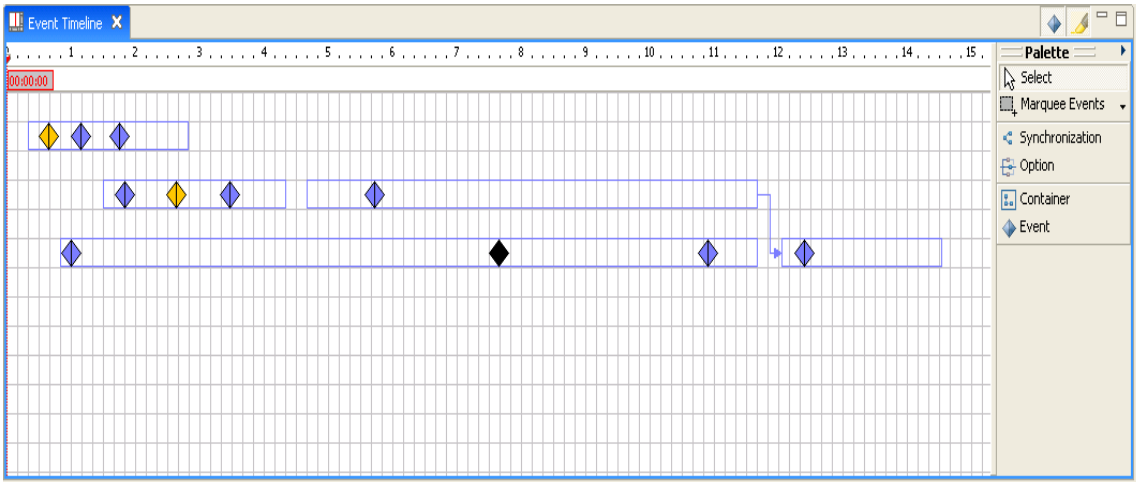

Figure 3: EM*ES Timeline

This visualization of the Event Timeline is similar to that used by Hardman et al. [1999] for the visualization and interactivity of hypermedia events and their dependencies. Events are represented as diamonds where begin and end times can be in the same place (in the middle), but support visual representation of those that represent durations. Arrows represent dependencies and rectangles visualize sequences. Events and sequences were designed to take on the following states:

- Paused – element will wait to execute until changed to pending.
- Pending – element is ready to execute.
- Disabled – element will not execute.
- Executing – element is executing.
- Completed – element has successfully executed.

- Error – an error has occurred during execution.
- Invalid – properties of the element are not valid for execution.
- Waiting – element's preconditions have not been met.

Upon entering a state, an event is colored to give feedback on the timeline.

In this version, collaborative authors can synchronously add, edit and delete timeline elements, although no mechanisms for conflict resolution or feedback are provided. To add an element, the user can click on a palette item and drop it on the timeline. Editing can be done visually by dragging various handles of the element or more precisely through selection-based property pages. Based on what is selected in the timeline, a set of properties for the element(s) are provided in a separate view, which are editable via form fields and controls. Deleting events and changing states can be accomplished via context-menu item selection. Dependencies are added using a *Synchronization* or *Option* timeline element. The Synchronization element establishes a hard dependency between events. This ensures that the dependency executes fully before the target. An Option element ensures that at least one optional dependency is met before the target element executes. If an element's dependencies are not met, the timeline will shift the element forward in time, changing its state to *waiting*. If an element is a dependency of another element and is disabled, the rest of the decision tree will be disabled. Sequences that overlap in rows are considered implicitly synchronized, meaning, if a predecessor shifts in time, all successors shift in time to preserve the spacing articulated by the designer. The user has complete influence over the execution process and can change the timeline as necessary to produce a desired effect even when time is running. The only additional view provided was a tabular outline of the timeline that provided event state colors, names and times for quick recognition and minor manipulation of elements.

After EM*ES was deployed, it was run in 3 different exercises involving 15 students, 6 controllers and 2 scenarios. It was clear by observation that the timeline was much more successful in supporting scenario editing than the previous version. Many of the previous problems were resolved, but not all. Some attempts at alleviating problems caused additional problems. The most significant limitations found in this iteration include:

- Event types were hard to distinguish since they are all shaped the same.
- The timeline enforced too much structure into the scenario, since templates of events could not be specified easily and the timeline elements could not be added without significant effort.
- Sequences could not be nested, which inhibited the grouping of elements into vignettes that include parallelism.
- Events did not support multiple targets (e.g. moving multiple resources to the same location)
- The lack of group and activity awareness features made co-authoring confusing.
- The relationship between explicitly defined dependencies and event execution was ambiguous, making the timeline seem unpredictable to the user.
- Large timelines were difficult to navigate because of the need to constantly be zooming and scrolling back and forth between various regions.

Scenario Understanding

In addition to the scenario editing timeline, a set of independent timeline interfaces for viewing information recorded during an exercise are included in this design. We call these *Correlation Timelines* as they allow controllers to combine and correlate information collected from various data points in EM*ES. Correlation timelines allow a controller at exercise run time to discover topical information that can be used to accomplish his or her goal. Certain group awareness mechanisms are provided by default. The actions of users in the exercise management system are logged and the author of those changes is identified in the correlation timeline.

The Resource Correlation Timeline provides information about the times a resource has been manipulated in the scenario (Figure 4). Rectangles indicate the duration of a particular resource order (e.g. move, patrol). The controllers can click on the elements to obtain details of actions. The Messaging Correlation Timeline is similar to the Resource Correlation Timeline but provides access to the message traffic generated in the system.

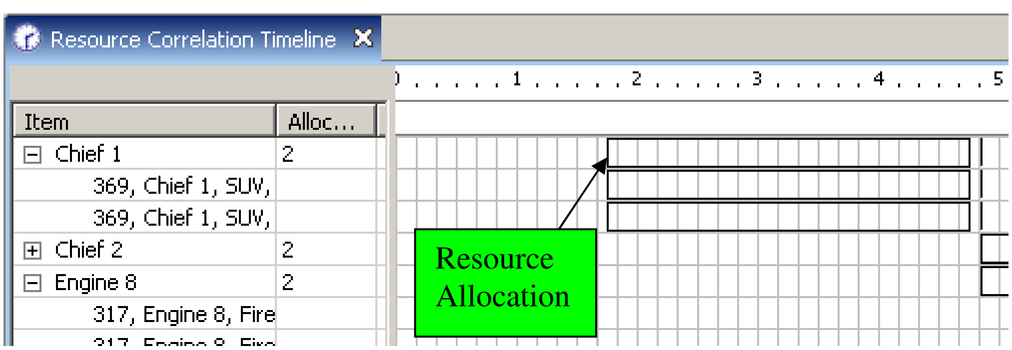

Figure 4: Resource Correlation Timeline

These timelines can be used with the Event Timeline to draw conclusions about temporally relative information from various components of EM*ES. For example, a controller might observe that within a certain range of time from the execution of an event, a series of messages about the stimulus were generated by a set of users. Such a discovery aids the production of an After Action Review and decision-making by allowing the reviewers to correlate the events of the timeline with student responses. A user can click on the timeline elements to see details about the particular action that has occurred and can correlate the time with elements on the Event Timeline.

In this iteration, the correlation timelines provided more information about the exercise, but the following limitations made them difficult to use in order to meet controller information needs:

- Correlation timelines did not scale well and overloaded the user with information as there was no means for filtering out non-relevant information.
- Vertical lines on the timeline made selection for viewing details difficult.
- Having one timeline per data source (e.g. messages, resources) did not scale well, as more timelines made the correlation timeline concept unmanageable.

Current Version – Design Iteration 3

STS is now in its third iteration of development and along with the EM*ES project, is being deployed with the MACC (Multi-Agency Coordination Center) training project. The new venue is a step above EOC level operations, whereby multiple EOCs can be supported by a MACC. However, the same needs for scenario management exist since the differences between scenarios relate to stimuli frequency and level of detail.

Scenario Editing

The current version of STS retains the concept of events, but replaces sequences with containers. Containers allow nesting of other containers and events to allow authors to more appropriately model scenario vignettes. The visualization of events is improved by showing an icon related to the type of event. This is similar to the visualization by Carroll et al. [2003]. Containers can be filled with a background color to allow associations that are not nested. In addition, an event toolbox is provided which allows the creation of event templates that can be executed directly, or dragged to the timeline. Events now support multiple targets (i.e. multiple resources can now be moved with one event). Elements of the timeline can be collaboratively added, edited and deleted. Edits to the timeline are logged and controllers can see who originated the change. When an event is disabled, the entire dependency hierarchy, or tree branch for the event is disabled as well. Dependencies are assignable between mixed types of elements and parent hierarchies. In other words, any element may be made a dependency on any other element. The new Event Timeline is shown in Figure 5.

