User Interface Lessons Learned from Distributed Simulations

Noah Evens, Brooke Whiteford, Geoff Frank, & Rob Hubal
Technology Assisted Learning Center, RTI International
Research Triangle Park, NC
{evens,brookew,gaf,rhubal}@rti.org

ABSTRACT

The Army is pursuing distance learning strategies to meet Soldiers' lifelong learning needs. One cost-effective approach to providing Soldiers with learning by doing is to develop and distribute simulation training systems. The U.S. Army Signal Center & FT Gordon is leading the implementation of this approach through a series of distributed simulations accessible via its University of Information Technology (UIT) portal.

This paper reports on the many user interface lessons learned from development of a series of distributed simulation systems, mostly but not solely for the Signal Center, developed over the past six years, as well as lessons already learned from several simulation systems currently being developed. These systems are designed to meet contractual requirements that they be downloadable and usable by Soldiers anywhere in the world with an AKO connection within 15 minutes, but able to run in standalone mode.

The paper describes a host of issues dealing with visualization and interactivity, context and usability, navigation and tool use, and policy and technology. The lessons learned include observations, solutions, and suggestions to these issues. The paper provides guidelines for future simulation systems building on a model that considers task affordances and demands, user characteristics, and the nature of the domain.

ABOUT THE AUTHORS

Noah Evens is a web specialist in RTI's Technology Assisted Learning (TAL) Center, with over 17 years experience in print and graphic design, audio production, multimedia, and web-based training. His areas of expertise include simulation, visualization, scripting languages, training analysis, user interface design, and audio engineering. He has been a key figure in the development of RTI's web-delivered training simulations for the military.

Brooke Whiteford is the Program Manager for the UIT simulations with RTI's TAL Center. She was also Project Manager on the Critical Care Nursing web served training application, one the first examples of 3D interactive training delivered via the web. Her background includes graphics design, simulation database development, optimization, and quality assurance.

Geoff Frank is Principal Scientist in RTI's TAL Center. He has led the training analysis of web-delivered simulations for the Army Signal Center. He is a co-author of the Signal Center Masterplan for Lifelong Learning. He is also a member of the IEEE Learning Technology Standards Committee.

Rob Hubal, Senior Research Engineer in RTI's TAL Center, conducts research focusing on development, presentation, and evaluation of learning and assessment materials. He is interested in experimental evaluations of usability, acceptance, and cost-effectiveness of training and assessment systems, and applications to everyday domains.

User Interface Lessons Learned from Distributed Simulations

Noah Evens, Brooke Whiteford, Geoff Frank, & Rob Hubal
Technology Assisted Learning Center, RTI International
Research Triangle Park, NC
{evens,brookew,gaf,rhubal}@rti.org

INTRODUCTION

The U.S. Army Signal Center & FT Gordon has been successful in implementing the Army's lifelong learning education and training strategy (Wilson & Helms, 2003). A part of this strategy involves developing and deploying, through its University of Information Technology (UIT) portal, distributed simulations for learning by doing of technical and tactical skills. In this vein, our group has developed and is developing a series of simulation systems for the Signal Center, to include:

- The 25Q/25P AN/TRC-173B radio terminal set.
- The 25S AN/GSC-52A ground strategic satellite communication station.
- The 25S 85/93 satellite communications systems.
- The 25U set of communications systems in support of FBCB2, including ASIP, SINCGARS, EPLRS, and PLGR.
- The 25Q HCLOS radio.
- The Brigade Subscriber Node (BSN).

These systems are all setup/operations trainers for Signal personnel, that is, their intended users are operations and maintenance personnel and their purpose is for assignment-oriented and sustainment training (Cooper, et al., 2004; Frank, et al., 2003). (See Figure 1 for a screenshot.)

Figure 1. FBCB2 Simulation

We have developed other distributed simulation applications as well. For instance, an early simulation was for a community college system to train critical care nurses in defibrillation techniques (see criticalcare.rti.org). Also, for the Armor School we developed an interactive motor pool maintenance training system for the soldier to identify preventive maintenance indicators, conduct operational checks on a variety of equipment and vehicles, and perform organizational maintenance and troubleshooting (McMaster, et al., 2002; see Figure 2 for a screenshot). We are currently developing an integrated digital systems trainer for the S6, and a satellite communications principles trainer to support common core training for the 25S.

