Proceedings of the Fourth Annual GIFT Users Symposium

May 2016

Edited by:
Robert Sottilare
Scott Ososky

Part of the Adaptive Tutoring Series

Proceedings of the 4th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym4)

Edited by:
Robert Sottilare
Scott Ososky

Printed in the United States of America
First Printing, July 2016

U.S. Army Research Laboratory
Human Research & Engineering Directorate
Orlando, Florida

International Standard Book Number: 978-0-9977257-0-4

We wish to acknowledge the editing and formatting contributions of Carol Johnson and Deeja Cruz

Dedicated to current and future scientists and developers of adaptive learning technologies

CONTENTS

From the Editors **1**

Theme I: GIFT Research **5**

Elements of a Learning Effect Model to Support an Adaptive Instructional Framework 7

Automated Detection of Cognitive and Metacognitive Strategies for Learner Modeling in GIFT 15

Motivational Feedback Messages as GIFT Interventions to Frustration 25

The Use of Social Media for Creating and Improving Learning Content 37

The Hidden Challenges of Team Tutor Development 49

Theme II: GIFT Feature Development **61**

The GIFT 2016 Community Report 63

Using GIFT Wrap to Author Domain Assessment Models with Native Training Applications 75

Extending GIFT with a Reinforcement Learning-Based Framework for Generalized Tutorial Planning 87

Integrating an Interoperable Competency Model with GIFT using the Experience API (xAPI) 99

An Adaptive AAR Capability for GIFT 111

Theme III: Practical Applications of GIFT 123

From Concept to Publication - A Successful Application of Using GIFT from the Ground Up 125

The Updated Research Psychologist's Guide to GIFT.......................................135

Designing the User Experience of the GIFT Cloud authoring tools 145

From the Editors

We are proud of what we have been able to accomplish with the help of our user community. This is the fourth year we have been able to capture the research and development efforts related to the Generalized Intelligent Framework for Tutoring (GIFT) community which at the writing of these proceedings is now about 800 users in 52 countries. These proceedings are intended to document the evolutions of GIFT as a tool for the authoring of intelligent tutoring systems (ITSs) and the evaluation of adaptive instructional tools and methods. Papers in this volume were selected with the following goals in mind:

- The candidate papers describe tools and methods that raise the level of knowledge and/or capability in the ITS research and development community
- The candidate papers describe research, features, or practical applications of GIFT
- The candidate papers expand ITSs into previously untapped domains
- The candidate papers build/expand models of automated instruction for individuals and/or teams

The editors wish to thank each of the authors for their efforts in the development of the ideas detailed in their papers. As a community we continue to move forward in solving some significant challenges in the ITS world. For future efforts, we would like to list some of these challenges here:

- Unobtrusive learner data acquisition to support individual learner and team state classification
- Optimal selection of ITS strategies and tactics for individual learners and teams
- Assessment of individual and team learning, performance, retention, and transfer of skills from training and education environments to work environments
- Efficient ITS authoring experiences including methods to organize domain knowledge
- Automation of elements of the authoring process to reduce the time to produce adaptive instruction

We would also like to encourage readers to follow GIFT news and publications at www.GIFTtutoring.org. In addition to our annual GIFTSym proceedings, GIFTtutoring.org also includes volumes of the Design Recommendations of Intelligent Tutoring Systems, technical reports, journal articles, and conference papers. GIFTtutoring.org also includes a users' forum to allow our community to provide feedback on GIFT and influence its future development.

THEME I: GIFT RESEARCH

Elements of a Learning Effect Model to Support an Adaptive Instructional Framework

Robert Sottilare
U.S. Army Research Laboratory

INTRODUCTION

This paper describes the evolution of a learning effect model (LEM; Sottilare, 2012; Sottilare, Ragusa, Hoffman & Goldberg, 2013) to guide adaptive instruction within the Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden, 2012), an open-source architecture for authoring, delivering, guiding, and evaluating tailored, computer-based instruction for individual learners and teams of learners. Effect models may be used to demonstrate the relationship between the learner, the computer-based tutor, and the instructional environment (Figure 1) and how they influence each other during adaptive instruction.

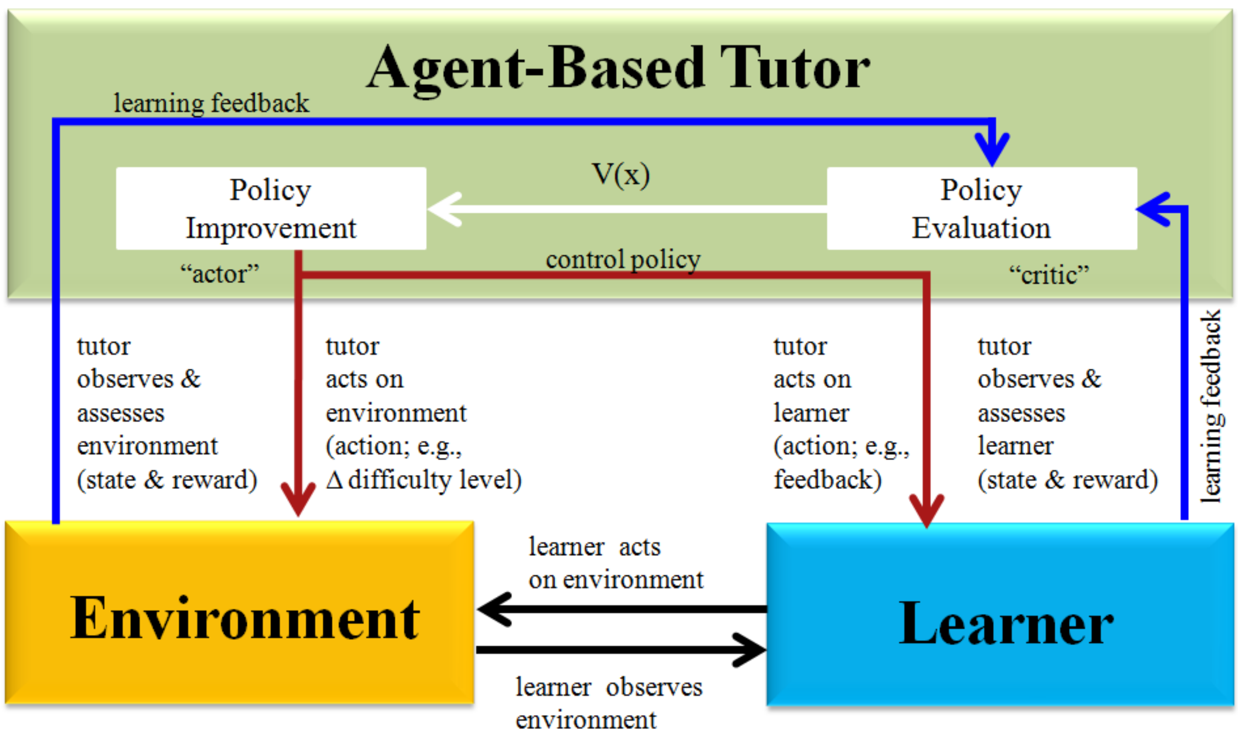


Figure 1: Interaction between the learner, the tutor, and the instructional environment

The GIFT LEM is focused on how these relationships facilitate desirable learning outcomes (e.g., knowledge and skill acquisition, performance, retention, and transfer of skills from training to the work environment). The LEM discussed in this paper began as a set of strategies (plans) and tactics (actions) used by a GIFT-generated tutor to drive tailored training experiences in real-time. The LEM has since evolved to encompass real-time and long-term models of both individual learners and teams of learners plus required knowledge, learning objectives, tailored learning events, and measures of learning and performance. This paper provides a detailed breakout of elements and processes that compose the LEM and a description of their function in the process of adaptive instruction.

Figure 2 illustrates the elements of the LEM and their interaction. The model is composed of three distinct processes: pre-tutoring, real-time instruction, and post-tutoring phases. Each is described in detail in the following sections of this paper.

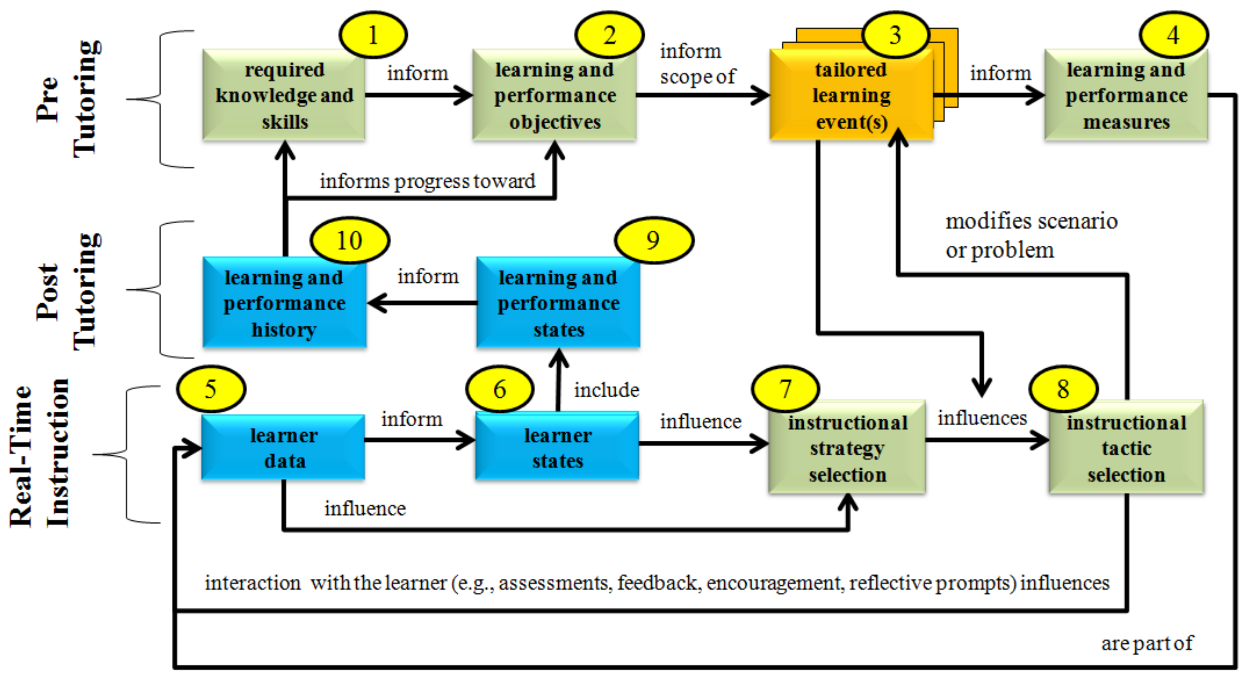


Figure 2: Updated Learning Effect Model

PRE-TUTORING PHASE

The pre-tutoring phase includes functions necessary to support authoring and initialization of the tutor. The author selects the tutoring domain, and defines the required knowledge and skills for a course or lesson. When the tutor initializes, a quick comparison of the learner's domain knowledge and skills with the required knowledge and skills identifies a set of learning gaps that drive tailored learning and performance objectives for the upcoming tutoring experience. This aids in narrowing the scope of the content that will be presented to the learner during the real-time instructional phase. The learner's goals (exploration, formal learning, or refresher training) are also considered in this phase and a tailored instructional event is created to expose the learner to new content while anchoring new content to old learning experiences. This historical information is stored in a long-term learner model in what is generally referred to as a record store. Finally, the author identifies associated learning and performance measures along with sources for these data to determine the learner's progress toward objectives during real-time instruction.

While this is not an exhaustive list, it is a list of common measures or variables of interest which influence the adaptiveness of the tutor and its perception of the learner and the training environment. Measures are determined by the tutor author during the pre-tutoring phase and may include, but are not limited to, the following:

- Current learner states
    - domain competence or prior knowledge
    - engagement performance
    - learning (knowledge and skill acquisition)
    - emotional states which influence learning (e.g., boredom, frustration, joy, confusion)
- Concepts under instruction
- Course flow and hierarchical relationships between concepts
- Progress toward learning objectives
- Learner data sources
    - Learner input
    - Behavioral and physiological sensors
    - Learner records
- Assessment methods (data interpretation and state projection)
- Available tutor actions (feedback, changes to tutoring environment)
- Reward functions associated with available tutor actions
- Minimum standards and other measures

In the next section, we discuss elements within the LEM that support real-time adaptive instruction.