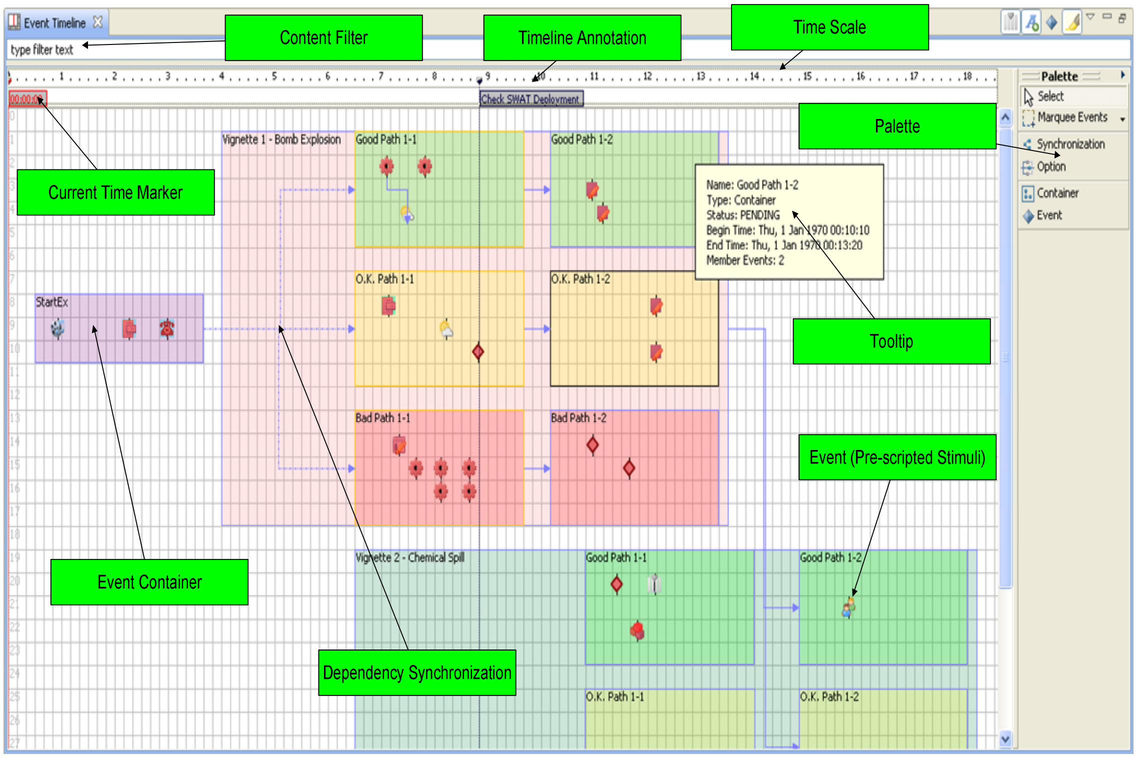

Figure 5: Timeline Visual Elements

The timeline can be zoomed and scaled according to seconds, minutes, days, months and years to support exercises that model long periods, or that start after an elapsed amount of time. The timeline can scroll with the red current time marker, and additional markers can be placed to share information among controllers. Additionally, the user can filter the timeline using a text field located at the top of the view. Any names or descriptions that do not match the filter are hidden in the visualization.

In previous versions, large timelines were difficult to manage because of the need for frequent zooming and scrolling to various regions. In this version, the user can open a new instance of the Event Timeline that can be independently zoomed and scrolled. This allows the user to have focus on various disparate regions of the timeline at the same time.

Some primary issues that still need to be addressed include:

- Proximity clustering or folding of timeline elements at various levels of scale and zoom.
- Improvement of group and activity awareness capabilities to support presentation of activities related to a particular controller.

Scenario Understanding

In this iteration, the correlation timelines are merged together to form a master timeline. Simple keyword filtering is added to minimize noise. The merged visualization (Figure 6) allows controllers to maximize screen real estate while still keeping a global perspective of the scenario. But as learned from previous limitations, a simple depiction is not adequate. Besides viewing the details of an artifact, the user can filter the correlation timeline to only show items that contain the text of the filter in one of its properties. Sample artifact details are shown in Figure 7 and general information for selected artifacts is shown in Figure 8.

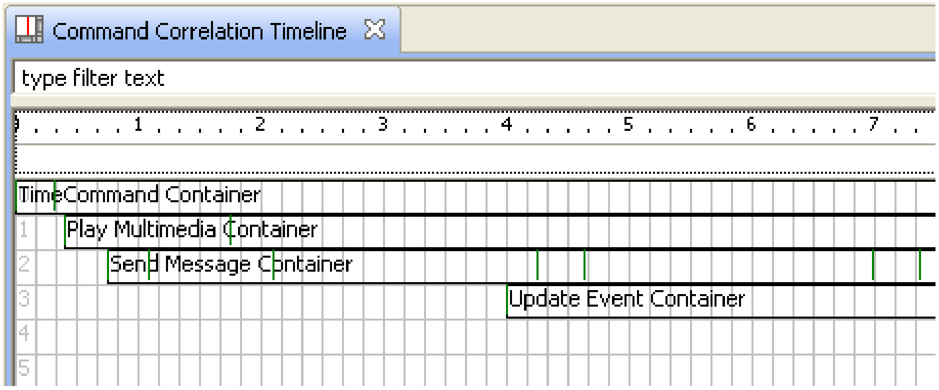

Figure 6: Actively Contributed Scenario Artifacts

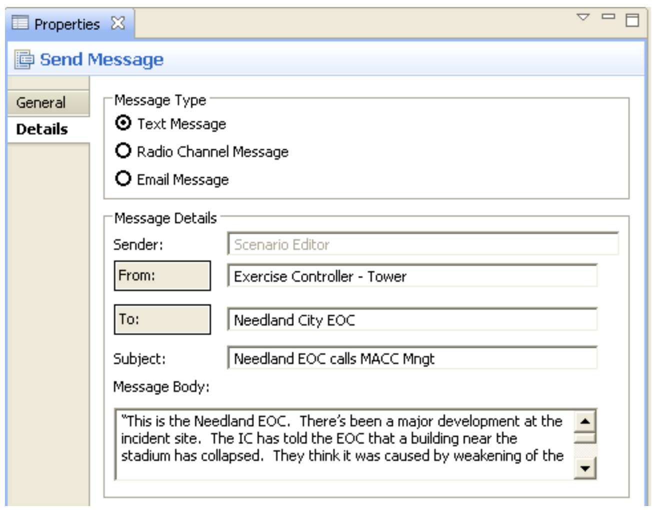

Figure 7: Sample Artifact Details

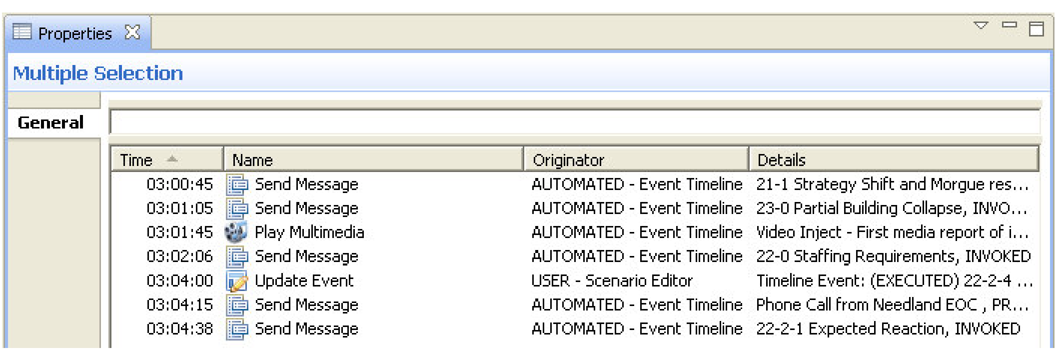

Figure 8: General Information on Selected Correlation Elements

As already mentioned, in addition to the Event Timeline, stimuli can also be contributed asynchronously through the other exercise system components, such as the messenger or resource manager. Due to the varied ways that stimuli can be injected, there is potential for confusion between the controllers. Tasks can overlap as well as the need for particular finite virtual resources such as fire trucks or police cars. This introduces a collaboration challenge since controllers need real-time information about the decisions of other controllers in the past, present and prospective future to accomplish their goals.

There are two particular types of system controllers that have the greatest need for complete scenario information access: role players and evaluators from the Training Assessment Facility (TAF). Role players have the duty of interacting with students through various means of communication. This can involve electronic messaging, phone and radio conversations, and other scenario event stimuli. These controllers play the roles of persons with whom the students must communicate to manage the incident. Such roles may include the mayor, the police chief, structural engineers, firefighters and other involved citizens. A role player's task can be difficult even to the most experienced controllers, since information given to students must be not only plausible, but sensible and consistent with the information that has been given already and which is planned to be given as part of future stimuli.

TAF members are those controllers that monitor the exercise to assess how well students have met the course training objectives. These objectives may be high-level or deal with certain specific situations that may arise during the exercise. The TAF has the job of monitoring all of the loose ends left by students and aiding other controllers in determining what stimuli should be injected next. The work of these controllers is very important since most of the other controllers are focused on executing the scenario and can have a difficult time keeping a high-level perspective. Since the assessment of student performance is based heavily on the observations of these members it is important that they be able to quickly access relevant information about the global state of the scenario.

For role players and TAF members, questions like "Can I give them the resources they ask for?" and "Have the students achieved this objective?" have no clear-cut answers. The answers to such questions require a synthesis of information from a variety of angles, which is difficult to achieve without the proper tools. The information need for controllers could be characterized by the berry-picking model [Bates 1989], since there are various aspects of the scenario that the controller must consider and different sources for the information.

Given the previous examples, the information tasks of role players generally require higher precision results. They are more concerned about very relevant, recent artifacts. In the case of the TAF members, recall is generally more important since their information tasks tend to involve a large span of time and a more global assessment of the collection as time goes on. In addition, different types of artifacts can be difficult to compare to each other, which complicates the process of ranking. One artifact of a certain type may be just as relevant as another based on what that the user deems as important. The role players generally need information during the exercise, while the

TAF members need information during and after the exercise. In the next sections, four experimental tools currently in development for scenario understanding are presented.

Scenario Oracle

The first experimental tool for scenario understanding is called the Scenario Oracle, which combines the concept of correlation timelines with context querying. The Scenario Oracle attempts to address problems with information overload by providing ranking, clustering and filtering capabilities to a visualization that correlates the data sources on a single radial timeline (Figure 9).