Figure 2. Interactive Motor Pool

For all of our applications we are interested in formal and/or informal evaluations to consider how soldiers use the simulators and feel about using the simulators, and derive lessons learned to feed into subsequent simulations (Hubal, 2005). For these lessons learned, we analyzed videotapes taken of soldiers using one of our simulations, notes taken by observers of soldiers using a different simulation, notes taken from conversations and e-mail with Signal Center personnel, ratings sheets filled out by soldiers and instructors, established user interface (UI) standards, and existing literature. The focus is on UI issues, however, related policy and technology issues are also presented.

The remainder of this paper is organized around our lessons learned. For each, we present evidence from our data analyses supporting the lesson, explain the lesson, and then describe recommendations for following the lesson in future applications.

LESSONS LEARNED

We have categorized lessons learned into four broadly labeled interconnected areas: visualization, usability, navigation/interaction, and technology.

Visualization

This first area encompasses lessons associated with aesthetics, game-based UI, and use of multimedia elements.

Visual aesthetics

Our intent is simulation for training. Aesthetics are important, but should subserve the intent. One sample using the HCLOS simulation, of all thirty soldiers from three different classes during Summer 2005, nearly unanimously agreed that the graphics benefited learning. We stress, though, that we focus not on visual fidelity but on training content.

Clients are increasingly interested in consistency of look and feel across all aspects of the application and between related applications. Mandating a common layout and graphic motif would be of benefit because students would not have to adapt to differing vendors' formatting discrepancies.

Though we understand the motivations underlying this premise, students do not necessarily have difficulty adapting to different formats (Grudin, 2002). Consider that the UI includes colors, shapes, labels, locations, key commands, menus, means for navigation, schemata of use, and metaphors. Commonality or consistency in an interface must thus account for visual, syntactic, semantic, communicative, navigational, conceptual, and physical components (Bailey, 2001; Uhlir, 1988). Not all components affect adaptability equally; as described further below, the nature of the content to be learned, the tasks that the student must perform, and student characteristics all factor into how easily a student can adapt from one application to another. Hence, commonality across applications is desirable to an extent but should not be blindly mandatory. (See www.tradoc.army.mil/tpubs/pams/p350-70-2.doc.)

Game-like interface

Clients and users are increasingly using games, game-based UI, and gaming technology as a reference point, particularly those focused on battle command and operations. Indeed, simulation training developers use many of the same or similar tools to what game developers use, and sometimes render their simulations via a game engine. But training is not a game, it has direct relevance to real-world need, with defined learning objectives. Game features that do not distract from the training effectiveness and that add to enjoyment or usage or training effectiveness might be adopted, but those features are in essence secondary. The learning objectives should remain primary, and UI methods should reflect those objectives. Game features that add to enjoyment or usage or training effectiveness might be adopted, but not those that detract.

While observing a COHORT using the BSN simulation at FT Gordon in Summer 2004, we overheard soldiers commenting on the game-like quality of the simulation. We viewed these comments as both positive and negative. They are positive because they imply realism, suspension of disbelief, and engagement. They are negative because they also hint of lack of seriousness and limitations to the system. In one case, a difficult fault threw the soldier, a sergeant, for a loss. Again, this had positive and negative sides. It was positive in that it forced the soldier to try again and focus on the fault. It was negative in that after about three NOGO outcomes the soldier decided simply to skip the lesson as if the fault, and his failure to understand the cause of the fault, were not important.

The implication is for future developers to incorporate game-based UI features that will engage soldiers, but at the same time strive to make clear to the soldiers that it is a training simulation and not a game that is being distributed.

Audio and hyperlinks

In a distributed simulation the footprint (that is, the size of the program to be downloaded from a central server to the student's computer) needs to be small. In these applications we were required to reduce the footprint so that a soldier could download up to 70 hours of virtual reality based training within a very short time (15 minutes) over a relatively low-bandwidth network (28.8 kbps modem).

One consequence of reducing the simulation footprint is minimization of multimedia elements. For instance, in our distributed simulations, environmental audio is present only when deemed necessary according to the training needs. Soldiers using the HCLOS simulation were equally divided on whether or not additional audio would improve the learning process.

Similarly, help functions within the simulations might benefit from narration. A series of design principles guide how multimedia might be used in distributed simulations (Mayer, 2001). For instance, narrations that signal key steps that the student should take, and that co-occur with animations such as the highlighting that we have incorporated in our acquisition lessons, would be expected to improve learning. Similarly, students learn better when they can control the pace of presentation rather than when they receive a continuous presentation. This we already do in our simulations, however, we use on-screen text and references to a technical manual (TM), and the research shows use of text is not as effective as use of another modality, such as narration.