REAL-TIME INSTRUCTIONAL PHASE

As with all phases of instruction, the real-time instructional phase of tutoring is managed by GIFT through the LEM (**Figure 2**). Data collected from the learner or the learner's record store (historical data including experiences, achievements, and demographics) are used by GIFT to assess/predict learner states (e.g., performance, learning, emotions, engagement). Learner data may also include learner traits (e.g., personality, educational level), which can also be used standalone to trigger adaptations by the tutor.

In the previous section we discussed the importance of identifying measures during the authoring process. A key set of measures centers on learner characteristics as a basis for adaptation decisions. Most intelligent tutoring systems (ITSs) use performance and achievement as the primary measures to trigger adaptation, but there are many more important attributes that should be considered based on their influence on learning, performance, retention, and transfer. Depending upon the ease with which measures can be captured and states classified, the following additional learner characteristics should be considered: working memory capacity, prior knowledge of the domain under tutoring, and current emotional state.

Once we have identified sources of adaption, we can begin to link them to targets of adaptation. According to Goldberg et al (2012), target adaptations might include changes to the sequence of instruction, presentation of information, degree of learner control, feedback frequency or content, task complexity, or the pace of instruction. GIFT uses learner data and states to select optimal strategies or plans for action.

Instructional strategies within GIFT are domain-independent plans for action and may be associated with course navigation decisions (e.g., mastery of concept A achieved; okay to move to concept B) or intervention decisions (e.g., tutor feedback, prompts, questions, or changes to scenario difficulty). The tutor’s selection of specific actions or tactics are influenced by the strategy selection and bounded by the conditions of the scenario at the time the intervention decision by the tutor is triggered. Course navigation decisions in GIFT are largely driven by Merrill’s Component Display Theory (CDT; 1983).

Figure 3 illustrates course navigation decisions by GIFT-based tutors for a lesson consisting of 3 concepts. First, the learner is guided through the rules quadrant, where they are exposed to the principles for the domain under tutoring (e.g., hemorrhage control). We assume no hierarchical dependency between the concepts being tutored, so they can be learned in any order. As the rules for one concept are reviewed, the learner is guided to other concepts (green arrows). Mastery based on preliminary *checks on learning* can be inserted prior to decisions to move forward to new concepts. Low scores for checks on learning or off-task behavior (e.g., rapidly clicking through material) can result in being redirected to new material on the current concept instead of moving on to new concepts (red arrows). Completing the reviews for all of the concepts’ rules results in transition to the examples quadrant.

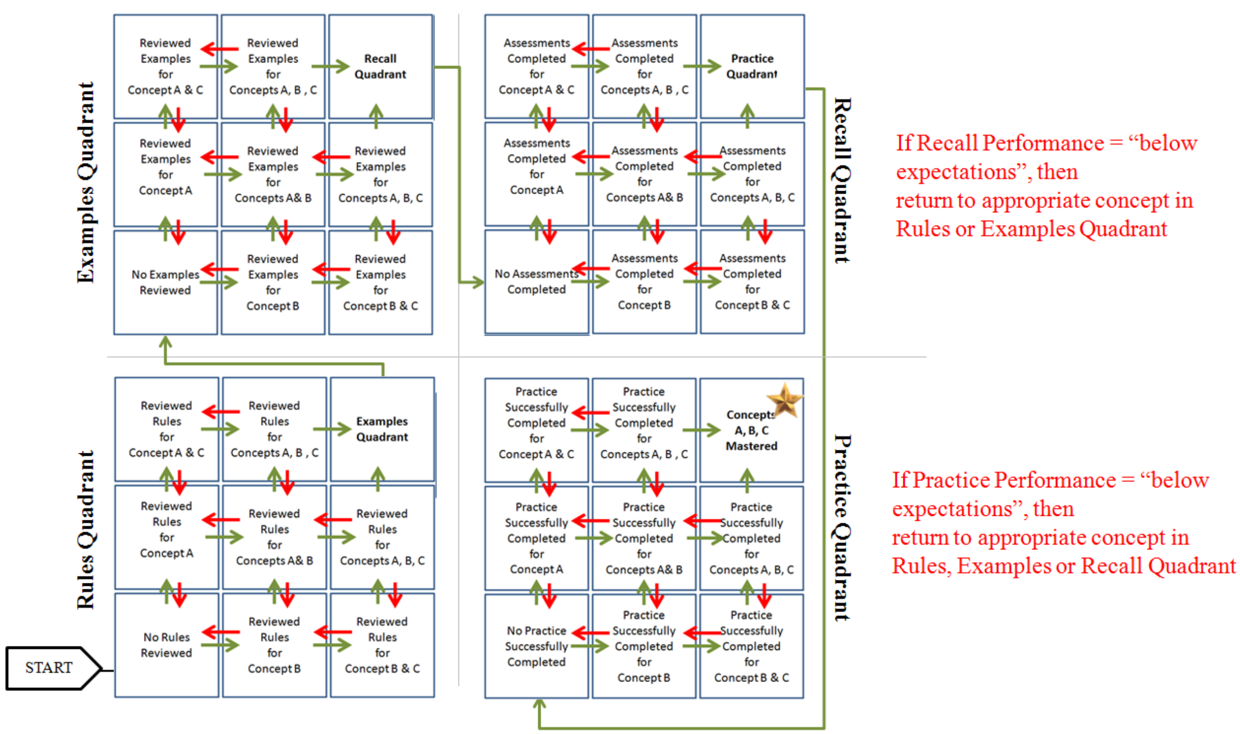


Figure 3: Adaptive Course Navigation in GIFT based on CDT and a lesson consisting of three concepts or learning objectives (A, B, and C)

As with rules, successful examples are reviewed and may also contain preliminary checks on learning. If the examples for all the concepts are successfully reviewed, the learner is prompted to move on to the recall quadrant where more substantial assessments of their domain knowledge are conducted. If the recall quadrant is successfully mastered, the learner moves to a practice environment to apply their knowledge and exercise their skills. If they do not perform to standard in the recall quadrant on any of the concepts, the learner is redirected to either the rules or examples quadrant for targeted remediation for only the underperforming concepts (see red arrows and text in figure).

The practice quadrant is focused on skill development which is generally tracked through the learner’s behaviors (e.g., decisions, actions). Some tutors differentiate between the behaviors of learner and an expert model. Expert models trace the actions of experts to define various paths of success. For example, if we tracked the behaviors of writers for an essay writing tutor, most writers would begin with an outline of their writing project and then expand details under each heading. Others might write a short synopsis and use this as a basis for developing an outline. Still others might develop goals for the story prior to writing. All of these methods might be considered effective by experts so multiple paths leading to success can be developed by the author as shown in the expert model (**Figure 4**).

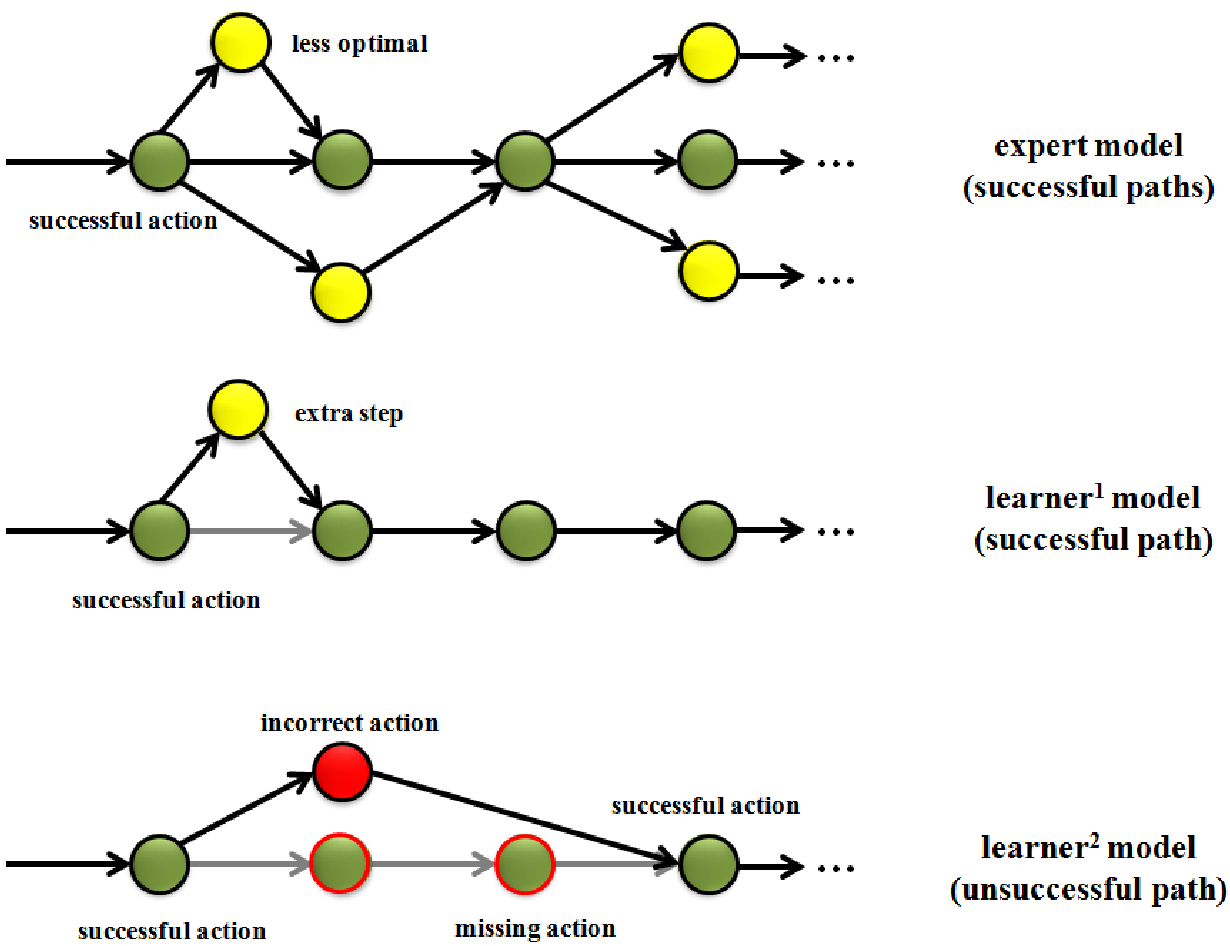


Figure 4: Directed Graph for Course Navigation showing multiple successful paths (expert model) and learner models with successful and unsuccessful paths

Examining the expert model diagrammed in the directed graph in Figure 4, we note one optimal path (straight line) and additional paths that may be viable but are sub-optimal due to extra steps. Examining the first learner model (middle graph) we see an direct imitation of an expert performance, but with one extra (unneeded) step. Finally, in examining the performance of the second learner (bottom graph) we see the learner execute an unsuccessful path with an incorrect action and a missing action. The path is unsuccessful not just because of the errors, but because the errors are significant or critical to success. If the errors were minor, the path might be deemed to be successful. Note that the unsuccessful path can contain successful actions and successful paths can have minor errors.

The conditions within the tailored learning event (**Figure 2**) at the time of assessment by the tutor along with the learner's behaviors (**Figure 4**) and the instructional strategy selected (**Figure 2**) based on the learner's states and traits determine the tutor's tactical selection. These elements along with reinforcement of past successful tactics drive future tactic selection (Equation 1).

Tactic Selection = f(environment conditions, learner behaviors, instructional strategy selection, past tactic selections) (1)

Instructional strategy selections are based on learner states and traits, and reinforcement of past successful strategy recommendations (Equation 2).

Instructional Strategy Selection = f(learner states, learner traits, past strategy recommendations) (2)

Resulting tactics (successful or unsuccessful) also drive intended and unintended changes in the learner's behavior and physiology (**Figure 2**) which may influence stress levels or motivation. These changes can result in accelerated or decreased progress toward learning objectives.