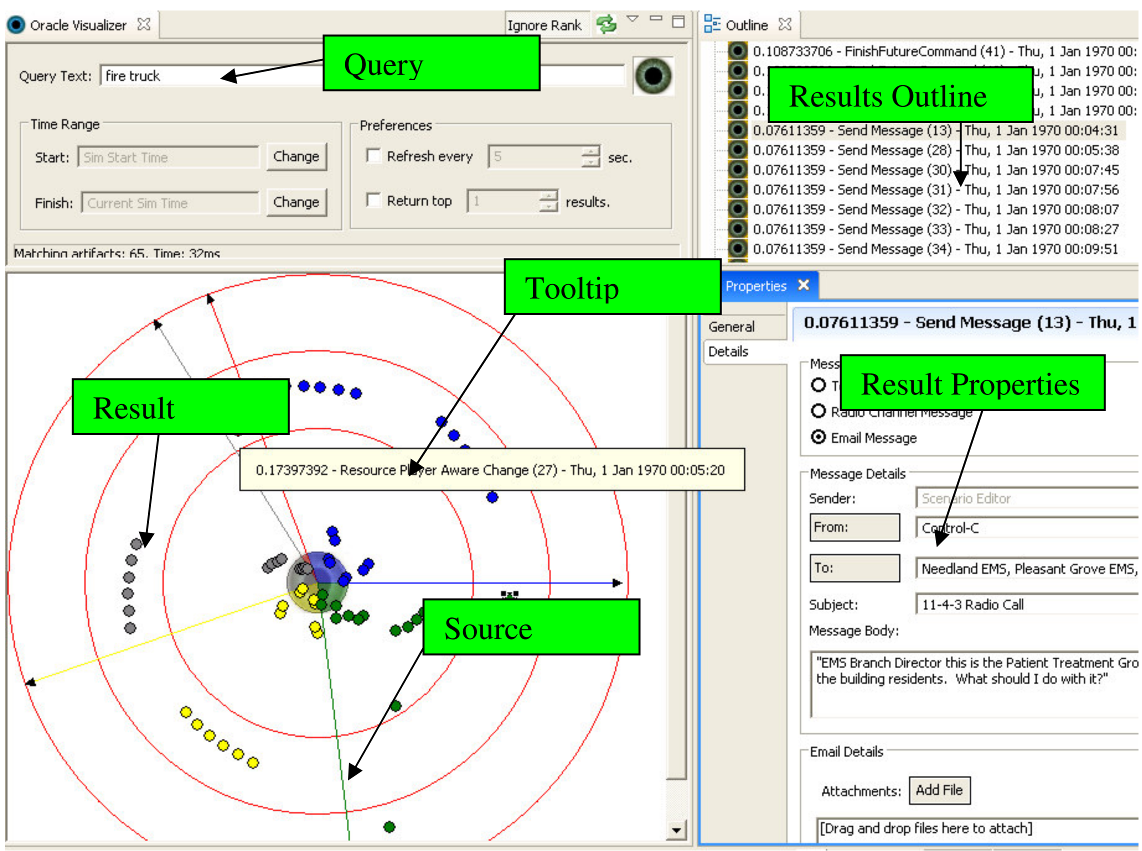

Figure 9: Scenario Oracle Interface

Information is presented in a graphical way, accommodating for the unique features of the data such as time of occurrence and data source component. The system automatically categorizes results by source and shows them on a radar-style plot. This is significant since the relevance of the type of artifacts (e.g. messages, resource orders, etc.) depends on the information task. In the future, the controller will be able to refine the query by visually adjusting the percentages of results for a particular type and removing items visualized. Time is depicted as concentric rings moving out from the center. Ranking is indicated by color alpha values and the timescale can be adjusted to allow for a greater or smaller temporal range of information. The system can also facilitate the execution of a query starting at a chosen time. Controllers can gain a contextual perspective of the running scenario, without collecting and managing such information manually, by querying the Scenario Oracle for information about a particular resource or theme. Future versions of the Scenario Oracle could use features of the controller's context to further improve the selection of information presented. The effectiveness of the Scenario Oracle interface has yet to be analyzed or determined, but the concept of querying for information in a scenario is arguably important.

Command Archive Manager

The work done on the Scenario Oracle resulted in a means to index scenario commands, which are the primary means for affecting change in the system. Simply having an index of every scenario command that executed during an exercise opens the door to vast possibilities for AAR development. The Command Archive Manager allows a user to open a command archive (index) and play back portions of it.

Commands can be played back instantly, or over time just as they were archived. One interesting point is that an entire scenario can be played back as if it were happening live. Commands from one archive could even be executed in a running exercise. The primary intent of this feature, however, is to enable TAF members to manually analyze previous executions of certain scenarios in detail to identify points of interest in the exercise at hand. The main interface (Figure 10) presents the archived commands in a tabular timeline view. The whole archive can be played back or just certain selected commands. As commands are executed, they are turned grey in the table, and show up in the aforementioned graphical Correlation Timeline just as it they were when the commands were first executed in the scenario.

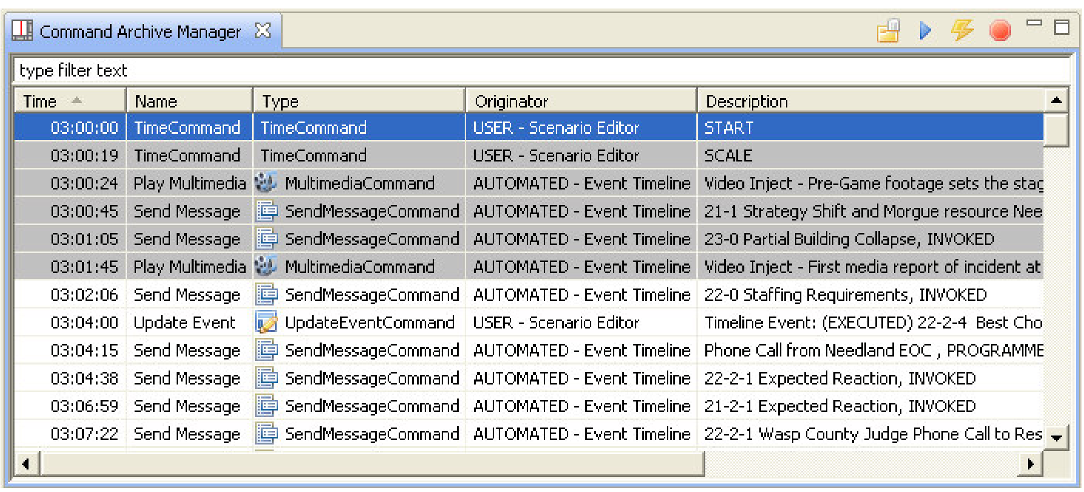

Figure 10: Command Archive Manager Execution

<u>Command Archive Analyzer</u>

Another experimental tool for scenario understanding is the Command Archive Analyzer. This tool facilitates the presentation of certain metrics that are derived from a selected set of command archives. The interface (Figure 11) presents the user with a means for opening a set of archives and generating graphical reports concerning particular metrics. The user can select which archives to process from the list of those open, and select the metric to derive via a context-menu.

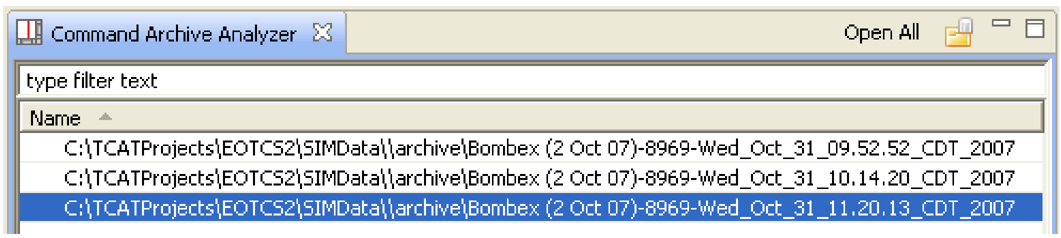

Figure 11: Command Archive Analyzer with Opened Archives

Currently, AAR development is a subjective process. There are no standard metrics defined in the domain that are considered authoritative for drawing conclusions about student performance. Current AAR practices tend to be very Socratic, where points of interest are identified and dialogue is facilitated such that critique is elicited from the students themselves. This is primarily due to the fact that there are many correct decisions made that can and often do result in an unfavorable outcome and many incorrect decisions that result in a positive outcome. It has been said by controllers that one can do everything right and three thousand people will still die. Realizing this, one might ask whether a valid assessment is possible. Since there is an interest in teaching the NIMS protocol, there can be valid critiques on deviation from it, but the remaining issues are reconciled against the expertise of the controllers. Nonetheless, certain metrics may be useful to a TAF member for drawing attention to certain periods of time for deeper analysis. Some experimental metrics provided by the analyzer include:

- Total Traffic Over Time – This consists of a line graph that may be useful for identifying expected periods of high traffic volume following injected stimuli.
- Total Categorical Traffic Over Time – This consists of a bar graph of traffic volume broken up according to category/type.
- Event Timeline Traffic Over Time – This is a line graph showing the number of injects over time, which may be useful for developing a benchmark for the exercise.
- Resource Allocation Volume Over Time – This is a line graph indicating the percentage of total resources allocated over time. This could be useful since there are distinct response phases that normally are associated with high resource allocation.
- Resource Return Rate Over Time – This is a line graph showing the percentage of allocated resources deactivated over time.
- Temporal Execution Difference – This is a line graph that shows the differences between the execution times of timeline injects for a set of scenarios. This could be useful if a delay from the pre-defined inject time could be considered a negative.

Even though there are no standard values available for benchmarking, these metrics could be used for comparison against previous exercises, or even a derived average. There are significant open research questions about the use of metrics in training evaluations for this domain. When the user selects a set of archives and the metric to report, a graphical chart similar to Figure 12 is displayed in a view. Multiple chart views can be opened at a time allowing the user to identify certain patterns that may be significant.