Also, we might make more use of visuals such as signal flow diagrams and animated schematics. Along these lines we found that a block diagram lesson linking the objects in the 2D diagrams with their 3D environment counterparts helped the students learn how to cable together the FBCB2 components (Frank, et al., 2004). We also observed in our BSN signal flow lesson the power of hyperlinks enabling the student to go from the 2D schematic to the 3D world and 'fly' through the simulation following a signal.

Usability

This second area encompasses lessons associated with access to content, establishing context, and referencing.

Access to content

In these simulations there is a lot of content (deriving from established training objectives) that the student needs to access to achieve proficiency in the lesson. Aside from the 3D virtual environment, the simulations present other content such as signal flow diagrams, block diagrams, patch panels, cut sheets, and simulated computer screens as well as initiation information such as tasks, conditions, and standards in the media best suited to convey that element. For example, 2D is used for representations of specific equipment screens which are available to be manipulated as required by steps in the TM. All of this content is accessible via a tabbed interface. The media type does not imply priority of importance of training content therefore the tabbed interface allows the content to be displayed with equal value.

Our UI design intentionally avoided when possible a multiple windowed approach, as many users tend to lose track of which window is where (thus, perhaps, the lack of use of the TM during skills acquisition) (see www.uiaccess.com/spawned.html). Additionally, using the virtual environment as the interface to 2D elements, though it may represent visual consistency, does not support navigational or conceptual consistency (Hartling, et al., 2005; Nielsen, 1998), and may encumber training by obscuring important content and requiring extra effort by the student to complete the lesson. It is partly for this reason that we incorporate a two-screen solution for our desktop trainers (McMaster, et al., 2002), and have implemented 2D views on different tabs in our later distributed simulations.

Some of this content, particularly reference content, is not often accessed. For instance, once the soldiers read the tasks, conditions, and standards or the status cutsheet data they rarely refer back to this information (though in the 85/93 simulation we are using a reconfigurable cutsheet to force the students' attention). One suggestion is to replace tabs with a single additional modal window for auxiliary content overtop the virtual environment window, though that implies that the virtual environment is always primary when in fact it is only one component of the simulation. Another suggestion that does not conflict with using tabs is to provide content-appropriate access to supportive learning material, such as hyperlinks that build relationships between 3D objects and block diagrams.

A suggestion that we do not like is to display controls alongside the main content area employing drop-down (also called combo) boxes. Drop-down boxes hide detail at the benefit of saving space, with associated UI implications (Schaffer & Sorflaten, 1995). Typically, a drop-down box is used to select from long or established lists, such as U.S. states or military occupational specialties. Unlike a set of tabs, a drop-down box is not directly associated with the content it is controlling, so any relationship between the selection from a drop-down box and its effects in another pane are imposed by the design (Caminos & Stellmach, 2004). Students reluctant or unwilling to explore the UI may miss out on valuable information hidden in drop-down boxes, leading to frustration and making it difficult for them to develop a complete mental map of the available functions (Ceaparu, et al., 2004; Mayer, 2003). Instead, students should be able to readily identify the purpose/contents of the drop-down box and select appropriate options.

In contrast, when the training system specifies the student access one of several equally weighted content screens, we recommend a tabbed interface representing overlapping panes, since it presents an observable, consistent method of content access and context switching, that is easy to use and consistent with implementations in numerous common applications such as Microsoft Excel and Mozilla Firefox. In addition, this approach resembles the approach taken by many gaming systems in requiring the user to escape from the simulation and access content and controls via menus. An additional benefit of the tabbed approach is that the lesson designer can specify the starting order of the tabbed windows. If, for instance, a lesson needs to start with the student at a computer watching a system alarm indicator, the lesson designer can open the lesson with the 2D computer screen tab as the initial view.

These comments get at the key issue of management of screen real estate. Our approach has been to use some expand and contract methods to provide drill-down on details (e.g., the TreeView described below) but generally context-switching methods. We propose leaving it incumbent on the student to refer to content (such as tasks, conditions, and standards) that is needed, as that is part of the learning process.