POST-TUTORING PHASE

As discussed in the previous section, the learner is part of an interactive system and may be positively or negatively affected by changes or interactions that occur within the system. Since a primary goal of ITSs is to adapt and guide the learner to progress toward their learning and performance objectives, tracking the learner's achievements of these objectives is important in evaluating the performance and effectiveness of the tutor in the post-tutoring phase. While evaluation of the tutor may occur on a continuous basis, cumulative data provide insight into both the learner's tutoring experience and the areas in need of improvement within the curriculum content. With this in mind, we recommend the examination of learner data sources and tutor decisions to support:

- long-term modeling of learner attributes to identify domain competencies
- understanding of learner habits and trends to enable more efficient future adaptation by the tutor
- discovery of paths to achievement and misconceptions within a domain of instruction

FUTURE DIRECTIONS

As noted in Equation 1 and Equation 2, several variables combine to influence strategy and tactic selection within the LEM and GIFT-based tutors. The scope of influence of these variables on tutor selections is not well understood. Future research should focus on discovering the behavior and sensitivity of these variables with respect to tutor decisions and their influence on each other. Data-mining techniques should be employed to capture ITS performance data with respect to learning, retention, performance, and transfer of skills from instructional environments to work/operational environments. Finally, effort should be focused on understanding the influence of learner and environmental variables across various instructional domains and domain taxonomies (i.e., cognitive, affective, psychomotor, and social).

ACKNOWLEDGMENTS

The research described herein has been sponsored by the U.S. Army Research Laboratory, Human Research & Engineering Directorate (ARL-HRED). Statements and opinions expressed in this paper do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

REFERENCES

Goldberg, B., Brawner, K. Sottilare, R., Tarr, R., Billings, D., & Malone, M. (2012). Use of Evidence-based Strategies to Expand Extensibility of Adaptive Tutoring Technologies. In Proceedings of the Interservice/Industry Training Simulation & Education Conference, Orlando, Florida, December 2012.

Merrill, M.D. (1983). Chapter 9: Component Display Theory. In Charles M. Reigeluth (Ed.) Instructional Design Theories and Models: An Overview of Their Current Status.

Sottilare, R. (2012). Considerations in the development of an ontology for a Generalized Intelligent Framework for Tutoring. International Defense & Homeland Security Simulation Workshop in Proceedings of the I3M Conference. Vienna, Austria, September 2012.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Sottilare, R., Ragusa, C., Hoffman, M. & Goldberg, B. (2013). Characterizing an adaptive tutoring learning effect chain for individual and team tutoring. In Proceedings of the *Interservice/Industry Training Simulation & Education Conference*, Orlando, Florida, December 2013.

ABOUT THE AUTHOR

Dr. Robert A. Sottilare *leads adaptive training research within the US Army Research Laboratory where the focus of his research is automated authoring, automated instructional management, extension of tutors to psychomotor tasks, and evaluation tools and methods for intelligent tutoring systems. His work is widely published and includes articles in the Cognitive Technology Journal, the Educational Technology Journal, and the Journal for Defense Modeling & Simulation. Dr. Sottilare is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT), an open-source tutoring architecture, and he is the chief editor for the Design Recommendations for Intelligent Tutoring Systems book series. He is a visiting scientist and lecturer at the United States Military Academy and a graduate faculty scholar at the University of Central Florida. Dr. Sottilare received his doctorate in Modeling & Simulation from the University of Central Florida with a focus in intelligent systems. In 2012, he was honored as the inaugural recipient of RDECOM's Modeling & Simulation Lifetime Achievement Award, and in 2015 he was honored with the National Training and Simulation Association (NTSA) Governor's Award for Modeling & Simulation Lifetime Achievement.*

Automated Detection of Cognitive and Metacognitive Strategies for Learner Modeling in GIFT

Michael Tscholl[1], Gautam Biswas[1], Benjamin S. Goldberg[2], and Robert A. Sottilare[2]
[1]Institute for Software Integrated Systems, Department of Electrical Engineering & Computer Science, Vanderbilt University, [2]U.S. Army Research Laboratory

INTRODUCTION

Promoting students' learning of cognitive and metacognitive strategies that may generalize across domains is increasingly seen as an important component of intelligent tutoring systems (ITSs), especially those that support open-ended complex problem solving and decision making. Such open-ended learning environments (OELEs) allow learners to make choices in their approach to developing, monitoring, and managing their evolving solution paths (Segedy, Kinnebrew & Biswas, 2015). To be successful, learners have to become adept at employing cognitive, metacognition and self-regulation processes and strategies in developing their solutions (Butler & Winne, 1995; Kinnebrew, Segedy & Biswas, 2016; Zimmerman & Schunk, 2001). Such processes and strategies typically encompass information acquisition, situation awareness, plan development and refinement taking into account resource limitations and trade-offs, solution monitoring, evaluation, and, finally, reflection.

Research on learning with OELEs has produced mixed results. Students may show large learning gains, but may also experience frustration from the inability to manage the complexity of task (Basu & Biswas, 2016; Segedy, Kinnebrew & Biswas, 2015). Therefore, a key to successful learning in OELEs is providing scaffolds and support that is adapted to students' proficiency and learning behaviors. Detecting students' proficiency and learning behaviors is essential to helping them develop effective learning strategies (Goldberg & Spain, 2014; Basu & Biswas, 2016).

In a project supported by the U.S. Army Research Laboratory (ARL), we have been designing a metacognitive tutoring framework for the Generalized Intelligent Framework for Tutoring (GIFT), "a computer-based tutoring framework to evaluate adaptive tutoring concepts, models, authoring capabilities, and instructional strategies across various populations, training tasks and conditions" (Goldberg & Cannon-Bowers 2013; Sottilare, et al. 2012; Sottilare & Holden 2013). GIFT provides three primary services for instructional system designers and developers: (1) tools that support authoring of tutoring system content, which includes domain concepts and remedial instruction modules, (2) management of instructional processes that emulate the practices of human tutors, and (3) an assessment methodology to evaluate the effectiveness of the tutoring system and its components (Sottilare et al., 2012). Our goal is to extend the domain knowledge module to include metacognitive and self-regulation processes and strategies, and develop methods for parsing and analyzing learners' action sequences within a training environment to derive their learning behaviors and map them onto known processes and strategies. We will also extend the learner modeling in GIFT to capture a more continual and fine-grained assessments of learners' capabilities, and then use these assessments to provide adaptive scaffolding and feedback to learners as they work on their problem-solving tasks.

In this paper, we present our work on modeling students learning about counterinsurgency (COIN) operations with UrbanSim (McAlinden, et al., 2009), a turn-based game environment, where users take on the role of a battalion commander to deal with fictional counterinsurgency scenarios. We track student problem solving and analyze student performance using the extensions of the GIFT tracking and learner modeling capabilities that we are implementing to develop metacognitive tutoring in GIFT. The analysis of turn-by-turn student performance is a first step toward analyzing students' metacognitive and problem-solving processes. The data we analyze in this paper are data logged by UrbanSim collected in a study conducted with Reserve Officers' Training Corps (ROTC) officers-in-training at a major U.S. University. We analyze students' operations in the context of the state of the simulation. We present our analysis methods, and discuss how the results will help us define learner models that capture students cognitive and metacognitive processes.

COUNTERINSURGENCY

Understanding of COIN doctrine and strategies supported in UrbanSim are critical to successful problem-solving abilities and performance in UrbanSim. Counterinsurgency is the comprehensive civilian and military efforts designed to simultaneously defeat and contain insurgencies and address their root causes. Legitimacy – fostering effective governance by a legitimate government – is its main objective. COIN operations, therefore, aims to defeat insurgents while also working with local political and religious leaders to increase population support, separate (to protect) the population from insurgents, and ultimately install host nation (HN) governance that promotes self-sufficiency and economic growth.

As HN security forces often have insufficient capabilities to defeat the insurgents, coalition forces may initially shoulder the burden of being the primary counterinsurgents. The overall approach is governed by a stated Army doctrine called Clear-Hold-Build (CHB). Operations are conducted to engage and flush out insurgents in the *Clear* phase, clamp down and prevent insurgent activity in the *Hold* phase, and address some of the root causes of the insurgency and promote self-governance and economic viability in the *Build* phase.

CHB offers a broad guideline of how to conduct operations, and the following two variations are examples of specific guidelines on how these strategies may be executed. The *Inkblot* strategy is designed to enable the effective execution of CHB in large areas with limited assets. The strategy consists of establishing a home base in a friendly region, and then Clear and Hold regions that are adjacent to it. In the *Search & Destroy* strategy insurgents are actively sought out and engaged. This "hard" approach contrasts with a "soft" approaches that COIN suggests, designed to turn the population against an insurgency by satisfying the populations needs.

THE URBANSIM LEARNING ENVIRONMENT

UrbanSim (McAlinden et al., 2009), shown in Figure 1, is a turn-based simulation environment in which users assume command of a COIN operation in a region of a fictional Middle-Eastern country. Users have access to information that includes *Intelligence* reports (situation reports [SITREPs], significant activities [SIGACTs]), *Information* on the operational environment of each region (political, military, economic, social, information, infrastructure [PMESII]); *Progress* in increasing population support and the primary lines of effort (LOEs): improving civil security, governance, economic stability, HN security forces readiness, developing essential services, and cooperating with the local population; and *Causal Effects* of operations and events on population support, LOEs, and PMESII.

The LOEs are intended to support planning operations that link multiple tasks to focus effort toward establishing operational and strategic conditions. Users have a limited amount of resources at their command to perform COIN operations, which have to be directed toward making progress along the specified LOEs. Operations are conducted as fragmentary orders (FRAGOS) to available units (e.g., E Company B – E CO b) in the *Synch Matrix* (Figure 1, lower left). Once committed, the simulation executes the orders and models their effects on the regions of operation in the scenario. During this phase, additional events caused by other agents (e.g., the insurgents, the local population) can occur (e.g., the detonation of an improvised explosive device [IED]). The combination of all activities may result in net changes to key values.

Figure 1. UrbanSim: city map; Synch Matrix (lower left), LOE values (lower right); and SITREPs, SIGACTs (left border), Intel Officers S2, S3 (right border)

Students in UrbanSim conduct operations as a surrogate for the Battalion Commander (BN). The BN analyzes the area of operation (AO) with respect to the stated mission (defined by the Brigade [BDE] Commander), decides on the allocation of resources (by proxy through the player), and assesses progress toward achieving the mission goals. The analysis of an AO is expressed and displayed as a set of interrelated PMESII variables. The role of PMESII values is to organize and aggregate the information received during COIN operations to understand the consequences of previous operations, and assist in planning of subsequent operations. Interpreting PMESII values is a key competence of commanders, and COIN operations are decided and justified in relation to these values, and other intelligence information that may be become available.

UNDERSTANDING STUDENT ACTIONS

A Framework to Infer Metacognitive Skills

To represent student proficiency in domain-specific strategies and their more general cognitive and metacognitive counterparts, we have developed a task model hierarchy that contains a set of cognitive processes that are directly linked to the interpretation of relevant tasks that can be performed in the domain of operations, shown in Figure 2. The cognitive processes are themselves linked to strategic competencies (when should this action be executed and what are the expected consequences) that experts see as basic requirements in COIN operations. In UrbanSim, they include domain/task-specific actions, such as conducting CHB operations, a user action that links up to the more domain-general task of Solution Construction (SC). Students' *View* actions involve clicking on an interface item to display a page with information on individuals or groups, and clicking on items to view pop ups that provide information on PMESII values and results of intelligence gathering. These actions are linked to Information

Acquisition and Interpretation Actions (IAIs). Students' *Analysis* actions involve opening pages with causal graphs presenting effect relations (increase, decrease) between operations or events and population support, LOEs, and PMESII, an action linked to solution assessment (SA).