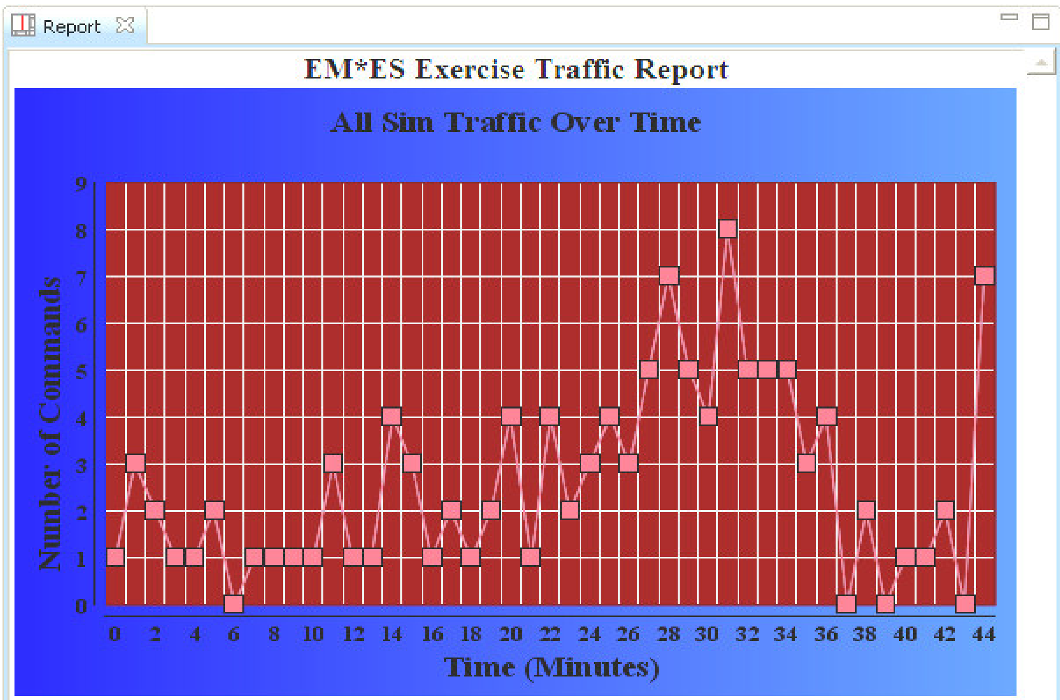

Figure 12: Sample Analyzer Report

The underlying mechanisms of this tool can be used during a running exercise, such that if certain standard metric patterns do emerge through experience, they could be tracked automatically. At that point a controller could be notified, and appropriate action could be taken before it is too late to change. One example would be if the system detects that certain injects are executing much later than the average execution time of the previous five scenario runs, a controller could be notified so that additional stimuli could be presented or a note could be taken for the AAR.

<u>Command Archive Visualizer</u>

The final experimental tool for scenario understanding is the Command Archive Visualizer. This tool allows the live execution or playback of a scenario to be visualized from a logical layout perspective. A user can load an image representation of the classroom and position proxies over computers or other locations that will act as models for participant's machines (Figure 13). When a scenario is executed, the information that is particular to a participant (e.g. an email message) can be accessed from the proxy via a timeline similar to the Messaging Correlation Timeline presented mentioned previously. As information flows in "real-time" to the proxies, summaries show above the proxy and fade away so that the scenario can be "watched". Every logical position in the scenario can be represented including the main timeline executor machine.

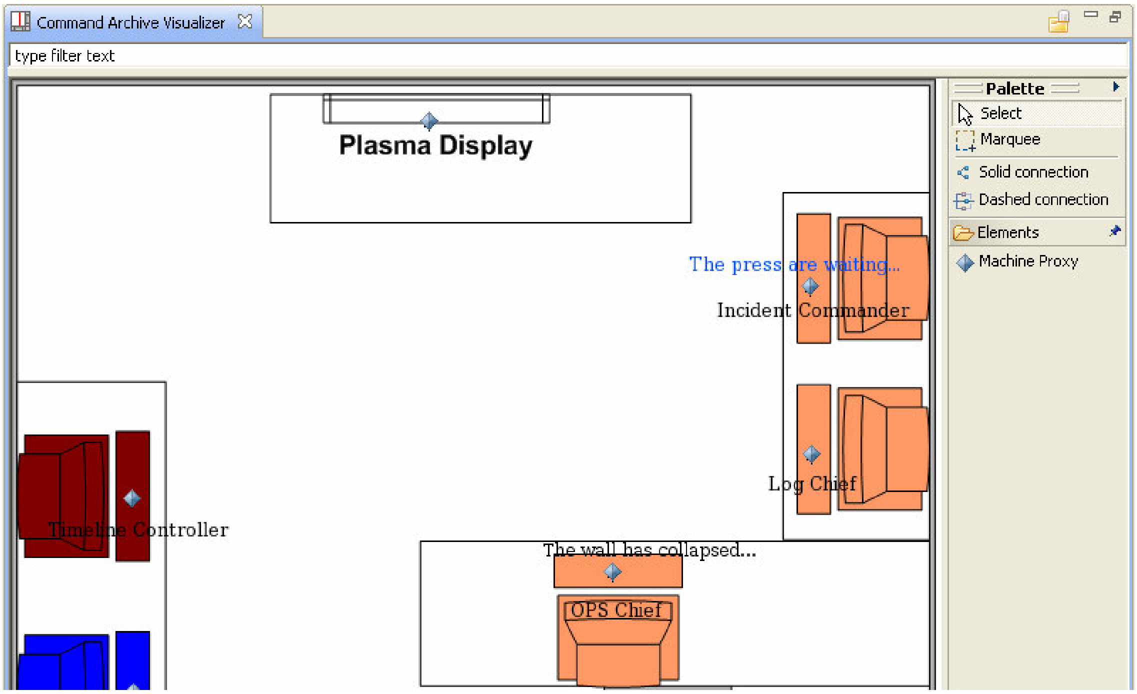

Figure 13: Command Archive Visualizer

The Correlation Timeline mentioned previously gives the user an analytical perspective, but is missing information that is grouped by participant. The Command Archive Visualizer attempts to model particular important aspects of the scenario from a participant-centric view. This tool, combined with the Correlation Timeline may allow a Scenario Developer to test some aspects of a scenario by analyzing the effects of injects, not just the source. During an exercise, a role player could monitor the message traffic of an individual to determine how well they are communicating about a problem or solution. The TAF members can use it during playback of a scenario in conjunction with the Command Archive Manager to analyze certain participant behaviors, especially in relation to the actions of controllers.

Lessons Learned

As STS has evolved through three design iterations, many limitations have been identified and taken into account. From these limitations come a set of internal lessons learned which can be used as guidelines for further development of STS or the creation of new systems that aid in the management of scenarios in the domain of emergency response training:

- Scenario Editing
  - Provide Group and Activity Awareness features.
  - Provide a means for depicting the edit history of a timeline element.
  - Provide notification of concurrent editing.
  - Provide feedback about why edits are not allowed due to certain constraints (e.g. not enough room, children sizes, dropping in the past)
- Visualization
  - Minimize information overload by providing filtering, scoping and scaling features.
  - Clearly depict parallelism and time scale.
  - Provide a means for clearing up congested areas and overlapping elements.
  - Clearly identify timeline elements.
  - Notify controllers about events that are not valid for execution (i.e. their properties have not been set properly).
- Scenario Structure
  - Allow assignment of notional activities or effects to events.

  - Enable containment and nesting of elements.
  - Support the definition of logical dependencies that are simple to understand at exercise runtime.
  - Allow scenario authors to work roughly by providing a way to make event templates, while encouraging structured timelines by making them simple to define.
- Event Manipulation
  - Enable events to have states that can be manipulated (e.g. paused, disabled, and completed/executed).
  - Make the execution of events based on dependencies predictable.
- Scenario Understanding
  - Minimize information overload by providing filtering and scoping features.
  - Facilitate information capture through integration with exercise management systems.
  - Provide information querying, retrieval and filtering.

There are three contexts in which the timelines are used: Exercise or scenario design time, execution or run time, and After Action Review (AAR) time. During design time, co-authors should be able to structure event stimuli in the form of decision trees, or other vignette representations, which allow certain paths to be executed during an exercise based on student behavior. During execution time, controllers should be able to add, delete, modify, pause, disable, or execute events upon demand, providing a high degree of flexibility to the controllers. Also, timelines that support scenario understanding should offer tools to obtain information that is relevant to controller information needs and decision-making processes. After the scenario is executed, the timelines should also provide overview and detailed information about the training exercise that can be used to construct an AAR and provide feedback to the students.

Since these lessons are internal observations, they do not provide a realistic view of the effectiveness of STS. Therefore, a user study has been conducted to obtain more insight into the positives and negatives of STS and the overall usefulness of graphical interactive timelines in the domain of scenario management in emergency response training.

EVALUATION

Objectives

To gain insight into the effectiveness of graphical timelines for scenario management in this domain, the Event and Correlation Timelines of STS were evaluated to determine which features are considered useful and not, and which are actually employed in practice. The focus on the Event and Correlation Timelines to the exclusion of other experimental timelines and tools is important because these two tools most fundamentally represent graphical timeline-based scenario authoring and understanding in STS. The desired result of this user-based evaluation is to obtain evidence that supports or refutes that graphical timelines can be used effectively for scenario management, not just if they can be used at all.

Rationale

There are many different methodologies that can be used for evaluating STS. They all have strengths and weaknesses in their approach and varying ways that their results can be contextualized. Any single method of evaluation in social and behavioral contexts is inadequate for obtaining reliable results since it is impossible to maximize the generalizability of results, precision measurement of behaviors, and realism of context in any one experiment. An increase in one aspect necessarily equates to a decrease in at least one of the others [Baecker 1994]. It is therefore important to use a range of evaluation methodologies which focus on one particular aspect to obtain results that can be correlated.

This evaluation is comprised of two methods, a judgment study and a field study. The judgment study was chosen as a means to increase the generalizability of results and precision of measurement of behaviors. The field study was chosen to increase the realism of results.

Method 1 – Judgment Study

Objectives

This study attempts to obtain evidence that can help identify the positive and negative features of the Event and Correlation Timelines, and how valuable they are for the management of large-scale scenarios.

Actors

The actors in this study were selected based on their degree of experience with STS and their background in training with scenario-based emergency management techniques in general. The total number of participants consisted of 6 controllers who have diverse backgrounds in training and have used STS for scenario development, exercise management, role playing, or AAR development. The set of actors cover a range of training levels from ICP to MACC. The actors in this study were volunteers and their confidential identities are represented as ID numbers.

Data Sources and Protocol

This judgment study was performed at the Emergency Operations Training Center (EOTC) at the Texas A&M Brayton Fire Field. The EOTC currently facilitates most of the emergency management training described, which made it a suitable location to conduct the survey.

Participants in this study first filled out a demographics survey (Appendix A) that included questions regarding their experience in the domain and with STS. The subjects were then asked to fill out a questionnaire (Appendix B) that consisted of yes/no and 7 point Likert Scale [1932] responses that ranged from "Strongly Agree" to "Strongly Disagree". A computer with STS was made available to each subject with a pre-loaded timeline to allow for better recall or experimentation. The time limit for filling out the survey was 45 minutes.