Establishing context

There are different types of context for the student to maintain. For instance, students progress through four stages of learning, what has been labeled the familiarize, acquire, practice, and validate modes (Frank, et al., 2003). Once the student has selected a mode from the Lesson Manager (our standalone lesson selection application), there is no visual indication of mode other than the title text. However, what functions are available to the student differ across modes. In acquisition but not validate mode, the TreeView (a hierarchical presentation of the graphical elements within the environment) can be used to highlight equipment, likewise in practice but not validate mode, assistance on actions to achieve performance measures can be provided. Our designers have proposed subtle color shifts of the user interface in concert with learning stages, the intent being for the student to develop associations between the UI and the criticality of the training session.

Another type of context relates to the student's progress through the lesson. Once they have used the simulation and have become comfortable with navigating and content lookup, soldiers generally move through lessons smoothly. So a typical proficient soldier that we observed would read the initial troubleshooting or setup description, reference the TM to determine next steps (at least in practice mode, since in acquire mode the steps are guided), navigate to equipment, reference schematics, manipulate equipment using available tools (e.g., a virtual wrench), use help and caution and warning messages, access 2D screens, and ultimately view an after-action report (AAR) of performance measure status.

To facilitate this learning, controls and assistance should be context sensitive, reflecting the content in the virtual environment. For example, the number of buttons can become overwhelming, especially when presented as icons, if they remain static across contexts. (We are talking here about control buttons specific to the UI, which the interface designer can manage, rather than button controls on the simulated equipment, which only the instructional designer can manage.) Only options that truly remain static, such as exiting, should be omnipresent. Buttons should be grouped logically if used, and alternate methods should be considered for infrequent controls, such as hotkey mapping, video rendering, and other like settings. Similarly, help text areas should be used exclusively for dynamic information, providing ready access to lesson references, context-sensitive help, and intelligent tutoring. We have seen proposals for using such areas as a catchall, host to a number of semi-related items listed, say, in a drop-down box. We advocate instead standard means of displaying and accessing contextual information, as we do for supportive content.

Finally, we have identified needs for comparing identical but differently portrayed systems, such as equipment in the virtual environment with its block diagram counterpart, and for visualizing relational systems such as a unit to the network. These comparisons and relationships enable the student to understand how equipment or functions fit into a larger picture. Our current means of supporting relational comparisons is via tab switching. Using this method the students track through the relationship between the 2D block diagram and the equipment in the virtual environment, helping them visualize the abstract concept of signal flow. Furthermore we continue to explore novel methods for the display, comparison, and association of and between seemingly disparate data. Such associations are fundamental to the learning process as they enable the student to develop a relational mental model from which to draw upon when presented with unanticipated situations and environments (Feltovich, et al., 2006).

Referencing

There are design tradeoffs regarding text aesthetics that may affect legibility, such as font (e.g., serif font or not), line spacing, kerning, and justification. However, there is no guarantee that the student will even read presented text. In one discussion between an observer and student who repeatedly received a NOGO (this occurred during Stryker COHORT in the summer of 2003, with the FBCB2 simulation), the observer described performance standards used by the simulation, and referred the student to the TM, which apparently the soldier never referenced. Soldiers had to open a separate window to show the TM, a slightly bothersome requirement. (The FBCB2 simulation was a composite of various TM's. During the running of a lesson the student needed to find the correct TM in the TM directory to get through that section of the lesson. The UI aided this requirement by specifying the TM reference for that particular step in the lesson.) Some soldiers did not use the TM while running acquire lessons, but had no trouble following it in the ensuing practice.

Similarly, we have observed that a text box that requires scrolling contains more text than soldiers are typically willing to read. Finally, a text box used for multiple purposes should be avoided as soldiers become confused as to what they should be reading.

There are also design tradeoffs regarding highlighting (visual cueing). In lessons where the student is acquiring skills (but not lessons where the student is practicing skills) we highlight objects that are selected via a hierarchy displayed within a sidebar that we've labeled the TreeView (see Figure 3). Perhaps surprisingly, there have been no informal complaints of using the obscuring highlight method (that is, using non-transparent highlighting) compared to the non-obscuring version, though the non-obscured version has been not only been complemented, but requested as a feature by other clients. Other highlighting mechanisms are needed to support grouped items, subtle cues, signal tracing, etc.