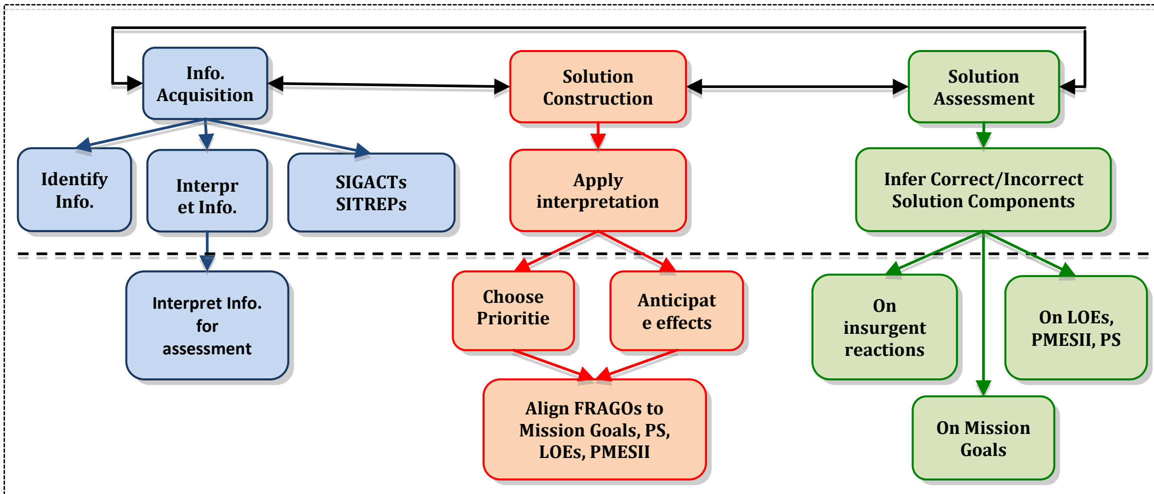


Figure 2: the UrbanSim task model (abbreviation: 'PS' is Population Support)

To advance our work on inferring metacognitive processes, we make the assumption that in this complex game-playing environment, good performance must require metacognitive awareness, therefore, metacognitive awareness can be inferred from students' performance. For effective problem solving, students need to employ metacognitive processes to maintain appropriate awareness. Metacognitive awareness directly influences the selection of operations: operations that advance the problem solutions are those that are conducted after the situation up to that point has been analyzed in terms of current state, and current and past trends. This requires assessing the effect of prior solution moves in terms of PMESII and LOE values and how they have changed over time, studying causal maps when available, and by incorporating the prediction of future game states.

Inferring Cognitive Skills and Domain-Specific Strategies

To track performance and make inferences on students' strategies, we distinguish between (1) performance values (e.g., LOE scores) and game state variables (e.g., *number of turns completed*), which we leverage to infer (2) more general structures directing behavior. These structures represent domain-specific strategies, such as implementations of aspects of the CHB doctrine. The concepts representing domain-specific strategies were compiled by interviewing ROTC officers with expertise in COIN, and analyzing video and audio records of student's working with UrbanSim. We thus obtained concepts to represent learners' common strategies/approaches as well as normative strategies that are seen by experts as basic requirements in COIN operations. The concepts at this level also represent strategies that are relevant in but not exclusive for COIN operations, such as situation awareness that involves seeking and interpreting information in the environment, dealing with trade-offs, and balancing negative and positive effects in a set of operations.

The analysis involves instantiating parameters (e.g., PMESII values) representing performance and make inferences on students' use of processes and strategies. Inferences on strategies are made by aggregating information made available in the UrbanSim logs from a small set of lower-level parameters and concepts. More formally, we leverage relations of dependency between processes and strategies, and performance values. However, this is not an easy task in the UrbanSim environment. This is primarily because UrbanSim does not keep a record of all of the information the user views. For example, UrbanSim records when students opens a page with a causal graph; however, the activation of the PMESII overlay is not logged. Therefore, the inferences we make are necessarily incomplete and uncertain.

An example of an instantiation of a strategy is the detection of CHB: by analyzing the PMESII values of all regions over a few turns, we obtain the number of regions in the Clear, Hold or Build phase, thus measuring students' ability to conduct operations aligned with CHB. In turn, when CHB is detected as a strategy, inferences on students' analysis of PMESII values can be made.

Tracking Performance and Detecting Strategies

At each turn, we leverage log data to detect students' performance and strategies by computing the values of the metrics presented now.

<u>CHB Strategy</u>: Once students have obtained and analyzed information, they are expected to conduct operations in line with that CHB strategy. PMESII analyses, and especially the M value (representing the degree of military control over a region), play a particularly important role in executing the strategy. We detect whether students' follow the CHB strategy by counting the number of regions in the Clear, Hold or Build phase at each turn. If students execute CHB consistently and appropriate to a region's PMESII value, the number of Clear regions will decline, and the number of Hold and Build regions will increase.

<u>Inkblot strategy</u>: *InkblotMatch* is the sum of values representing the distance of a region where an operation is conducted from a 'home base' (chosen by the students to be a base from which to fan out into adjacent regions). It is computed on values assigned to regions representing the distance to the home base. The values range from 0 (the 'home base') to 0.5 (the value of the region that is farthest from and that student normally select as the last region to conduct operations in).

<u>Search & Destroy Strategy</u>: *S&DMatch* is a measure of a 'hard' approach to *Clear*. The strategy is conducted by searching for and flushing out insurgents, and attacking them. *S&DMatch* is calculated by summing the number of 'aggressive' operations at each turn. These operations are: Cordon & Search, Patrol, Attack, Dispatch, Arrest, Seize, and Checkpoint.

<u>Lines of Effort</u>: the trend of the LOEs at each turn is tracked to obtain a measure of students' adherence to the Brigade Commander's intent.

<u>Population Support</u>: Population Support is logged as *for*, *against*, and *neutral* percentages, adding up to 100%. It is the key measure to assess student performance. UrbanSim scores performance at the end of the game with the formula: (*for* * 2) + *neutral* – *against*.

<u>Ineffectiveness</u>: the measure represents students' ability to select maximally effective operations, given the PMESII values of a region. The measure is the sum of ineffectiveness values for all operations in a turn. Ineffectiveness of an individual operation is calculated by summing its effect on 4 PMESII values (Military, Information, Social and Economic) and identifying the maximally effective operation. Ineffectiveness is the difference between the sum of effects of the conducted operation, and the sum of effects of the maximally effective operation. The calculation of the effect is weighted by the magnitude of PMESII values.

Events: The measure EventMatch is the sum of the number of responses to events at a turn. Events are reported in SIGACTs or SITREPs, or can be found by comparing the map of a current turn with the map of the previous turn (e.g. discovery of a hostile group). *EventMatch* is the sum of responses to events by turn.

Mission Goals: The Mission Description of the scenario explicitly requires the achievement of three *specific* mission goals: 1) increase the support of the town's Mayor; 2) prevent the influx of insurgents from the Mountains in the North, hence secure the Northern area and 3) repair the airport to facilitate the movement of personnel and goods. The measure *MissionGoalsMatch* is the sum of the number of operations at each turn conducted to further the specific mission goals.

STUDY

This is a study involving a novice population. UrbanSim has the player assume the role of a Battalion Commander. However, there is a significant experience and training gap between the player, and the role they are expected to play. That is, a Battalion Commander is an Officer with 18-20 years of experience and high-level training, whereas an ROTC Cadet has only several months of low-level training and no operational experience.

Aim and Method

In the past 2 years we have conducted four studies with ROTC students. We paired students to obtain verbal data from which to infer students' strategies and metacognitive behavior. Groups worked at a single computer with one student controlling the mouse. Talk and behavior (e.g., attention to a part of the map) is recorded as audio-video data from web-cams synced to a screen capture video. Data from 12 groups were obtained.

We conducted a qualitative analysis from which we developed summary accounts of students' strategies, and their attention to and processing of information. We extracted students' motivations to conduct operations and the information they attended from verbal data. Summary accounts are leveraged to validate computational techniques we developed for the automated detection of strategies, approaches, and analysis of information.

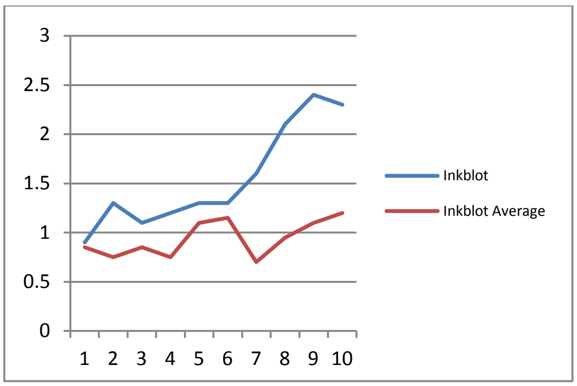
In this paper, we present summary accounts and quantitative results of two groups chosen because they differ markedly in strategies and information analysis, but also demonstrate some similarities. Thus, we illustrate how quantitative analyses discriminate between groups that follow CHB vs. those who don't; differ in their attention to values (focus on PMESII vs. on LOE); differ in situational awareness (responsive vs. not responsive to events); and how the analyses detect similarities in failing to integrate population support values in analysis and prediction.

Data Analysis and Interpretation

Group 1

Group 1 consists of two senior ROTC students (one male, one female). The group analyzes the map thoroughly and chooses to conduct operations in line with the Inkblot strategy; often discusses the map and their strategies before selecting operations; and focuses on region-specific values (PMESII, coalition support), the map, and events and pays little attention to LOEs and population support. The group occasionally analyzes the effect of operations on a region's PMESII values by consulting PMESII trends.

The group also conducts operations in line with CHB, and specifically with the Inkblot strategy. In the first turn, the group adopts a "soft" approach, but then decides to rely on a Search & Destroy approach in response to several violent events. Throughout all turns the group follows the Inkblot strategy when assets are available. We detect a good execution of *CHB* and an above average mean value of *Inkblot* (Figure 3, left). The group responds to all events. Students only very rarely motivate the choice of operations to increase population support; and very rarely select operations by considering LOE scores. We detect a low performance in *Population Support*, an average performance in increasing *LOE*s, and an average performance in choosing the most effective operations (Figure 3, right).



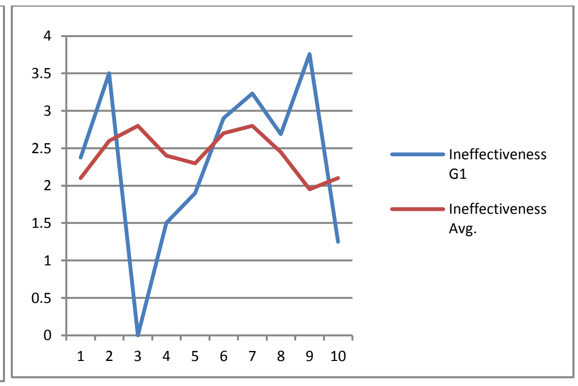


Figure 3 (left). Inkblot Scores of Group 1; Figure 4 (right). Ineffectiveness Value of Group 1, and Average

Inferences on Cognitive and Metacognitive Skills

Group 1 obtains high scores for *CHB* and *Inkblot*, but an average *Ineffectiveness* score, suggesting that that students are skilled in executing directives, but don't conduct systematic analyses on operation effects. Average *LOE* scores, a low *Population Support* score, and a high *Events* score suggest that the group focuses primarily on responding to events, and takes into account PMESII scores when conducting operations. These inferences are supported by the analysis of students' talk: students justify operations to execute Inkblot and react to events, leaving few assets available for operations that could increase Population Support.

Table 1 exemplifies the result of inference processes from primary data (e.g., the list of operations selected at each turn) or data computed on primary data (e.g., CHB score) to cognitive skills and metacognitive activities student may or may not have carried out.

Table 1. Inferences on primary or aggregated data of group 1. Bulleted items are alternative inferences.

Values and value patterns	Inferences
High *CHB* and *Inkblot*	Skilled in CHB and Inkblot Able to interpret PMESII
Average *Ineffectiveness*	- Moderate analysis of PMESII - Little analysis of operation effects
High *Events*	Attention to map, SIGACTs and SITREPs
Low *Population Support*, low *high-priority LOEs*, average Ineffectiveness,	- Persistence on approach or strategy - Little analysis of operation effects

Group 2

Group 2 consists of two male senior ROTC students. Also this group analyzes the map thoroughly and decides to implement the Inkblot strategy. However, the group doesn't analyze PMESII values before conducting operations. Rather, in many regions, the students conduct one security operation followed by a Recruitment operation to "hand over security of HN Forces". The students consider LOE scores, but pay little attention to population support. They respond to events infrequently. We detect a mean *Ineffectiveness* value that is significantly above average (t = 2.69, p = 0.015; see Figure 5, right), and below average performances in *Inkblot* (Figure 5, left) and in *Population Support*.