Results

Table 2 presents the abridged results of the demographic survey. Some specific personal information that was collected has not been presented over confidentiality concerns. All of the participants were males between the ages of 30 and 55. Their course training experience ranged from the MACC to ICP levels, and as a group, they cover all functional positions (e.g. TAF Member, Role Player, Principal Controller, etc.). Therefore, as a whole, the subjects represent all of the contexts where STS is used.

Table 2: Demographic Survey Results (Abridged)

Demographic Questions	Range
Time experience in scenario-based incident management training	1 month – 25 years
Time experience in using the Scenario Timeline System (STS) for scenario-based incident management training	1 month – 2 years
Time experience training scenario-based incident management using other computerized means	2 weeks – 15 years

Table 2 Continued

Demographic Questions	**Range**
Time experience training scenario-based incident management using other non-computerized means	5 – 20 years
Please circle your level of comfort with computers (1 = Not Comfortable, 7 = Extremely Comfortable)	5 - 7

Table 3 shows the averages and standard deviations of participant responses. Additionally, questionnaire items are grouped according to particular user behaviors and general system features. The items presented in the questionnaire can be categorized as positive, negative, frequency of use, and yes/no topics. The actual participant responses can be found in Table 4 in APPENDIX D.

Table 3: Questionnaire Results

Topic	**Average**	**StdDev**
Event Timeline Information Depiction	5.87	0.71
1. The Event Timeline depicts information in a way that is understandable.	5.83	0.75
2. I can understand the scenario vignettes depicted by the Event Timeline.	5.83	0.75
3. The Event Timeline clearly depicts the flow of the scenario up to the current time.	5.83	0.98
4. I can discern the potential paths that a scenario can follow using the Event Timeline.	6.17	0.75
5. The Event Timeline aids my understanding of a scenario as a whole.	5.67	1.21
Event Timeline Navigation	6.33	1.03

Table 3 Continued

Topic	**Average**	**StdDev**
6. I can navigate to areas of interest in the Event Timeline.	6.33	1.03
Event Timeline Frequency of Use	5.25	1.33
7. I examine the Event Timeline frequently during an exercise.	6.50	0.84
8. I examine the Event Timeline frequently after an exercise.	4.00	2.00
Event Timeline AAR Development	4.80	0.84
9. The Event Timeline is useful for producing an After Action Review.	4.80	0.84
Correlation Timeline Information Depiction	5.50	0.35
10. The Correlation Timeline depicts exercise activity in a way that is understandable.	5.80	0.45
11. I can discern periods of high and low activity using the Correlation Timeline.	5.00	1.00
12. From the Correlation Timeline, I can obtain details about particular activities that have occurred.	5.80	0.45
13. The Correlation Timeline aids my understanding of the activities of others in the exercise.	5.40	0.89
14. The Correlation Timeline and Event Timeline together allow me to correlate injected events to participant responses during exercises.	5.80	0.84
15. The Correlation Timeline aids my understanding of how an exercise is flowing.	5.20	1.30
Correlation Timeline Navigation	4.80	1.92
16. I can quickly navigate to areas of interest in the Correlation Timeline.	4.80	1.92
Correlation Timeline Frequency of Use	3.70	1.72
17. I examine the Correlation Timeline frequently during an exercise.	4.00	2.45

Table 3 Continued

Topic	**Average**	**StdDev**
18. I examine the Correlation Timeline frequently after an exercise.	3.40	1.14
Correlation Timeline AAR Development	4.40	0.55
19. The Correlation Timeline is useful for producing an After Action Review.	4.40	0.55
Inject Management	5.71	0.72
20. Editing scenario injects on the Event Timeline is easy.	5.17	0.75
21. I can adjust the times when injects (events) fire and vignettes start in the Event Timeline effectively.	5.83	1.17
22. The ability for events to change status on the Event Timeline is useful in managing an exercise.	6.00	0.89
23. The Event Timeline effectively allows me to construct a hierarchy of exercise injects and vignettes.	6.00	0.63
24. The ability to construct a hierarchy of injects and vignettes is important for running large-scale scenarios.	6.17	0.98
25. I can effectively define/manage dependencies among injects in the Event Timeline.	5.33	1.21
26. The ability to define/manage dependencies among injects is important for running large-scale scenarios.	5.50	0.55
Exercise Use	5.17	1.57
27. The Event Timeline is predictable when running/executing a scenario.	6.00	0.89
28. I edit the contents of the Event Timeline regularly during an exercise.	4.33	2.34
Collaboration Frequency (yes/no)	0.67	0.41
29a. I collaboratively develop scenarios with other authors using the timeline to edit the same scenario as others at the same time.	0.50	0.55
29b. I collaboratively develop scenarios with other authors using the timeline to edit the same scenario as others at different times.	0.83	0.41
Collaboration	5.80	0.84

Table 3 Continued

Topic	Average	StdDev
30. The Event Timeline aids my understanding of the activities of other collaborative scenario authors.	5.80	1.10
31. I can effectively and collaboratively author scenarios with others.	5.80	1.10
Scale of Use	5.67	0.61
32. The Event Timeline allows me to manage large amounts of information effectively.	5.83	0.41
33. The Correlation Timeline allows me to understand large amounts of information effectively.	5.40	1.34
General Satisfaction		
34. The timeline system as a whole is a distraction from my duties during an exercise.	1.83	1.17
35. My duties in running an exercise that presents large-scale incidents would be difficult without the timeline system.	6.17	1.60
36. Overall, I am satisfied with the timeline system.	6.33	0.52

Figure 14 depicts a summary of the averages with error bars based on the standard error of some particular categories of responses from the questionnaire. The values for the Event and Correlation Timelines are compared. Values for categories that do not apply to the Correlation Timeline are not represented in the figure.

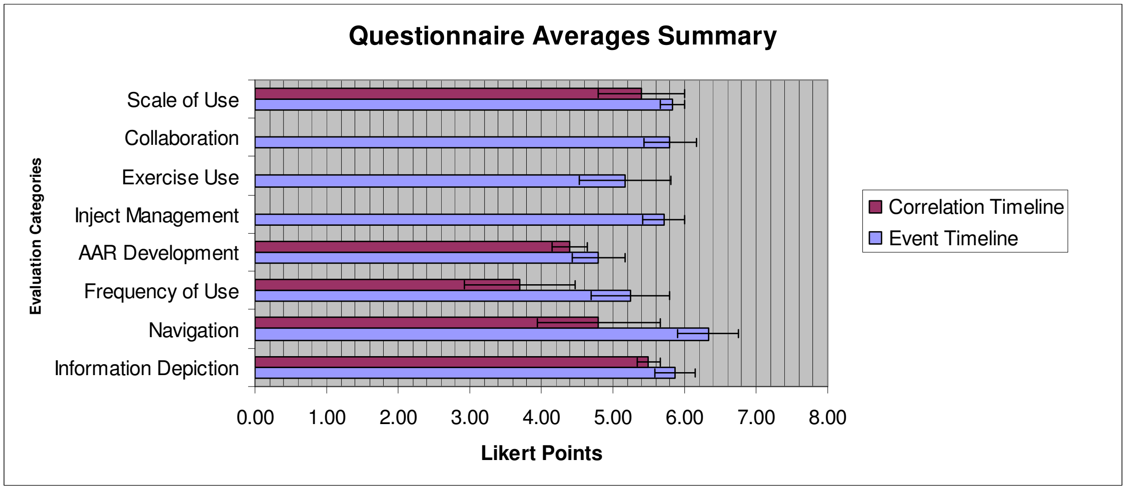

Figure 14: Questionnaire Averages Summary

Discussion

The results of from this study shed some light on the strengths and weaknesses of STS. Despite the small sample size, the subjects' responses are a valuable signpost for directions of improvement. A discussion of the results from a system feature perspective follows.

As can be seen in Table 3, according to the subjects, the Event Timeline generally depicts information well. The strongest way that information is depicted is in how a scenario could progress in the future. This is important for all controllers that need to make decisions based on what could happen in the near future. The ability to attain a holistic or "big picture" view of the scenario was rated the most significant shortcoming. The Event Timeline only gives information about pre-scripted injects that have executed, are currently executing, or may execute in the future, which leaves out a significant amount of information about an executing scenario. Even from the perspective of pre-scripted events, the Event Timeline may not adequately depict the scenario in a way that is conducive to attaining a “big picture”.

On average, the ability to navigate to areas of interest rated very high. This is an important rating, since all of the subjects’ experience with STS has been with scenarios that contain significant amounts of data.

The fact that the subjects frequently examined the Event Timeline during an exercise could be an indicator of usefulness, but more corroborating evidence is needed. The low rate of use after an exercise and mildly positive ratings for AAR development may indicate that the Event Timeline is limited in its usefulness in this context. However, the Event Timeline may be used for AAR development during an exercise. Additionally, the usefulness after an exercise depends heavily on the scenario injects structure to lend evaluative information. Scenario injects may have been structured such that certain alterations during execution indicate a positive or negative aspect to student activity. Some subjects’ lack of experience as TAF members may also be a contributing factor.

All of the results regarding the Correlation Timeline must be tempered by the fact that the subjects indicated that they do not frequently use it. Again, without more evidence, it is difficult to assume a level of usefulness based on this metric alone. The users may not study the timeline frequently because they obtain the information they need from it efficiently. The weakest, although positive, aspects of information depiction for the Correlation Timeline were discerning periods of high and low activity and how a scenario has unfolded in time. The first is surprising since the visualization is very similar to a histogram. It is clear that only one subject rated the latter poorly, which may be an indicator of some singular situation with that subject. Various aspects rated positively, and the depiction of information for correlation purposes by both timelines was rated well. Navigation using the correlation timeline was given a somewhat lukewarm rating, but again, one subject rated it poorly. AAR development was also lukewarm, but like for the Event Timeline, that may be due to the lack of experience of the subjects as members of the TAF.

Inject management was rated very positively overall, with the strongest features being the creation of inject hierarchies, which was also rated as an important task, and the usefulness of inject and vignette status changes. The weakest feature was inject editing. This depends heavily on the scenario structure, but it is clear that there needs to be a more simple way to edit injects and manage dependencies, which warrants further study.

The Event Timeline was rated highly for predictability, and it was edited frequently by subjects that have more than 6 months of experience as a Principal Exercise Controller. Collaboration was given a positive rating, for both supporting understanding of other collaborators' work, and allowing collaboration. The Event and Correlation timelines were also both given positive ratings for their ability to present large sets of data.