Figure 3. Near Term Digital Radio Highlighted

The TreeView and associated highlighting seem to be appreciated more by novice than advanced users. Most soldiers who had already run through simulation lessons used keyboard and/or mouse navigation to bring up specific objects or equipment, even if the TreeView is available. When pointed out to soldiers having trouble finding objects they tend to find the TreeView feature and its highlighting useful. In addition we have observed soldiers who use the TreeView to locate components of objects specified in the TM's. When confronted with a new complex piece of equipment and specialized task, the soldier can locate the switch by name in the TreeView, click on the name, and have the system navigate to the highlighted switch. This reduces the amount of hunting and pecking required of the student in the 3D environment.

To track and show progress against performance measures we use pop-up dialog boxes, but many students find these intrusive. A less intrusive, more intuitive method of indicating progress, perhaps within a hideable sidebar or additional tab, is needed, though there is some suggestion (e.g., from the gaming community) that soldiers are motivated by success, implying a ubiquitous progress indicator. To show results of performance at the end of each lesson we present the student with an AAR (Figure 4) with GO/NOGO status, high level errors (e.g., system or time violations), and details that tie student actions to performance measure criteria (Frank, et al., 2004). Though an investigation of formatting and visual presentation is a current area of internal research and development, the data contained in our AAR's present condition is considered a UIT standard for its informativeness.

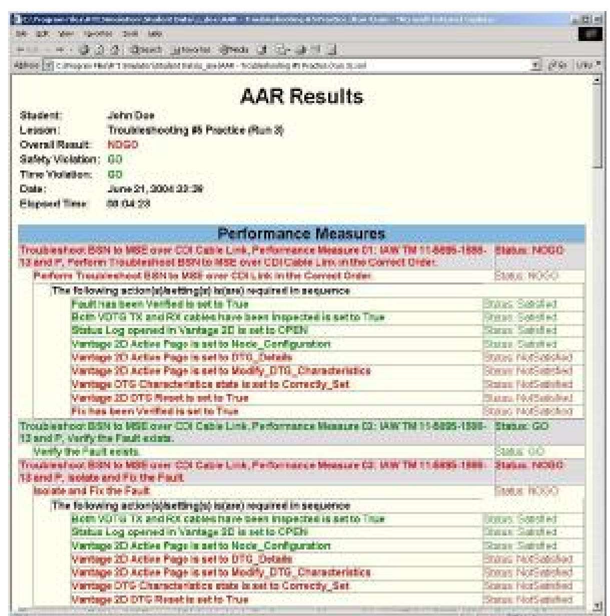

Figure 4. After-action Review Report

Navigation and Interaction

This third area encompasses lessons associated with window controls, virtual navigation, equipment interaction, and tool use and selection.

Window controls

We have identified several window control issues that relate to usability. First, the interactive motor pool application (refer back to Figure 2) had its control menus on the right hand side. These were moved to the left side for UIT projects, a change that is sometimes associated, for left-to-right languages such as English, with increased usability in studies of web pages (e.g., Kingsburg & Andre, 2004). Second, the students had a reasonable expectation that scrollbars should behave like system scrollbars, but those used by the TreeView do not work like system scrollbars and have been criticized. In contrast, the TreeView hierarchy is based on collapse and expand standards. Informal observations of soldiers using other vendors' training applications have exposed similar usability difficulties in areas that deviate from familiar daily applications such as Microsoft Outlook, Word, etc. Third, the ability to resize the window is a feature that most students seem to ignore. Those that do resize (typically maximize) their window appreciate the additional real estate. But since we currently employ a separate window for the TM that the student must access (a contractual obligation, not an architecture constraint), we allow the student control over determining window size rather than force it to full screen.

Virtual navigation

We provide multiple navigation methods for use in the virtual environment, a feature that has been repeatedly praised, but one that differs from games that commonly provide only one or two methods of navigation. Among the navigation methods are free navigation using the mouse and/or keyboard (employed usually by soldiers who have already run through simulation lessons or have gaming experience in navigating 3D worlds); preset navigation using the TreeView; LRU List (a flat summary list of line replacement units); a QuickNav pane that both act like an index into the virtual environment (rated useful by a large majority of soldiers engaged with the HCLOS simulation); and forced navigation during acquire lessons where the student is guided to the location of the contextually relevant objects. We have seen use of all of these methods.

Equipment interaction

Students are required to interact with virtual equipment much as they would in a live environment, but using the mouse and keys to manipulate virtual tools. Thus, students remove and replace cables, check settings, turn knobs to different positions, and in general (for the Signal applications) follow the procedures necessary to set up communications. In support of our implementation, students generally believe that the buttons, switches, cables, and other controls behave as expected.