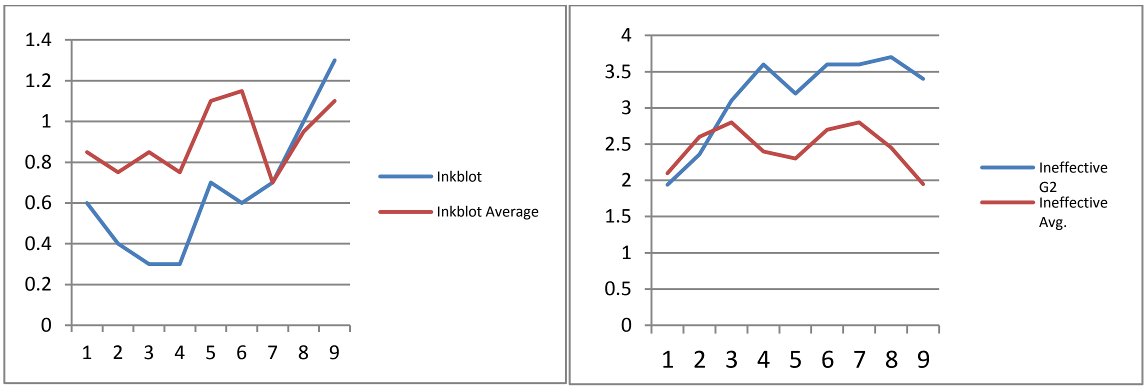


Figure 4 (left). Inkblot score of group 2 over 9 turns; (right) ineffectiveness value of group 2 over 9 turns, and average

Inferences on Cognitive and Metacognitive Skills

Group 2 obtains low scores in all metrics, except for *Civil Security* and *HN Security Forces*. We could not detect a strategy. The group also scores high on *Ineffectiveness*. *Events* score is average. Based only on quantitative data, the following explanations are possible: (1) students don't develop a strategy, (2) they have some misunderstandings on which operations are Clear, Hold, or Build operations, or (3) they don't analyze PMESII values. Figure 6 backs the conclusion that students' analysis of PMESII values is below average. The analysis of students' information acquisition behavior shows that students don't consult causal graphs. We also detect that students repeatedly use the same operation (Recruitment) and advance the conclusion that students follow a strategy without adapting it to local values or analyze its effect.

Table 2. Inferences on primary or aggregated data of group 2. Bulleted items are alternative inferences.

Values	Inferences
Low *CHB* and *Inkblot*	- Decide not to follow CHB or Inkblot - Don't view or unable to interpret PMESII - Misunderstandings on which operations are Clear, Hold or Build operations
Low *Population Support*, low *high-priority LOEs*, high Ineffectiveness	- Persistence on approach or strategy - Misunderstandings of operation effects - Little analysis of operation effects

DISCUSSION AND FUTURE WORK

Our analyses show that students are often skilled in executing directives, responding to events and counter-acting negative trends of one or two key values. Our analyses suggest also that students fail to integrate information and analyses to generate a picture of the operational environment based on the assessment of prior solution effects and by incorporating the prediction of future game states. Accounts of students' justifications of operations reported in the case studies show that before selecting operations students typically analyze a single region or value, or they may react to events. Expertise in COIN means, however, to be able to conduct operations for local *and* broader and long-term effects (including 2^{nd} and 3^{rd} order effects).

Our analyses have also detected that students frequently hold incomplete knowledge about the effect of operations. Small positive effects visible on the value indicators on the map interface appear to be sufficient for the students to repeatedly conduct the same operations. The measure of ineffectiveness has emerged as central to detect students' incomplete knowledge and their inclination to analyze operation effects – a critical metacognitive activity. Discriminating between incomplete knowledge and insufficient analysis will allow us to better model the learner in terms of cognitive and metacognitive skills.

However, our interpretation of students' strategies is an indirect inference that is incomplete, in general, and necessarily uncertain. This becomes a primary challenge in learner modeling and generating adaptive scaffolds. In future work, we will extend and generalize our hierarchical task and corresponding learner model to better capture the nuances of students proficiencies and their learning behaviors. In general, this hierarchy will include cognitive processes related to the task domain, expressed in domain-specific and domain-independent form at the lower levels of the hierarchy, cognitive and metacognitive strategies, again expressed in domain-specific and domain-general forms (when applicable) at the middle levels of the hierarchy, and metacognitive processes at the highest level of the hierarchy. The reason for including both domain-specific and domain-general nodes is that the domain-specific nodes imply the definition of detectors that we can design in the training environment to detect and analyze students' performance and learning and problem-solving behaviors, whereas the domain-general constructs apply across multiple training domains. Examples of domain-specific detectors, and their applications to analyzing student behaviors, have been illustrated in this paper. Currently, we are in the process of developing these detectors in the GIFT system. In future work, we will extend this approach to derive domain-general constructs, which will be integrated into the learner model in GIFT.

REFERENCES

Basu, S. & Biswas, G. (2016). Providing adaptive scaffolds and measuring their effectiveness in open ended learning environments, in Proc. of the International Conference of the Learning Sciences (ICLS2016), Singapore, June 2016.

Butler, D. L. & Winne, P.H. (1995) Feedback and self-regulated learning: a theoretical synthesis, Review of Educational Research, 65, pp. 245-281

Goldberg, B. & Cannon-Bowers, J. (2013). Experimentation with the generalized intelligent framework for tutoring (GIFT): a testbed use case. In AIED 2013 Workshops Proceedings Volume (Vol. 7, pp. 27-36).

Goldberg, B. & Spain, S. (2014). Creating the Intelligent Novice: Supporting Self-Regulated Learning and Metacognition in Educational Technology. In R. Sottilare, A. Graesser, X. Hu & B. Goldberg (Eds.) Design Recommendations for Intelligent Tutoring Systems, Vol. 2: Adaptive Instructional Strategies, U.S. Army Research Laboratory.

Kinnebrew, J., Segedy, J.R. & Biswas, G. (2016). Integrating Model-Driven and Data-Driven Techniques for Analyzing Learning Behavior in Open-Ended Learning Environments. IEEE Transactions on Learning Technologies.

McAlinden, R., Durlach, P., Lane, H., Gordon, A. & Hart, J. (2008). UrbanSim: A game-based instructional package for conducting counterinsurgency operations. In: Proceedings of the 26th Army Science Conference, Orlando, FL.

Roscoe, R. D., Segedy, J. R., Sulcer, B., Jeong, H. & Biswas, G. (2013). Shallow strategy development in a teachable agent environment designed to support self-regulated learning, Computers & Education, vol. 62, pp. 286 – 297.

Segedy, J. R., Kinnebrew, J. S. & Biswas, G. (2015). Using coherence analysis to characterize self-regulated learning behaviors in open-ended learning environments, The Journal of Learning Analytics, 2(1), 13-48.

Sottilare, R. A., Goldberg, B. S., Brawner, K. W. & Holden, H. K. (2012). A modular framework to support the authoring and assessment of adaptive computer-based tutoring systems (CBTS). In Proceedings of the Interservice/Industry Training, Simulation, and Education Conference.

Sottilare, R. A. & Holden, H. K. (2013). Motivations for a generalized intelligent framework for tutoring (GIFT) for authoring, instruction and analysis. In AIED 2013 Workshops Proceedings, vol. 7, pp. 1-9.

Zimmerman, B. & Schunk, D. (Eds.). (2011). Handbook of Self-Regulation of Learning and Performance. New York, NY: Routledge.

ABOUT THE AUTHORS

Dr. Michael Tscholl *is a Postdoctoral Researcher at the Institute of Software Integrated Systems at Vanderbilt University. His research focuses on developing measures of open-ended and independent problem solving behaviors to develop adaptive scaffolds such as contextualized conversational feedback and guided practice in open-ended learning environments.*

Dr. Gautam Biswas *is a Professor of Computer Science, Computer Engineering, and Engineering Management in the EECS Department and a Senior Research Scientist at the Institute for Software Integrated Systems (ISIS) at Vanderbilt University. He conducts research in Intelligent Systems with primary interests in simulation, and the design and development of Computer-Based Learning Environments. His research has been supported by funding from ARL, NASA, NSF, DARPA, and the US Department of Education.*

Dr. Benjamin Goldberg *is an adaptive training scientist at the Army Research Laboratory's SFC Paul Ray Smith Simulation & Training Technology Center. He leads research focused on instructional management within ARL's Learning in Intelligent Tutoring Environments (LITE) Lab and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT).*

Dr. Robert A. Sottilare *leads adaptive training research within the US Army Research Laboratory where the focus of his research is automated authoring, automated instructional management, extension of tutors to psychomotor tasks, and evaluation tools and methods for intelligent tutoring systems. His work is widely published and includes articles in the Cognitive Technology Journal, the Educational Technology Journal, and the Journal for Defense Modeling & Simulation. Dr. Sottilare is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT), an open-source tutoring architecture, and he is the chief editor for the Design Recommendations for Intelligent Tutoring Systems book series. He is a visiting scientist and lecturer at the United States Military Academy and a graduate faculty scholar at the University of Central Florida. Dr. Sottilare received his doctorate in Modeling & Simulation from the University of Central Florida with a focus in intelligent systems. In 2012, he was honored as the inaugural recipient of RDECOM's Modeling & Simulation Lifetime Achievement Award, and in 2015 he was honored with the National Training and Simulation Association (NTSA) Governor's Award for Modeling & Simulation Lifetime Achievement.*

Motivational Feedback Messages as GIFT Interventions to Frustration

Jeanine A. DeFalco[1], Vasiliki Georgoulas-Sherry[1], Luc Paquette[2], Ryan S. Baker[1],
Jonathan Rowe[3], Bradford Mott[3,], James Lester[3]
[1]Teachers College, Columbia University, New York, NY
[2] University of Illinois at Urbana-Champaign, Champaign, IL
[3] North Carolina State University, Raleigh, NC

INTRODUCTION

Determining how to effectively respond to learner affect is important not only in face-to-face learning environments (Pekrun, Goetz, Titz & Perry, 2002), but also within the field of intelligent tutoring systems (ITSs) (Goldberg et al., 2012). This requires not only tools to accurately identify affect, but also developing a suite of accompanying interventions that can respond to learner affect (D'Mello, Lehman & Graesser, 2011).

In an effort to help learners regulate their affective states, some computer tutoring systems researchers have used interventional feedback messages to motivate the learner through a frustrated state (Robison, McQuiggan & Lester, 2009). However, these researchers have noted that where frustration has been detected and feedback delivered, learners do not always respond positively to these interventions, but rather may react negatively to feedback provided by the system (Robison, McQuiggan & Lester, 2009). This has given rise to the need to take a closer examination of the design of motivational feedback messages delivered to learners in a frustrated state to determine the most effective approach for addressing learner frustration via interventional feedback messages.

Within this context, then, the gap addressed by this current work is an effort in determining what kind of motivational feedback messages delivered within an ITS effectively addresses the affective state of frustration within a simulation-based training game and promotes learning gains. Three theories of motivation were targeted to design feedback messages: (1) theory of control-value (Pekrun, Elliot & Maier, 2006); (2) theory of social identity (Tajfel & Turner, 1979); and (3) theory of self-efficacy (Bandura, 1977). These theories are distinct from each other in the way they target either a person's sense of what they value (control-value theory), who they are (social identity theory), and what a person believes they can achieve (self-efficacy theory).

THEORY AND PREVIOUS RESEARCH

Effectively supporting cognitive performance is increasingly understood to depend on a broader understanding of the relationship between affect, motivation, and cognition interactions. Prior research in the area of motivation and cognition has demonstrated that the presence of positive motivation enhances working memory, memory encoding, decision making, selective attention, response inhibition, and task switching (Locke & Braver, 2010). Further, motivational processes associated with affective states have been shown to have also had a significant impact on memory, perception, attention, and categorization (Harmon-Jones, Gable & Price, 2013).

Accordingly, this paper discuss the results of a study ran in September 2015 that examined the effect of motivational feedback messages delivered to participants playing the serious video game vMedic while participants engaged in a modified Tactical Combat Casualty Care (TC3) course delivered by the Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden, 2012). Using previously published sensor-free detectors of student frustration (Paquette et al., 2015), GIFT automatically detected whether students were highly frustrated, and if so, immediately delivered feedback messages to motivate the learner to persist in their learning task.