In general, the participants rated the timeline as not being very distracting from their work. Overall, the timeline system was considered very satisfying, and very useful in running large-scale scenarios.

Method 2 – Field Study

Objectives

This field study was conducted to gain insight into the ways in which STS is used in practice, in order to identify additional strengths and weaknesses that may not have been captured through the judgment study. It also serves as a means to rectify the results from the questionnaire, by identifying potential incongruities between observed behaviors and responses.

Actors

The participants in the judgment study agreed to a period of observation in their operational use of STS. Three of the original subjects were available for observation during the course of this evaluation period.

Data Sources and Protocols

This field study was conducted during an exercise that involved 4 days of training on different scenarios at the EOTC. All three of the participants were observed during the course of a single scenario, which lasted 3 hours, and one participant was observed through an additional scenario, which also took 3 hours. Results were recorded by myself, the principal investigator, and consisted of answers and notes guided by the form in Appendix C.

Results

The first observation period involved a subject who was the Principal Controller for the exercise, being observed during the course of a 3 hour scenario. More details about the scenario itself are omitted over confidentiality concerns. The Correlation Timeline was never used, although the Event Timeline was used extensively. The key points of interest from this observation session are the following:

- Information Depiction
    - The subject used the Event Timeline in bursts, and mainly for coordinating/preparing for what was about to happen in terms of injects.
    - The graphical view was used exclusively to obtain information about the present and future, and the tabular view was used once to obtain a complete picture about what happened in the past, when discrepancies arose.
    - The subject exhibited dissatisfaction with the visibility of inject status colors and icons at a high level of zoom.
- Navigation
    - Tooltips were mainly used to identify events.
    - Details were obtained by clicking and viewing the properties view.
    - Scrolling was used extensively.
    - The zoom level was initially set to fit the entire vertical span of the timeline without scrolling. The zoom level was never changed.
    - The timeline filter text field was never used.
    - Multiple timeline view instances were not used.
- Inject Management
    - Injects were edited using the graphical view, as opposed to the tabular view of the timeline, which were both visible at all times.

- Most vignettes/injects were initially paused, thus moving in time until un-paused by the user.
- Inject times were never edited directly, but rather, manipulated using status changes. The subject would either execute the inject using the "Execute Now" action, or by un-pausing a paused event.
- Dependency hierarchies set up in the scenario were used only as a visual organizer. They were never actually used to defer or synchronize events automatically.
- The subject desired to re-enable a disabled inject when its parent container had been disabled and time had already entered the parent, but could not succeed.
- Pre-paused vignettes that were not executed were either disabled or left paused as the scenario progressed.

- AAR Development
  - The Event Timeline was used for providing times for when certain key events occurred, verbally to a TAF member.
- General Behaviors/Observations
  - The subject never seemed confused or distracted because of the timeline representation of a scenario.
  - Other controllers were notified via radio of the time starting, speeding up, and slowing down, and of upcoming injects. Misunderstandings about

scenario flow were addressed via radio, using the timeline and a book of scenario details.

  - A time marker was added to mark a particular point of action by the students.

The second observation period involved all three participants in question for a 3 hour scenario. One subject was the Principal Controller, and the other two managed the role players in a room called the Green Cell. The Correlation Timeline was never used. The following key observations from all three participants include only those that were not observed in the previous observation period:

- Information Depiction
  - In addition to the present and future, the subjects in the Green Cell used the Event Timeline to view the immediate past.
  - The graphical view was used exclusively to obtain information about the past, present and future.
- Navigation
  - The zoom level was left at the default and was never changed. (Both Green Cell subjects).
  - The timeline filter text field was never used.
  - Multiple timeline view instances were used (one Green Cell subject).
  - All participants exhibited dissatisfaction with the features available to navigate between areas of the timeline where parallel or highly correlated events exist in different regions.

Discussion

Some of the results of the field study are surprising, while others are not. In the area of information depiction, it was informative to observe how users interacted with a scenario structured to inject information in bursts. The observations also indicate that for all participants, the graphical view was used more often than the tabular view. It makes sense that the tabular view would be used for resolving discrepancies about what has happened in the past, since it is a more concise presentation of the injects. It also seems that for managing large scenarios as the Principal Controller, there is a need for high zoom levels, which make the colors and icons difficult to see.

For navigation, it was not surprising that the tooltips were used frequently. However, it was unexpected that all of the subjects tended to scroll and use tooltips for navigating to an event. They all exhibited dissatisfaction with the features available for moving between different, but parallel events. The tabular view could have been used to automatically scroll to an event by browsing to and clicking on it, but it may have been too lengthy for browsing. Additionally, the filter text field at the top of the timeline was never used, which could have limited the number of elements viewed. The use of single zoom levels and more commonly a single timeline view were also surprising. These observations may indicate that the cognitive complexity of managing the scenario may cause users to desire a view space that does not change radically (e.g. zooming in and out), or that is as "single-threaded" as possible.

The fact that most of the injects in the scenario were paused as the default status when the exercise began indicates that the scenario author may not have wished to establish a static expected path or perhaps left the decisions about scenario flow up to the runtime controller. Also, the way in which inject times were managed gives more evidence that the users were perhaps interested in maintaining as static a view as possible, or at least one that changed in very definitive or stepwise ways. This may also explain why users tended to use the event dependency features as a visual guideline instead of for their power of automation. Perhaps the cognitive complexity of managing the scenario was too distracting, or perhaps that the users wanted more control.

Although the timelines are collaborative and can be accessed by other controllers, the TAF member developing the AAR (not a subject) asked the Principal Controller for certain inject times. This could simply be from a lack of their experience using the timelines.

A significant general observation was the large amount of radio traffic used to communicate about time changes and upcoming events. One helpful improvement may be to include an optional alarm or other indicator that notifies controllers of an upcoming event or time change.

Significance of Results

Analyzing the results of the judgment and field studies indicate that there are positive and negative correlations between certain questionnaire responses and recorded observations. It is clear that there are specific areas where STS needs improvement, particularly in information depiction at high zoom levels and navigation to related areas of the timeline. In addition, the Event and Correlation Timelines do not give the users an adequate holistic or “big picture” view of the scenario. Perhaps certain experimental tools like the Command Archive Visualizer may be helpful. In general, STS did seem to perform well for the subjects, despite these difficulties. There was nothing that prevented the subjects from accomplishing their objectives, and they all indicated that STS is highly effective for running a large-scale scenario.

CONCLUSIONS

Observations on the evolution of STS and its use in existing exercise management systems indicate that collaborative graphical timelines can aid in the management of large-scale scenarios. For the purpose of scenario editing, timelines with robust features for manipulating and visualizing logical structures can allow scenario developers to co-author more complex incidents, while enabling the flow of events to remain clear during an exercise. Timelines can also aid in the understanding of scenarios by depicting information that is categorized by source and correlated based on time.

STS has many areas where improvement can be made. Beyond the lessons learned throughout its evolution and evaluation, unresolved issues concerning the number of collaborative users arise as informal communication channels and existing group and activity awareness features become congested and noisy. Conducting an additional study with controllers collaborating intensely using STS would lead to a better understanding of the effectiveness of the awareness features in STS and what is further needed.

Despite the remaining challenges, newer designs of STS do provide effective means for the collaborative management of complex event-based scenarios. Tools for scenario creation and understanding, along with analytically-rich visualizations reduce the controller workload. By providing dynamic, collaborative timelines, complex, emergency response simulations can be successfully accomplished.

REFERENCES

AHLBERG, C., AND SHNEIDERMAN, B. 1994. Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Celebrating Interdependence* (Boston, MA). ACM Press, New York, NY, 313-317.

ALLEN, R. B., AND ACHESON, J. 2000. Browsing the Structure of Multimedia Stories. In *Proceedings of the Fifth ACM Conference on Digital Libraries* (San Antonio, TX). ACM Press, New York, NY, 11-18.

ARA (APPLIED RESEARCH ASSOCIATES, INC.). 2008. Automated Exercise and Assessment System. http://www.myaeas.com/.

BAECKER, R. M. 1994. *Readings in GroupWare and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration*, Morgan Kaufmann Publishers Inc., San Francisco, CA.

BATES, M. 1989. The Design of Browsing and Berrypicking Techniques for the Online Search Interface. *Online Review* 13, 5, 407-431.

C4IC (C4I CONSULTANTS, INC.). 2008. EDMSIM. http://www.c4ic.com/EDMSIM/EDMSIM.htm.

CARROLL, J. M., NEALE, D. C., ISENHOUR, P. L., ROSSON, M. B., AND MCCRICKARD, D. S. 2003. Notification and Awareness: Synchronizing Task-Oriented Collaborative Activity. *International Journal of Human-Computer Studies* 58, 5, 605-632.

CORLEY, J., AND LEJERSKAR, D. 2003. Simulation for Emergency Response: Homeland Defense Center Network - Capitalizing on Simulation, Modeling and Visualization for Emergency Preparedness, Response and Mitigation. In *Proceedings of the 35th Conference on Winter Simulation: Driving Innovation* (New Orleans, LA). Winter Simulation Conference, 1061-1067.

DHS (DEPARTMENT OF HOMELAND SECURITY). 2004. Fact Sheet: National Incident Management System (NIMS). http://www.dhs.gov/xnews/releases/press_release_0363.shtm.

ELLIS, C. A., GIBBS, S. J., AND REIN, G. 1991. Groupware: Some Issues and Experiences. *Communications of the ACM* 34, 1, 39-58.

ESI. 2008. WebEOC Professional Version 6.5. http://www.esi911.com/home/index.php?option=com_content&task=view&id=14&Itemid=42.

FRISHBERG, L. 2005. Looking Back at Plan AHEAD™: Exercising User-Centered Design in Emergency Management. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems* (Portland, OR). ACM Press, New York, NY, 988-1003.

GUTWIN, C., AND GREENBERG, S. 1998. Design for Individuals, Design for Groups: Tradeoffs Between Power and Workspace Awareness. In *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work* (Seattle, WA). ACM Press, New York, NY, 207-216.