A good example of equipment interaction is the cable alignment procedure. Subject-matter experts identified cable alignment as a concern in the live environment, where failure to pay attention can lead to bent pins. In the typical case, to remove and replace a cable, the student navigates to the cable, clicks on it with an appropriate tool, retrieves a new cable from a parts bin, and reattaches the cable. For reattachment, we force the student into an alignment procedure where the student must rotate the cable (Figure 5) so that pins do not get bent. Students seem to appreciate the continuous flow; they never have to leave the virtual environment, the procedure is modal (navigation is locked), and a non-invasive textual display is present to help guide them through the process. Indeed, one of the important training aspects of cable alignment is that locking the navigation changes the training tempo and forces the student to pay attention to the alignment process. Soldiers who are fluent in keyboard navigation traverse through the virtual environment very quickly, but the modal nature of cable alignment keeps them focused on the process as the pin alignment highlight rotates. SME's have commented informally that they appreciate the cable animations.

The communications tab has proved challenging because the goal is to train not specifically how to use a telephone or radio, but how to establish communications. For evaluation purposes, it is paramount that we track what the students are saying and to whom. We seem to have abstracted communications interactions from communications equipment, but are open to other implementations. For instance, in order to communicate with a distant unit, the student is required to exit the virtual world in favor of a menuing system providing text boxes allowing selection of which communications device to use, who to call, and what to say. We have considered the use of natural language processing but feel that for distributed simulations, and particularly our small footprint, this option is not yet viable. An interim solution we have explored is the use of an 'instant message' style box which overlays the virtual environment, not dissimilar to the style of communication between avatars in the massive multiplayer environment There (see www.there.com).

Figure 5. Cable Alignment

As mentioned, soldiers are required in some lessons to tab to 2D panes that show screens for specific equipment, and manipulate those screens (such as depressing button controls and entering text) as they would on actual equipment. We employ 2D screens because we've found in previous applications that rendering these changes in 3D to an acceptable resolution is difficult (Hubal, 2005). We've noticed no difficulties on the part of soldiers in interacting as normal with the content of 2D screens.

Tool use and selection

We provide tool icons in the simulations that change the cursor and are meant for actions such as inquiry and inspection (see Figure 6). However the icons and tool functions can be confusing to students. For instance, we intended inquire (third from top) to mean *What is this?* and go-to (third from bottom) to zoom in on an object, but the icons resemble each other. Similarly, in other applications, we noticed students using the hand icon to manipulate cables when a wrench is required, even though actually a remove/replace task might be manageable manually. (As an aside, a hand icon historically signifies drag or grab; we've extended it to signify push or turn, as a pushbutton or knob. Also, we've seen recommendations for use of the hand icon as a selection indicator, though precedent would suggest a finger cursor instead.)

Figure 6. BSN Toolbar

We have generally employed, and have seen specifications mandating, distinct navigation mechanisms and selection mechanisms (i.e., interaction with simulation objects). Forcing the student to switch between navigation and selection is sensible from an implementation perspective but counterintuitive, as in the 'real world' individuals freely scan the environment and naturally accomplish interactions (Bowman, et al., 2001). Individuals do not highlight a selected object nor obscure it with encumbrances of maneuver before selecting it; instead they move and manipulate fluidly. A reasonable approach is to intelligently switch between navigation and selection, using the cursor and the position of content objects as indicators of the current mode, as is the case with finding and interacting with hot spots.

Technology

This fourth area involves UI design decisions associated with technology advancement and policy.

Technology advancement

We have been developing these distributed simulations since 2001, and in that time have seen great increases in computing power and screen real estate. So we recommend definitions of a target frame rate and control response time, and minimum and maximum window and frame sizes. Nevertheless, there is still a wide variety of target platforms for these training simulations, and we are required to design for the lowest common denominator. Application demands might influence these definitions, but they themselves can influence development by helping to specify screen layout, simulation fidelity, and available tools.

We have also seen an increase in use of markup languages to format data to facilitate compatibility and reuse. Though we encourage this trend, and have been responsible for packaging our simulations as shareable content objects (i.e., SCORM-compatible), we wish to make clear we feel a common externally defined UI (sharing markup language formats) will not so much help establish a standard for data sharing between vendors and promote interoperability, but that it is the application content that drives data sharing. That is, we advocate the separation of UI from application content, and do not support use of UI standards to influence components of training or assessment.