Sensor-free detectors

Sensor-free detectors are computational models that automatically detect learners' affective states from their interaction with online learning. For this September 2015 study, we used the sensor-free affect detector for frustration developed by Paquette and colleagues (2015), built using log data and Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) field observations from a previous study conducted at West Point (United States Military Academy [USMA]) (September 2013), the same setting as the current study. Machine-learning algorithms implemented in the RapidMiner tool were used to identify the relationship between features of participants' interaction and observations of frustration, and a model was built that was able to predict when a student was frustrated. The resulting model takes summary features of the learner's behavior as an input and outputs its confidence that the learner is frustrated (the confidence is a probability between 0 and 1). For the purpose of this paper's interventions, we treat a confidence of > 0.5 as evidence that the participant is highly frustrated; values below that are treated as not frustrated.

PROJECT DESIGN

The experiment used a modified version of the U.S. Army's TC3 course on tactical field care and care under fire, focusing specifically on hemorrhage control and bleeding. The main study used a pre- and post-test, control group design. Conducted on laptops, the tasks of this experiment included a demographics questionnaire, a pre-test, the modified TC3 PowerPoint, five scenarios of vMedic, the Short Grit Scale Survey (Duckworth & Quinn, 2009), a Presence survey (Witmer & Singer, 1994), and a post-test.

Participants completed five scenarios within vMedic: (1) a relatively easy to solve introductory scenario, (2) multiple injuries, (3) a no-win situation (referred to as Kobayashi-Maru), (4) multiple injuries again, and (5) a second no-win situation. These were sequenced in this manner to elicit the most amount of frustration that could be reasonably manipulated without risking complete disengagement from the game.

There were five conditions in this experiment: (1) control-value motivational feedback messages, (2) social identity motivational feedback messages, (3) self-efficacy motivational messages, (4) non-motivational feedback message condition – factoids related to hemorrhage control and tourniquets (control condition 1), and (5) no intervention (full control; control condition 2) (see Appendix).

In the four message conditions, GIFT used the sensor-free detectors to trigger frustration adaptations. Upon the detection of high frustration, a single audio motivational feedback message would be delivered to the participant by GIFT. The motivational and non-motivational feedback messages were delivered a total of once per scenario.

The data collected in this experiment included all answers to the questionnaires and surveys, in addition to the log files that contained all the data of the experiment and participant interaction – including system detected rates of frustration recorded for each participant. These log files were extracted from GIFT via the Event Report Tool, a function within GIFT that exports all data of participants logged into GIFT while taking the course/experiment.

Participants in the experiment included 141 volunteers from the Corps of Cadets at the USMA in West Point, NY. The ages of the participants ranged from 17 to 25. Pre- and post-test measures were collected for 141 participants. Out of those, 17 participant log files had a gap in the output where the participant either did not have a pre-test or a post-test due to a technical failure, resulting in loss of data. Subsequently, these 17 participants were dropped from the data analysis. In total, the final data analysis was run on 124 participants (14 females and 110 males) who participated in this study: (1) 26 participants in the control-value motivational feedback messages (condition 1), (2) 26 participants in the social identity motivational feedback messages (condition 2), (3) 24 participants in the self-efficiency motivational messages (condition 3), (4) 25 participants in the non-motivational feedback message condition (control condition 1), and (5) 23 participants in the no intervention (full control; control condition 2).

RESULTS

Analysis of the logs of interventions in vMedic indicated that every participant in a feedback condition received a message in every vMedic scenario except for the first. This result was not unexpected as the sequence of the vMedic scenarios were designed to have the first scenario be relatively easy to solve, thereby not eliciting a high level of frustration.

The condition with the greatest frequency of system-detected frustration was the no message condition, (the full control condition 2), with a mean frequency of 6.70 times that the sensor-free affect detectors detected high frustration across all scenarios. The two conditions with the lowest frequencies detected for high frustration were the control-value condition (condition 1), with a mean of 6.19 detected high frustration events, and the self-efficacy condition (condition 3), with a mean of 6.33 detected high frustration events (Figure 1).

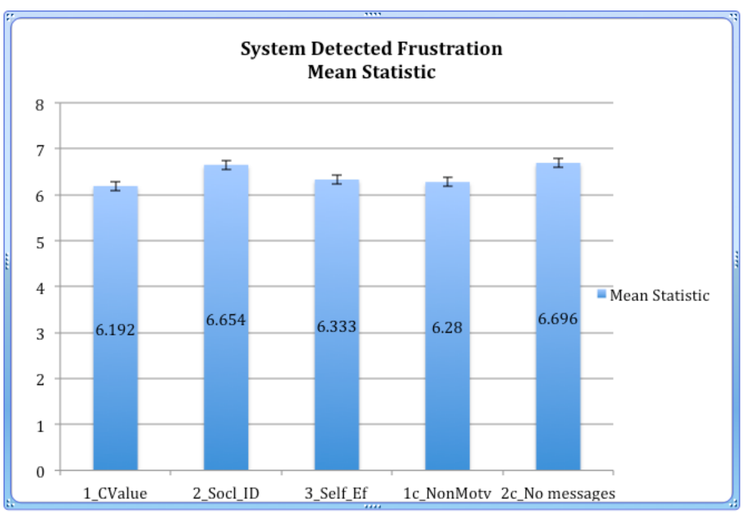


Figure 1. Mean frequency of system detected frustration by condition and standard error.

There were no statistically significant differences in the frequency of frustration between conditions, $F(4,119) = 0.581$, $p = 0.677$.

To test if there was a statistically significant difference between motivational feedback vs. non-motivational conditions, a two-way mixed design repeated measures analysis of variance (rANOVA) design was used to analyze the effect of two independent factors on the dependent variable (tests), where one of the factors was the between subjects (condition) and the other was a within-subjects factor (system detected frustration). Comparing the motivational conditions (conditions 1, 2, and 3) to the control conditions (conditions 4 and 5), when testing for a three-way interaction between tests-frustration-condition, there was a statistically significant difference in pre-post test scores (rANOVA): $F(1, 120) = 5.578$, $p = 0.020$. $\eta p2 = 0.044$, power = 0.649.

Conducting a post-hoc, simple main analysis to investigate the three-way interaction of condition and frustration on pre-post test scores, independent pairwise rANOVA's were run comparing each motivational condition separately to each control condition, using the Benjamini-Hochberg[1] alpha adjustment procedure to control for false discovery rate in multiple comparisons (see a summary of findings in Table 1).

Table 1

Summary of pairwise analyses (rANOVA's) between intervention conditions vs. control groups

Intervention	*Control group*	*df*	*F*	*Sig*	*Adjusted α*	ηp2	*Power*
Control-value	Non motivational messages	1	1.079	.304	0.033	.022	.175
Social identity	Non motivational messages	1	.650	.424	0.041	.014	.124
**Self efficacy*	*Non motivational messages*	1	9.945	.003	0.008	.181	.870
Control-value	No messages	1	2.200	.145	0.025	.047	.306
Social identity	No messages	1	.345	.560	0.500	.008	.089
**Self-efficacy*	*No messages*	1	7.355	.010	0.016	.146	.755

* statistically significant after Benjamini-Hochberg correction

[1] The Benjamini-Hochberg procedure is an approach to controlling the false discovery rate in multiple comparisons (Benjamini & Hochberg, 1995; Thissen, Steinberg & Kuang, 2002) which is thought to balance between Type I and Type II error better than more traditional family-wise error rate tests such as Bonferroni.

A statistically significant difference was found between the self-efficacy condition ($N = 24$) and the non-motivational feedback control group ($N = 25$), (rANOVA): $F(1, 45) = 9.945$, $p = 0.003$, $\eta p2 = 0.181$, power = 0.870. Using the Benjamini-Hochberg adjusted alpha, these results are still significant: $p = 0.003 <$ B-H $\alpha = 0.008$.

Also, there was a statistically significant difference between the self-efficacy condition ($N = 24$) and the no messages control group ($N = 23$), (rANOVA): $F(1, 43) = 7.355$, $p = 0.010$, $\eta p2 = 0.146$, power = 0.755. Again, using the Benjamini-Hochberg adjusted alpha, these results are still significant: $p = 0.010 <$ B-H $\alpha = 0.016$. No other comparisons were significant when using the Benjamini-Hochberg procedure.

Tests were also conducted to examine the relationship between presence and grit on student learning. Taking measures from an administered Presence questionnaire (Witmer & Singer, 1994) and the Short Grit Scale (Duckworth & Quinn, 2009), separate two-way mixed design rANOVA analyses were conducted examining the effect of presence and grit on student learning. Presence did not have a statistically significantly effect size associated with pre-post test scores, (rANOVA): $F(1,114) = 1.639$, $p = 0.203$, $\eta p2 = 0.014$, power = 0.246, and no statistically significant interaction was found between presence and condition on pre-post test scores, (rANOVA): $F(4,114) = 0.162$, $p = 0.957$, $\eta p2 = 0.006$, power = 0.083.

There was a statistically significant interaction effect of grit by condition and pre-post test scores (rANOVA): $F(4,114) = 2.903$, $p = 0.025$, $\eta p2 = 0.092$, power = 0.768. Given this significant interaction, an analysis on the simple effects of grit by condition were conducted, running simple main effect analyses separately at each level of condition. The results of this simple means analysis showed that grit had a statistically significant effect with pre-post test outcomes only within the control-value condition (condition 1) $F(1, 24) = 7.304$, $p = 0.012$, $\eta p2 = 0.233$, power = 0.737. However, in examining the Benjamini-Hochberg alpha adjustments, the control-value condition marginally misses significance $p = .012 >$B-H $\alpha = 0.01$.

Splitting the data further into high and low grit groups, using the mean grit value of 3.80, a statistically significant difference was found between the pre and post tests for low grit participants in the control-value theory condition (condition 1), (rANOVA): $F(1, 25) = 35.000$, $p = 0.001$, $\eta p2 = 0.883$, power = 0.999. After making Benjamini-Hochberg alpha adjustments, the low-grit condition remained significant: $p = 0.001 <$ B-H $\alpha = 0.005$, and low grit participants in the control-value condition had positive pre-post test outcomes, which was different than the high grit participants who did not have statistically significant learning. This suggests that the control-value messages had a positive impact on participants with low grit scores, perhaps encouraging them to see the value in the experiment or the learning activity more broadly. In contrast, for the high grit participants, it seems as if these participants might have seen the control-value messages as unnecessary, annoying, or even frustrating – perhaps causing some disengagement with the experiment/learning activity.

CONCLUSIONS

In conclusion, the results of this experiment support previous theories and empirical research that have recognized the need to identify and address affective states that lead to disengagement in learning (D'Mello, Lehman & Graesser, 2011), and gives further evidence that providing interventions in the form of feedback messages can positively affect the learning of domain content in ITSs (Roll, Aleven, McLaren & Koedinger; 2011). We find that self-efficacy based interventions are associated with better learning, when controlling for frustration, though they do not specifically reduce frustration themselves.

This study also provides further evidence of the complex interaction of affect, motivation, and cognition. Specifically, this study illuminates the mediating effect that frustration can bring to bear on learning, and provides evidence that through the development of trait-based and situationally grounded motivational messages – and connected to an automated detector that infers student frustration – positive learning outcomes can be enhanced in an intelligent tutoring system platform such as GIFT.

Future studies should test to see whether older, active members of the U.S. Army would respond differently to the existing body of motivational messages employed in this study. Also, to establish generalizability of these findings, future research should replicate this study on a more heterogeneous population. Lastly, further studies are needed to examine other motivational theory-based designs, as well as how other psychological traits interact with frustration and motivation in order to support cognitive performance more broadly.

ACKNOWLEDGMENTS

We thank our research colleagues, COL James Ness and Dr. Michael Matthews in the Behavioral Science & Leadership Department at the United States Military Academy, and Dr. Robert Sottilare and Dr. Keith Brawner, US Army Research Laboratory for their assistance in conducting this study. The research described herein has been supported in part by a cooperative agreement #W911NF-13-2-0008 between the U.S. Army Research Laboratory, Teachers College Columbia University, and North Carolina State University. Any opinions, findings, and conclusions expressed in this paper are those of the authors and do not necessarily reflect the views of the U.S. Army Research Laboratory.

REFERENCES

Baker, R.S.J.d., D'Mello, S.K., Rodrigo, M.M.T., Graesser, A.C. (2010). Better to be frustrated than bored: The incidence, persistence, and impact of learners' cognitive-affective states during interactions with three different computer-based learning environments. In *International Journal of Human-Computer Studies*, 68, 223-241.

Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological review*, 84(2), 191.

Benjamini, Y. & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society. Series B (Methodological), 289-300.

Braver, T. (2015). (Ed.) *Motivation and Cognitive Control*. London, GB: Routledge.

Buck, R. (1985). Prime theory: An integrated view of motivation and emotion. Psychological review, 92(3), 389.

Chapman, P. L., Cabrera, L. D., Varela-Mayer, C., Baker, M. M., Elnitsky, C., Figley, C., ... & Mayer, L. P. (2012). Training, deployment preparation, and combat experiences of deployed health care personnel: key findings from deployed U.S. Army combat medics assigned to line units. Military medicine, 177(3), 270-277.

Craig, S., D'Mello, S., Witherspoon, A. & Graesser, A. (2008). Emote aloud during learning with AutoTutor: Applying the facial action coding system to cognitive-affective states during learning. *Cognition & Emotion*, 22(5), 777-788.

D'Mello, S. & Graesser, A. (2011). The half-life of cognitive-affective states during complex learning. *Cognition & Emotion*, 25(7), 1299-1308.

D'Mello, S. K., Lehman, B. & Graesser, A. (2011). A motivationally supportive affect-sensitive autotutor. In *New perspectives on affect and learning technologie*s (pp. 113-126). Springer New York.

D'Mello, S., Picard, R. W. & Graesser, A. (2007). Toward an affect-sensitive AutoTutor. *IEEE Intelligent Systems*, (4), 53-61.

D'Mello, S. K., Strain, A. C., Olney, A. & Graesser, A. (2013). Affect, meta-affect, and affect regulation during complex learning. In *International handbook of metacognition and learning technologies* (pp. 669-681). Springer New York.

De Lorenzo, R. A. (2001). Medic for the millennium: the U.S. Army 91W health care specialist. Military medicine, 166(8), 685.

DePillis, L. (2013). The return of the tourniquet: What we learned from war led to lives saved in Boston. New Republic, April 17, 2013. Retrieved on February 8, 2015 at http://www.newrepublic.com/article/112939/boston-marathon-bombing-return- tourniquet.

Duckworth, A. L. & Quinn, P. D. (2009). Development and validation of the Short Grit Scale (GRIT–S). *Journal of personality assessment*, 91(2), 166-174.

Goldberg, B. S. (2013). Explicit Feedback Within Game-based Training: Examining the Influence of Source Modality Effects on Interaction (Doctoral dissertation, University of Central Florida Orlando, Florida).

Harmon-Jones, E., Gable, P. A. & Price, T. F. (2013). Does negative affect always narrow and positive affect always broaden the mind? Considering the influence of motivational intensity on cognitive scope. Current Directions in Psychological Science, 22(4), 301-307.

Jacobs, L., McSwain, N., Rotondo, M., Fabbri, W., Eastman, A., Butler, F. & Sinclair, J. (2013). Improving survival from active shooter events: The Hartford Consensus. American College of Surgeons and Federal Bureau of Investigation: Joint Committee to Create a National Policy to Enhance Survivability from Mass Casualty Shooting Events. April 2, 2013. Retrieved on 8 February 2015 at http://www.scribd.com/doc/136325175/Improving-Survival-From-Active-Shooter-Events

Kragh Jr, COL (ret), J. F., Dubick, M. A., Aden, J. K., McKeague, A. L., Rasmussen, T. E., Baer, D. G. & Blackbourne, L. H. (2015). U.S. Military Use of Tourniquets from 2001 to 2010. Prehospital Emergency Care, 19(2), 184-190.

Kragh Jr, J. F., Walters, T. J., Wesstmoreland, T., Miller, R. M., Mabry, R. L., Kotwal, R. S., ... & Cain, J. S. (2013). Tragedy into drama: an American history of tourniquet use in the current war. Army Inst Of Surgical Research, Fort Sam Houston, TX.

Locke, H. S. & Braver, T. S. (2010). Motivational influences on cognitive control: A cognitive neuroscience perspective. In R. R. Hassin, K. N. Ochsner & Y. Trope (Eds.), Self-control in society, mind, and brain (pp. 114– 140). New York: Oxford University Press.

Maddox, W. T. & Markman, A. B. (2010). The motivation-cognition interface in learning and decision-making. *Current Directions Psychological Science*, 19 (2), 106– 110. Doi:10.1177/0963721410 364008.

Paquette, L., Rowe, J., Baker, R., Mott, B., Lester, J., DeFalco, J., Brawner, K., Sottilare, R., Georgoulas, V. (2015). Sensor-Free or Sensor-Full: A Comparison of Data Modalities in Multi-Channel Affect Detection. In *Proceedings of the 8th International Conference on Educational Data Mining*.

Pardos, Z. A., Baker, R. S., San Pedro, M., Gowda, S. M. & Gowda, S. M. (2014). Affective states and state tests: investigating how affect and engagement during the school year predict end-of-year learning outcomes. test, 1(1), 107-128.

Pekrun, R., Goetz, T., Titz, W. & Perry, R. P. (2002). Academic emotions in students' self-regulated learning and achievement: A program of qualitative and quantitative research. Educational psychologist, 37(2), 91-105.

Pekrun, R., Elliot, A. & Maier, M. (2006). Achievement goals and discrete achievement emotions: A theoretical model and prospective test. Journal of Educational Psychology, 98(3), 583-597.

Picard, R. W., Papert, S., Bender, W., Blumberg, B., Breazeal, C., Cavallo, D., ... & Strohecker, C. (2004). Affective learning—a manifesto. *BT technology journal*, 22(4), 253-269.

Robison, J., McQuiggan, S. & Lester, J. (2009, September). Evaluating the consequences of affective feedback in intelligent tutoring systems. In Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on (pp. 1-6). IEEE.

Roll, I., Aleven, V., McLaren, B. M. & Koedinger, K. R. (2011). Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and Instruction*, 21(2), 267-280.

Shuker, S. (2011). The immediate lifesaving management of maxillofacial, life-threatening hemorrhages due to IED and/or shrapnel injuries: "When hazard is in hesitation, not in the action." Journal of Cranio-Maxillo-Facial Surgery, 40: 534-540.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Tajfel, H. & Turner, J. C. (1979). An integrative theory of intergroup conflict. The social psychology of intergroup relations, 33(47), 74.

Thissen D., Steinberg L. & Kuang D. (2002). Quick and easy implementation of the Benjamini-Hochberg procedure for controlling the false positive rate in multiple comparisons. J Educ Behavioral Stat., 27(1):77–83.

Wagster, J., Tan, J., Wu, Y., Biswas, G. & Schwartz, D. (2007). Do learning by teaching environments with metacognitive support help students develop better learning behaviors. In *The twenty-ninth Annual Meeting of the Cognitive Science Society,* Nashville, Tennessee (pp. 695-700).

Witmer, B. G. & Singer, M. J. (1994). Measuring immersion in virtual environments. *Tech. Rep*. 1014, U.S. Army Research Institute for the Behavioral and Social Sciences.

Woolf, B., Burleson, W., Arroyo, I., Dragon, T., Cooper, D. & Picard, R. (2009). Affect-aware tutors: recognizing and responding to student affect. *International Journal of Learning Technology*, 4(3-4), 129-164.

ABOUT THE AUTHORS

***Ms. Jeanine A. DeFalco** is a full-time clinical faculty member at Pace University; a PhD candidate at Teachers College (TC), Columbia University in the Department of Human Development, Cognitive Studies; and a doctoral research fellow in Dr. Baker's Lab. Her research focuses on motivation, engagement, and instructional design in face-to-face and tech-based learning platforms. She holds an MA in education (theatre and English) from New York University and a Masters in drama studies from The Johns Hopkins University.*

***Ms. Vasiliki Georgoulas-Sherry**, is a research psychologist for USMA and a doctoral research fellow in the Department of Human Development at TC, Columbia University. Her research focuses on cognitive processing and psychological resilience, soldier performance in combat, and other related contexts in the Army.*

***Dr. Luc Paquette** is an assistant professor of curriculum and instruction at the University of Illinois at Urbana-Champaign where he specializes in educational data mining and learning analytics. He earned a PhD in computer science from the University of Sherbrooke, where he worked on the development of Astus, a model-tracing tutor authoring framework. As part of the Astus team, his main research project involved developing algorithms allowing Astus to generate pedagogical content by examining the model of the tutored task. He was a post-doctoral research associate working with Dr. Ryan Baker, TC, Columbia University. His current research focus on integrating knowledge engineering and educational data mining approaches to create better and more general models of students who disengage from digital learning environments by "gaming the system".*

***Dr. Ryan S. Baker** is an associate professor of cognitive studies at TC, Columbia University, and Program Coordinator of TC's Masters of Learning Analytics. He earned his PhD in human-computer interaction from Carnegie Mellon University. He was previously assistant professor of psychology and the learning sciences at Worcester Polytechnic Institute, and served as the first technical director of the Pittsburgh Science of Learning Center DataShop. He is currently serving as the founding president of the International Educational Data Mining Society and as associate editor of the Journal of Educational Data Mining. His research combines educational data* *mining and quantitative field observation methods to better understand how students respond to educational software, and how these responses impact their learning.*

Dr. Jonathan Rowe *is a research scientist in the Center for Educational Informatics at North Carolina State University. He received PhD and MS degrees in computer science from North Carolina State University, and a BS degree in computer science from Lafayette College. His research is in the areas of artificial intelligence and human-computer interaction for advanced learning technologies, with an emphasis on game-based learning environments, intelligent tutoring systems, user modeling, educational data mining, and computational models of interactive narrative.*

Dr. Bradford Mott *is a senior research scientist in the Center for Educational Informatics at North Carolina State University. His research interests include artificial intelligence and human-computer interaction, with applications in educational technology. In particular, his research focuses on game-based learning environments, ITSs, and computational models of interactive narrative. He has many years of software development experience from industry, including extensive experience in the video game industry, having served as technical director at Emergent Game Technologies where he created cross-platform middleware solutions for Microsoft's Xbox and Sony's PlayStation video game consoles.*

Dr. James Lester *is a distinguished professor of computer science at North Carolina State University, where he is Director of the Center for Educational Informatics. His research centers on transforming education with technology-rich learning environments. With a focus on adaptive learning technologies, his research spans ITSs, game-based learning environments, affective computing, and tutorial dialogue. The adaptive learning environments he and his colleagues develop have been used by thousands of students in K-12 classrooms. The recipient of a National Science Foundation CAREER Award, he has been named a AAAI Fellow by the Association for the Advancement of Artificial Intelligence.*

Appendix: Feedback Messages

Condition 1: Control-Value Theory

1. "Studies have shown that between 17%–19% of deaths in Vietnam could have been prevented if tourniquets had been used."

2. "A 2008 study from a hospital in Baghdad found an 87% survival rate with use of tourniquets."

3. "There is no room for hesitation or consultation in facial injuries, and quick action (3–10 minutes) is critical to the survival and recovery of injured soldiers."

4. "The number one cause of preventable deaths in active shooter events is blood loss, and the best way to stop blood loss is to properly apply a tourniquet."

5. "The first U.S. casualty to die in the war from enemy fire was a Special Forces Soldier, SFC Nathan Chapman, who died during medical air-evacuation on 4 January2002 from isolated limb exsanguination without tourniquet use," (Kragh et al., 2013)

Condition 2: Social Identity Theory

1. "As General Maxwell Thurman said, "Make good things happen for our Army."

2. "Remember, solder, what General Patton said: An Army is a team. It lives, sleeps, eats, and fights as a team."

3. "Every single man in this Army plays a vital role", said General Patton. "Don't ever let up. Every man has a job to do and he must do it."

4. "General MacArthur once said: Duty, Honor, Country, are three hallowed words that dictate what you ought to be, what you can be, what you will be."

5. "General Patton said that the soldier is both a citizen and the Army, and the highest obligation and privilege of citizenship is the bearing arms for one's country."

Condition 3: Self-Efficacy Theory

1. "In this important combat situation, your best outcomes will be achieved if you persist."

2. "You can succeed in this because you've been trained to succeed under all conditions."

3. "Tell yourself that you will succeed because failure is not an option in this high stakes combat zone."

4. "Difficult doesn't mean impossible. It means work harder till your combat mission is achieved."

5. "In all combat situations, success comes from overcoming the things you thought you couldn't."