HARDMAN, L., OSSENBRUGGEN, J. V., MULLENDER, K. S., RUTLEDGE, L., AND BULTERMAN, D. C. A. 1999. Do you have the time? Composition and Linking in Time-Based Hypermedia. In *Proceedings of the Tenth ACM Conference on Hypertext and Hypermedia : Returning to our Diverse Roots* (Darmstadt, Germany). ACM Press, New York, NY, 189-196.

JAIN, S., AND MCLEAN, C. 2003. Simulation for Emergency Response: A Framework for Modeling and Simulation for Emergency Response. In *Proceedings of the 35th Conference on Winter Simulation: Driving Innovation* (New Orleans, LA). Winter Simulation Conference, 1068-1076.

KARAM, G. M. 1994. Visualization Using Timelines. In *Proceedings of the 1994 ACM SIGSOFT International Symposium on Software Testing and Analysis* (Seattle, WA). ACM Press, New York, NY, 125-137.

KRISHNAN, A., AND JONES, S. 2005. TimeSpace: Activity-Based Temporal Visualisation of Personal Information Spaces. *Personal and Ubiquitous Computing* 9, 1, 46-65.

KUMAR, V., FURUTA, R., AND ALLEN, R. B. 1998. Metadata Visualization for Digital Libraries: Interactive Timeline Editing and Review. In *Proceedings of the Third ACM Conference on Digital Libraries* (Pittsburgh, PA). ACM Press, New York, NY, 126-133.

KURIHARA, K., VRONAY, D., AND IGARASHI, T. 2005. Flexible Timeline User Interface Using Constraints. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems* (Portland, OR). ACM Press, New York, NY, 1581-1584.

LIKERT, R. 1932. A Technique for the Measurement of Attitudes. *Archives of Psychology* 140, 1-55.

LIPOVSZKI, G. 2004. Simulating Complex Systems in LabVIEW. In *Proceedings of the 16th European Simulation Symposium* (Budapest, Hungary). SCS Press, San Diego, CA, 325-330.

LIVNAT, Y., AGUTTER, J., MOON, S., AND FORESTI, S. 2005. Visual Correlation for Situational Awareness. In *Proceedings of the 2005 IEEE Symposium on Information Visualization*. IEEE Computer Society, Washington, DC, 95-102.

MATHWORKS, INC. 2008. SimEvents 2 Model and Simulate Discrete Event Systems. https://tagteamdbserver.mathworks.com/ttserverroot/Download/38765_91354v01_SimEvents_DS.pdf.

MONROY, C., FURUTA, R., AND MALLEN, E. 2003. Visualizing and Exploring Picasso's World. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries* (Houston, TX). IEEE Computer Society, Washington, DC, 173-175.

OWEN, R. N., BAECKER, R. M., AND HARRISON, B. 1994. Timelines, a Tool for the Gathering, Coding and Analysis of Usability Data. In *Conference Companion on Human Factors in Computing Systems* (Boston, MA). ACM Press, New York, NY, 7-8.

PLAISANT, C., MILASH, B., ROSE, A., WIDOFF, S., AND SHNEIDERMAN, B. 1996. LifeLines: Visualizing Personal Histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Common Ground* (Vancouver, British Columbia, Canada). ACM Press, New York, NY, 221-ff.

SDS (SDS INTERNATIONAL INC.). 2008. EM2000. http://www.em2000.com/.

SHIAW, H.-Y., JACOB, R. J. K., AND CRANE, G. R. 2004. The 3D Vase Museum: A New Approach to Context in a Digital Library. In *Proceedings of the 4th ACM/IEEE-CS Joint Conference on Digital Libraries* (Tucson, AZ). ACM Press, New York, NY, 125-134.

SWAN, R., AND ALLAN, J. 2000. TimeMine (demonstration session): Visualizing Automatically Constructed Timelines. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Athens, Greece). ACM Press, New York, NY, 393.

WOLBER, D. 1998. A Multiple Timeline Editor for Developing Multi-Threaded Animated Interfaces. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA). ACM Press, New York, NY, 117-118.

APPENDIX A

Demographics and Domain Survey
Interactive Timeline System Evaluation

Please fill in / circle value(s) or use an X to indicate your response.

1. Age: ________________

2. Gender: Male / Female (circle appropriate)

4. _Time experience in scenario-based incident management training:

5. Time experience in using the Scenario Timeline System (STS) for scenario-based incident management training:____________

6. Time experience training scenario-based incident management using other computerized means:___________

7. Time experience training scenario-based incident management using other non-computerized means:___________

8. Describe the other means of scenario-based incident management training used (computer based and non-computer based):
____
____
____
____

9. Please circle your level of comfort with computers:

   Not Comfortable 1 2 3 4 5 6 7 Extremely Comfortable

10. Describe the duties or jobs you currently perform or have performed in executing scenario-based incident management training programs:

Program Incident Management Level (e.g. ICP, EOC, MACC)	**Job Description (e.g. Tower, Green Cell Monitor, Role Player, TAF, Floor O/C)**

APPENDIX B

Task Description & Questionnaire
Interactive Timeline System Evaluation

The Event Timeline view presented on the computer screen shows a partially executed sample decision tree that could be considered part of a fictitious training scenario. The view supports the management of timeline elements via mouse dragging, a palette on the right hand side or by "right-click" menus. There is also a filter field at the top. The Command Correlation Timeline is also provided with sample data depicted. Feel free to utilize the system while answering the following questions.

Please circle a number which corresponds to the degree that you concur with the following statements.

1. **The Event Timeline depicts information in a way that is understandable.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*
2. **I can understand the scenario vignettes depicted by the Event Timeline.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*
3. **The Event Timeline clearly depicts the flow of the scenario up to the current time.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*
4. **I can discern the potential paths that a scenario can follow using the Event Timeline.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*
5. **The Event Timeline aids my understanding of a scenario as a whole.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*
6. **I can navigate to areas of interest in the Event Timeline.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*
7. **I examine the Event Timeline frequently during an exercise.**
   1 2 3 4 5 6 7
   Strongly disagree *Neutral* *Strongly agree*

8. **I examine the Event Timeline frequently after an exercise.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

9. **The Event Timeline is useful for producing an After Action Review.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

10. **The Correlation Timeline depicts exercise activity in a way that is understandable.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

11. **I can discern periods of high and low activity using the Correlation Timeline.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

12. **From the Correlation Timeline, I can obtain details about particular activities that have occurred.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

13. **The Correlation Timeline aids my understanding of the activities of others in the exercise.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

14. **The Correlation Timeline and Event Timeline together allow me to correlate injected events to participant responses during exercises.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

15. **The Correlation Timeline aids my understanding of how an exercise is flowing.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

16. **I can quickly navigate to areas of interest in the Correlation Timeline.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

17. **I examine the Correlation Timeline frequently during an exercise.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

18. **I examine the Correlation Timeline frequently after an exercise.**
1 2 3 4 5 6 7
Strongly disagree *Neutral* *Strongly agree*

19. The Correlation Timeline is useful for producing an After Action Review.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

20. Editing scenario injects on the Event Timeline is easy.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

21. I can adjust the times when injects (events) fire and vignettes start in the Event Timeline effectively.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

22. The ability for events to change status on the Event Timeline is useful in managing an exercise.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

23. The Event Timeline effectively allows me to construct a hierarchy of exercise injects and vignettes.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

24. The ability to construct a hierarchy of injects and vignettes is important for running large-scale scenarios.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

25. I can effectively define/manage dependencies among injects in the Event Timeline.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

26. The ability to define/manage dependencies among injects is important for running large-scale scenarios.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

27. The Event Timeline is predictable when running/executing a scenario.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

28. I edit the contents of the Event Timeline regularly during an exercise.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

29. I collaboratively develop scenarios with other authors (place an X in as many as apply):

*Using the timeline to edit the same scenario as others at the same time*________

*Using the timeline to edit the same scenario as others at different times*________

30. The Event Timeline aids my understanding of the activities of other collaborative scenario authors.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

31. I can effectively and collaboratively author scenarios with others.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

32. The Event Timeline allows me to manage large amounts of information effectively.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

33. The Correlation Timeline allows me to understand large amounts of information effectively.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

34. The timeline system as a whole is a distraction from my duties during an exercise.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

35. My duties in running an exercise that presents large-scale incidents would be difficult without the timeline system.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

36. Overall, I am satisfied with the timeline system.

1 2 3 4 5 6 7

Strongly disagree *Neutral* *Strongly agree*

APPENDIX C

Interactive Timeline System Evaluation – Observation Guidelines

Observer: ____

Observation Period: ____

Course: ____

Scenario Description:
____
____
____
____
____

Reminder: The goal of this field study is to obtain insight into the work practices of the users. How do they use the system and for what purposes?

Describe how often the subject uses a timeline, for what and under what conditions:
Correlation Timeline:
____
____

Event Timeline:
____
____

Does the subject edit the Event Timeline: yes / no

What is the most common form or type of editing? Through what means (e.g. properties view, graphical)?
____
____
____

Does the subject use the Correlation Timeline: yes / no
If yes, for what/how?
____
____

How does the subject adjust the times of elements and under what conditions (e.g. precise/properties view, rough/visual dragging, event status changes):

____
____
____

How does the subject navigate to areas of interest and under what conditions (e.g. tabular timeline view, graphical timeline scrolling, filtering, multiple timeline views, mixed methods):

____
____
____
____

Describe the event status change behavior of the subject and the desired intent (e.g. disable, pause, unpause, decision tree management):

____
____
____

Describe any situations where the goal of the user was accomplished using more that a few simple steps (e.g. decision tree management):

____
____
____
____
____
____
____
____
____
____
____
____
____

Did the subject ever seem confused or distracted because of a timeline representation of the scenario? yes / no

Why?____
____
____
____
____
____

Were any timelines used for producing an AAR? yes / no
Under what circumstances?

____
____

Log any actions of interest:

Sim Time of Action	**Action**	**Interest**

Was the subject ever frustrated by any of the timelines? yes / no
If yes, under what conditions?

____
____
____
____
____
____
____

Is there anything that the subject could not do that they wanted to do? yes / no
If yes, what?

____
____
____
____

APPENDIX D

Survey Questionnaire Responses

Responses to the survey questionnaire are presented in Table 4. Items that were not answered by a participant are indicated with "N/A".