Policy

We are engaged in standards committees focused on information technology for learning, education, and training (see jtc1sc36.org) and SCORM simulation interfaces (see ieeeltsc.org/wg11CMI/cmi-sim/). These working groups are beginning to consider UI issues related to distributed simulations. Other published standards (such as ISO standards on human-centered design processes for interactive systems, software ergonomics for multimedia user interfaces, and ergonomics of human-system interaction) address relevant topics (e.g., multimedia navigation and control, usability methods), including some topics related to our simulated environments.

We have also used the Army's multimedia courseware development guide (see reference above to 350-70) and its interactive multimedia instruction implementation standards to level IV. Though simulation and gaming appear to be significant departures from the interactive multimedia that these documents were designed to address, to our knowledge the soldiers have registered very few complaints about the interface directly, tending to focus on training issues such as how they are being evaluated and slight differences in procedures not encapsulated in the simulations. To maintain this success, a final lesson learned is to consider mid- and post-project usability and acceptance testing and possibly even early project focus groups.

MODEL-BASED GUIDELINES

At one point during our observations we heard the question "You guys learning or playing?" and the response "Learning, Sergeant.". Later, an instructor told us that soldiers who used live equipment for a capstone training exercise were much more comfortable in the real vehicle after working with the simulation, that they commented on the similarity between the simulation and vehicle, and that soldiers started out much more competently on those tasks that were done in common between the simulation and the vehicle (cf. Hubal & Helms, 1998).

The point here is that the UI sufficiently engaged the soldiers and presented an environment sufficiently similar to the real environment to enable acceptance and accessibility. We present three dimensions that should be considered in implementing the recommendations of this paper while developing distributed simulations.

Task Demands

Improved processors and graphics have enabled greatly increased modeling realism, however, increased appearance fidelity in the 3D world is just one important characteristic of our simulations. Additionally, we've focused on tools, interactions, navigation, and performance measures that reflect the demands placed on the student by the skill to be learned.

In choosing appropriate UI elements for a distributed simulation, based on our lessons learned, we recommend that the developer consider (i) whether or not a simulated environment is necessary for the training, (ii) how realistic the environment needs to appear to ensure transfer of training to a live environment, (iii) how the students should navigate through the environment so that navigation is easy to learn and easy to use, (iv) what common tools should be made available, or even that they are necessary to be made available, and (v) how the student would normally need to access reference material and how to present that material without interfering with learning.

User Characteristics

Improved systems have also enabled greater choices for adapting to students. For instance, the systems have presented increasing control to soldiers in ways to navigate and interact with simulation objects. Similarly, basic functions such as audio feedback, window resizing, and control settings for display (e.g., rendered in software or hardware, frame rate, perhaps even color schemes) should be given to students to adapt to the varied platforms that would be expected within distributed systems. Even better, since users may not alter adjustable settings, we recommend an adaptive application that accesses content appropriate for a given hardware platform (our IMP simulation had a capability to check for client system capabilities), user level, competency measurement, etc.

In addition, tools and procedures that work in a live environment under specific conditions need to work under those same conditions in a simulated environment (with the caveat that contractual or implementation reasons may prevent highly realistic simulation of some procedures, such as communications in our simulations). Also, means of access to reference material and performance measures should be readily granted, or relationships among performance measures, student actions, and reference material inferred via metadata, as different students will make use of these sources of information differently as they learn the requisite knowledge and skills. (Again, there may be restrictions, such as being contractually prevented from modifying reference materials to make them more usable.) These choices address the issue of user characteristics, since whenever possible it should be up to the user to choose modes and means of interaction (Stanney, Mourant, & Kennedy, 1998).

Nature of the Domain

Aspects of the domain being simulated can also affect the UI. For instance, the procedural nature of many of the signal systems that we simulated allowed for presentation of appropriate equipment and tools, ease of navigation in familiarization and acquisition because the next step at any point was specified, and reasonably easy access to reference material such as the TM. Knowing that the procedures also involve physical manipulation, and understanding the limitations imposed by distributing the system (hence, an inability to develop any sort of immersive simulation training system), led to UI features such as a changing cursor based on the tool selected and special animations such as cable alignment that demand important consideration by the student.