Table 4: Questionnaire Participant Responses

Topic	S1	S2	S3	S4	S5	S6	Average	StdDev
Event Timeline Information Depiction	6.20	6.60	6.00	4.80	5.20	6.40	5.87	0.71
1. The Event Timeline depicts information in a way that is understandable.	6	7	6	5	5	6	5.83	0.75
2. I can understand the scenario vignettes depicted by the Event Timeline.	6	6	7	5	5	6	5.83	0.75
3. The Event Timeline clearly depicts the flow of the scenario up to the current time.	6	7	6	4	6	6	5.83	0.98
4. I can discern the potential paths that a scenario can follow using the Event Timeline.	6	7	6	6	5	7	6.17	0.75
5. The Event Timeline aids my understanding of a scenario as a whole.	7	6	5	4	5	7	5.67	1.21
Event Timeline Navigation	5.00	7.00	7.00	7.00	5.00	7.00	6.33	1.03

Table 4 Continued

Topic	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**Average**	**StdDev**
6. I can navigate to areas of interest in the Event Timeline.	5	7	7	7	5	7	6.33	1.03
Event Timeline Frequency of Use	5.00	6.00	3.50	4.00	6.00	7.00	5.25	1.33
7. I examine the Event Timeline frequently during an exercise.	7	7	5	6	7	7	6.50	0.84
8. I examine the Event Timeline frequently after an exercise.	3	5	2	2	5	7	4.00	2.00
Event Timeline AAR Development	6.00	5.00	4.00	4.00	N/A	5.00	4.80	0.84
9. The Event Timeline is useful for producing an After Action Review.	6	5	4	4	N/A	5	4.80	0.84
Correlation Timeline Information Depiction	5.83	5.50	5.33	5.00	N/A	5.83	5.50	0.35
10. The Correlation Timeline depicts exercise activity in a way that is understandable.	5	6	6	6	N/A	6	5.80	0.45
11. I can discern periods of high and low activity using the Correlation Timeline.	6	4	6	4	N/A	5	5.00	1.00
12. From the Correlation Timeline, I can obtain details about particular activities that have occurred.	6	6	6	5	N/A	6	5.80	0.45

Table 4 Continued

Topic	S1	S2	S3	S4	S5	S6	Average	StdDev
13. The Correlation Timeline aids my understanding of the activities of others in the exercise.	6	6	4	5	N/A	6	5.40	0.89
14. The Correlation Timeline and Event Timeline together allow me to correlate injected events to participant responses during exercises.	6	5	7	5	N/A	6	5.80	0.84
15. The Correlation Timeline aids my understanding of how an exercise is flowing.	6	6	3	5	N/A	6	5.20	1.30
Correlation Timeline Navigation	5.00	7.00	2.00	6.00	N/A	4.00	4.80	1.92
16. I can quickly navigate to areas of interest in the Correlation Timeline.	5	7	2	6	N/A	4	4.80	1.92
Correlation Timeline Frequency of Use	3.00	6.00	2.00	2.50	N/A	5.00	3.70	1.72
17. I examine the Correlation Timeline frequently during an exercise.	3	7	1	3	N/A	6	4.00	2.45
18. I examine the Correlation Timeline frequently after an exercise.	3	5	3	2	N/A	4	3.40	1.14
Correlation Timeline AAR Development	5.00	4.00	5.00	4.00	N/A	4.00	4.40	0.55

Table 4 Continued

Topic	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**Average**	**StdDev**
19. The Correlation Timeline is useful for producing an After Action Review.	5	4	5	4	N/A	4	4.40	0.55
Inject Management	6.43	6.43	4.57	5.43	6.00	5.43	5.71	0.72
20. Editing scenario injects on the Event Timeline is easy.	6	5	4	6	5	5	5.17	0.75
21. I can adjust the times when injects (events) fire and vignettes start in the Event Timeline effectively.	7	7	4	5	6	6	5.83	1.17
22. The ability for events to change status on the Event Timeline is useful in managing an exercise.	7	7	5	5	6	6	6.00	0.89
23. The Event Timeline effectively allows me to construct a hierarchy of exercise injects and vignettes.	6	7	6	6	6	5	6.00	0.63
24. The ability to construct a hierarchy of injects and vignettes is important for running large-scale scenarios.	7	7	5	6	7	5	6.17	0.98
25. I can effectively define/manage dependencies among injects in the Event Timeline.	6	6	3	5	6	6	5.33	1.21

Table 4 Continued

Topic	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**Average**	**StdDev**
26. The ability to define/manage dependencies among injects is important for running large-scale scenarios.	6	6	5	5	6	5	5.50	0.55
Exercise Use	7.00	7.00	3.00	4.50	4.50	5.00	5.17	1.57
27. The Event Timeline is predictable when running/executing a scenario.	7	7	5	5	6	6	6.00	0.89
28. I edit the contents of the Event Timeline regularly during an exercise.	7	7	1	4	3	4	4.33	2.34
Collaboration Frequency (yes/no)	0.50	1.00	1.00	0.00	0.50	1.00	0.67	0.41
29a. I collaboratively develop scenarios with other authors using the timeline to edit the same scenario as others at the same time.	0	1	1	0	0	1	0.50	0.55
29b. I collaboratively develop scenarios with other authors using the timeline to edit the same scenario as others at different times.	1	1	1	0	1	1	0.83	0.41
Collaboration	7.00	6.00	5.00	N/A	6.00	5.00	5.80	0.84

Table 4 Continued

Topic	S1	S2	S3	S4	S5	S6	Average	StdDev
30. The Event Timeline aids my understanding of the activities of other collaborative scenario authors.	7	6	4	N/A	6	6	5.80	1.10
31. I can effectively and collaboratively author scenarios with others.	7	6	6	N/A	6	4	5.80	1.10
Scale of Use	6.00	6.00	4.50	6.00	6.00	5.50	5.67	0.61
32. The Event Timeline allows me to manage large amounts of information effectively.	6	6	6	6	6	5	5.83	0.41
33. The Correlation Timeline allows me to understand large amounts of information effectively.	6	6	3	6	N/A	6	5.40	1.34
General Satisfaction								
34. The timeline system as a whole is a distraction from my duties during an exercise.	1	2	1	2	1	4	1.83	1.17
35. My duties in running an exercise that presents large-scale incidents would be difficult without the timeline system.	7	7	6	3	7	7	6.17	1.60
36. Overall, I am satisfied with the timeline system.	7	7	6	6	6	6	6.33	0.52

The following topical statements and questions are provided to more clearly indicate the purpose of the Likert statements that were presented to the subject. They correspond by number to the questionnaire entries.

Event Timeline Information Depiction:

1. How well does the Event Timeline depict information?
2. How well does the Event Timeline depict information at different levels of interest (micro/macro etc.)?
3. How well does the Event Timeline depict how a scenario has unfolded in time (what happened in the past and is currently happening)?
4. How well does the Event Timeline depict how a scenario could progress in the future?
5. How well does the Event Timeline depict the scenario holistically (the big picture)?

Event Timeline Navigation:

6. How well does the Event Timeline allow parts of the scenario to be viewed/navigated?

Event Timeline Frequency of Use:

7. How frequently is the Event Timeline examined during an exercise?
8. How frequently is the Event Timeline examined after an exercise?

Event Timeline AAR Development:

9. How useful is the Event Timeline for analyzing student learning?

Correlation Timeline Information Depiction:

10. How well does the Correlation Timeline depict information?
11. How well does the Correlation Timeline depict levels of activity in the scenario?
12. How well does the Correlation Timeline present details about activities that have occurred in the scenario?
13. How well does the Correlation Timeline present information about the activities of others?
14. How well do the timelines depict information that can be correlated?
15. How well does the Correlation Timeline depict how a scenario has unfolded in time (what happened in the past and is currently happening)?

Correlation Timeline Navigation:

16. How well does the Correlation Timeline allow parts of the scenario to be viewed?

Correlation Timeline Frequency of Use:

17. How frequently is the Correlation Timeline examined during an exercise?
18. How frequently is the Correlation Timeline examined after an exercise?

Correlation Timeline AAR Development:

19. How useful is the Correlation Timeline for analyzing student learning?

Inject Management:

20. How easy can injects be edited?
21. How effectively can inject and vignette times be adjusted?
22. How useful are inject and vignette status changes?

23. How effectively can inject hierarchies by created?

24. How important is inject hierarchy creation for running large-scale scenarios?

25. How effectively can inject dependencies by managed?

26. How important are inject dependencies for running large-scale scenarios?

Exercise Use:

27. How predictable is the timeline while running a scenario?

28. How regularly is the timeline edited during an exercise?

Collaboration Frequency:

29a. Subject uses the Event Timeline to edit the same scenario as others at the same time

29b. Subject uses the Event Timeline to edit the same scenario as others at different times

Collaboration:

30. How well does the Event Timeline support the understanding of other collaborative scenario authors?

31. How well does the Event Timeline allow collaboration?

Scale of Use:

32. How effectively can the Event Timeline support the management large amounts of information?

33. How effectively can the Correlation Timeline support the management large amounts of information?

General Satisfaction:

34. How distracting is the timeline system from accomplishing duties?
35. How difficult would running a large-scale exercise be without the timeline system?
36. How satisfying is the timeline system?

VITA

Austin Christopher Riddle received his Bachelor of Science degree in Computer Science from Texas A&M University (TAMU) at College Station in December 2001. While concurrently working as a Software Developer for the TEES Texas Center for Applied Technology, he received an M.S. degree in Computer Science from TAMU in May 2008. His research interests include computer-human interaction, collaborative workspaces and authoring tools, digital libraries, and knowledge formalization, visualization, and retrieval.

Austin has played a vital role in developing innovative software projects in the domain of emergency response including:

- Interactive emergency response technologies for ambulance and triage centers.
- Data management tools for the purpose of pre-hospital intervention effectiveness research.
- Chemical, Biological, Radiological, Nuclear and Explosive (CBRNE) terrorism incident management training simulations.

Mr. Riddle may be reached at the Texas Center for Applied Technology, 1470 William D. Fitch Pkwy., College Station, TX 77845-4645. His email address is austin.riddle@tamu.edu.