Hence, we recommend that future developers consider how the domain itself might influence design decisions. A simulation training system must reflect the differing tasks demanded of and the differing objects of interest within different domains.

ACKNOWLEDGEMENTS

We wish to thank the many coworkers who have helped in UI design decisions and in observation and testing of our distributed simulation training systems. A partial list includes Alan Amsden, Randy Brown, George Cooper, Chip Hill, Jeff Johnson, Patti Krizowski, Kevin Merino, Hal Waters, Jan White, and Geoff Yarbrough. The opinions expressed here do not necessarily represent the official positions of RTI or of any of the sources of funding for these distributed simulation training systems.

REFERENCES

Bailey, B. (2001). Web site consistency. User Interface Update Newsletter, Human Factors International, May, 2001.

Bowman, D., Kruijff, E., LaViola, J., & Poupyrev, I. (2001). An introduction to 3-D user interface design. PRESENCE: Teleoperators and Virtual Environments, 10(1), 96-108.

Caminos, R., & Stellmach, T. (2004). Cross platform user interface design. Gamasutra, March 26, 2004.

Ceaparu, I., Lazar, J., Bessiere, K., Robinson, J., & Shneiderman, B. (2004). Determining causes and severity of end-user frustration. International Journal of Human-Computer Interaction, 17, 333-356.

Cooper, G., Brown, R., Whiteford, B., Frank, G., Perkins, K., & Lizama, J. (2004). Embedded distributed training: Combining simulations, IETMs, and operational code. Proceedings of the 26th I/ITSEC (pp. 205-213), Orlando, FL.

Feltovich, P.J., Prietula, M.J., & Ericsson, K.A. (2006). Studies of expertise from psychological perspectives. In K.A. Ericsson, N. Charness, P.J. Feltovich, & R.R. Hoffman (Eds.), Cambridge Handbook of Expertise and Expert Performance. New York, NY: Cambridge University Press.

Frank, G., Whiteford, B., Brown, R., Cooper, G., Evens, N., & Merino, K. (2003). Web-delivered simulations for lifelong learning. Proceedings of the 25th I/ITSEC (pp. 170-179), Orlando, FL.

Frank, G., Whiteford, B., Hubal, R., Sonker, P., Perkins, K., Arnold, P., Presley, T., Jones, R., & Meeds, H. (2004). Performance assessment for distributed learning using after action review reports generated by simulations. Proceedings of the 26th I/ITSEC (pp. 808-817), Orlando, FL.

Grudin, J. (1992). Consistency, standards, and formal approaches to interface development and evaluation. ACM Transactions on Information Systems, 10(1), 1164-1173.

Hartling, P.L., Bierbaum, A.D., & Cruz-Neira, C. (2005).Tweek: Merging 2D and 3D interaction in immersive environments. Journal of Systemics, Cybernetics and Informatics, 2(6).

Hubal, R. (2005). Design and usability of military maintenance skills simulation training systems. Proceedings of the 49th Human Factors and Ergonomics Society Conference (pp. 2110-2114), Orlando, FL.

Hubal, R.C., & Helms, R.F. (1998). Advanced learning environments. Modern Simulation & Training, 5, 40-45.

Kingsburg, J.R., & Andre, A.D. (2004). A comparison of three-level web menus: Navigation structures. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 48, 1513.

Mayer, R.E. (2001). Multimedia learning. New York: Cambridge University Press.

Mayer, R.E. (2003). Learning and instruction. Upper Saddle River, NJ: Prentice Hall.

McMaster, L., Cooper, G., McLin, D., Field, D., Baumgart, R., & Frank, G. (2002). Combining 2D and 3D virtual reality for improved learning. Proceedings of the 24th I/ITSEC (pp. 246-254), Orlando, FL.

Nielsen, J. (1998). 2D is better than 3D. Alertbox, November 15, 1998.

Schaffer, E., & Sorflaten, J. (1995). Pull-down menus: Out of sight, out of mind. The X Journal, November/December.

Stanney, K.M., Mourant, R.R., & Kennedy, R.S. (1998). Human factors issues in virtual environments: A review of the literature. PRESENCE: Teleoperators and Virtual Environments, 7, 327-351.

Uhlir, S. (1988). Enabling the user interface. IBM Systems Journal, 27(3), 306-314.

Wilson, W.R., & Helms, R.F. (2003). A business model for lifelong learning. Proceedings of the 25th I/ITSEC (pp. 1089-1098), Orlando, FL.