Control Condition 1 – Non-Motivational Feedback Messages

1. "Battlefield care emerged in Europe when Post-Revolutionary France established a system of prehospital care that included a corps of litter-bearers to remove wounded individuals from the battlefield," (Chapman et al., 2012).

2. "The modern combat medic has its roots in the American Civil War, when enlisted soldiers served as hospital stewards." (De Lorenzo, 2001).

3. "As of 10 September 2001, the unreliable, World War II–era U.S. Army tourniquet was the only widely fielded tourniquet in the U.S. military," (Kragh et al., 2013).

4. "In 2003, in the farmlands around Fort Bragg, Amanda Westmoreland became a tourniquet maker by melting and bending plastic tourniquet components in her living rooms, packaging and distributing thousands of assembled tourniquets early in the war against Iraq," (Kragh et al., 2013).

5. "The use of a tourniquet went from a means of last resort to a means of first aid and became the prehospital medical breakthrough of the wars in Afghanistan and Iraq," (Kragh et al., 2013).

Control Condition 2:

No messages

The Use of Social Media for Creating and Improving Learning Content

Rodney A. Long[1], Irene T. Boland, PhD[2], Doug Raum[2], Dan Silverglate[2], Ed Sims, PhD[2]
[1]Army Research Laboratory (ARL)
[2]Vcom3D, Inc.

INTRODUCTION

The US Army trains and educates over a half million individuals per year in a course-based, throughput-oriented system. Much of the Army's web-based instruction is in the form of static PowerPoint presentations, with little tailoring to individual soldier needs. With the ever-changing landscape of full spectrum operations, today's soldiers are facing ill-structured problems and have little time for the ideal levels of reflection and repetition needed to promote critical thinking, adaptability, and mastery of complex skills. Additionally, the current time frame for updating courses (3 to 5 years) does not support the modern Army's fast-paced learning needs.

In pursuit of more powerful training tools, the US Army Research Laboratory (ARL) has sponsored research resulting in the Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden, 2012; Sottilare, Holden, Goldberg & Brawner, 2013), an open source architecture to lower the skills and time needed to author, deliver, and evaluate adaptive instruction. To enhance the content authoring and management capabilities of GIFT and other instructional frameworks, ARL has sponsored research into a Social Media Framework (SMF) that enables organizations to crowd-source and crowd-vet new learning content and improvements to existing courses. The research questions we seek to answer in our current research include the extent to which the SMF and GIFT can: (a) promote critical thinking, collaboration, adaptability, effective communication, and problem solving; (b) help close the gap between formal training and operational application of the training to missions in the field; (c) reduce the time required to locate and use learning resources; (d) reduce the time required to incorporate feedback from the field into formal instruction; and (e) reduce instructor workload, while maximizing the efficacy of the instructor's time.

BACKGROUND: SOCIAL MEDIA FRAMEWORK

Previously, we investigated a research-based suite of affordances that support the sharing and vetting of information amongst peers. The objectives of the project were to: identify lessons learned from commercial, academic, and US Government applications of social media to knowledge management and learning; and consider the unique requirements and constraints of the military learning environment and how successful commercial and academic models for learning can be adapted to military applications.

CURRENT RESEARCH

Research Objectives

At a high level, our research aims to investigate the extent to which SMF integrated with GIFT can do the following:

- Promote critical thinking, collaboration, adaptability, effective communication, and problem solving within adaptive instruction
- Help close the gap between formal training and operational application of the training to missions in the field
- Reduce the time required to locate and use learning content and resources
- Reduce the time required to incorporate feedback from the field into formal instruction
- Reduce instructor workload, while maximizing the efficacy of the instructor's time

Experimental Methodology

This research project has followed a sequence of overlapping/spiral events, including a literature review (ensuring that our proposed research furthers the body of knowledge), an experiential review (hands-on examination of existing tools to ensure that the affordances we test are extending the state of the art), test bed development (creating the suite of affordances to enable testing of our research hypotheses), and quantitative and qualitative research (testing our hypotheses and soliciting feedback from participants).

Test Bed Architecture

Prior to the creation of GIFT Cloud, we expanded the SMF to provide a cloud-based, "headless" instance of GIFT, allowing multiple users to connect to GIFT across the internet (Figure 1). In this configuration, we run server-only instances of GIFT, the Nuxeo content management system (CMS), and ActiveMQ, which allow us to provide an entire GIFT instance to multiple users, without the need for dedicated desktop systems.

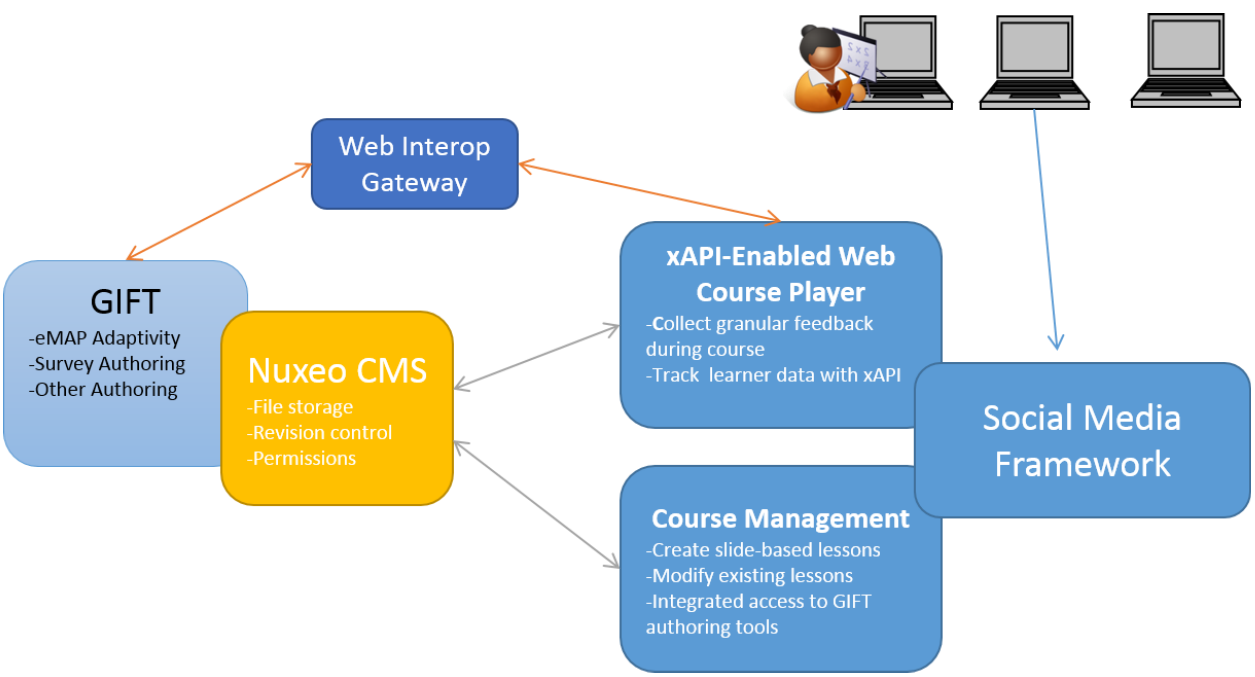


Figure 5. SMF/GIFT Integrated Architecture

GIFT was also extended to include a gateway interoperability module that allows connection to a web-based course player. The course player, suitable for expansion to mobile devices, plays course content that automatically generates experience application programming interface (xAPI) statements for tracking the learner's interactions (Advanced Distributed Learning, 2013). The course content is stored in the Nuxeo CMS, which provides revision control mechanisms. A SMF-based front-end allows for simplified course creation and management, adding the ability to author an entire web-based course. Using Nuxeo in this way allows us to leverage the GIFT toolset, which also uses Nuxeo, to tie the two systems together, so that they can share learning assets and access controls. Through the gateway interoperability module, the course player communicates to the GIFT Engine for Management of Adaptive Pedagogy (eMAP), allowing adaptivity within the course driven by GIFT's advanced adaptive capabilities. The web-based course player includes the ability for courses to collect social media feedback on granular aspects of the course (e.g., paragraphs of text, images, videos, etc.).

Using annotation-style commenting, the feedback is collected and stored within the SMF for crowd-comment and review after the course is completed. In addition, the GIFT user interface (UI) has been modified to allow other GIFT transitions (surveys, learning materials, after action reviews) to collect feedback in a similar manner. This feedback is also made available within the SMF for crowd comment and interaction.

Experimental Research

Our research in social media-enabled learning and knowledge management includes three major phases, each with a data collection. In 2015, Data Collection 1 focused on instructional systems designers (ISDs) and subject-matter experts (SMEs) using a learning content management system (LCMS) to enter content and build a course. Data Collection 2, conducted in summer 2015, involved learners taking the course and providing granular feedback about how they think the course can be improved as well as using social media tools to discuss the feedback of others. In Data Collection 3 (Spring/Summer 2016), the ISDs and SMEs will review the feedback from learners and decide what improvements they will make to the course. They will then be able to use the SMF to update and republish the course based on the learner feedback.

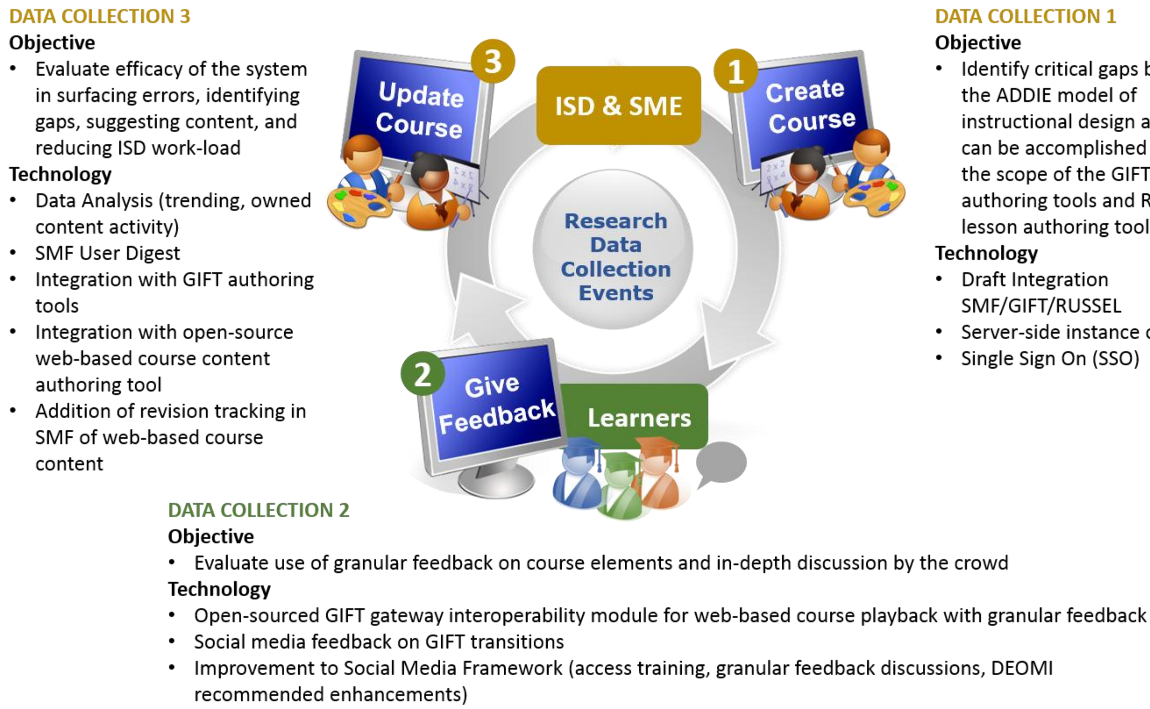


Figure 6. Experimental Research Overview

This three-part research demonstrates the speed with which experts in the field and fleet could provide real-world feedback that could then promptly incorporate changes into the official course by the schoolhouse. This addresses key goals within the Army Learning Model (ALM; TRADOC, 2011), which seeks, among many other goals, to include the ever-evolving knowledge from the field into official training as quickly as possible.

Data Collection 1 Procedure

At the time of this data collection, GIFT ran as a desktop application. Expanding on the existing SMF, a cloud-based, "headless" instance of ARL's GIFT platform was created, which allowed GIFT to run independently of a specific workstation. Utilizing this configuration, we deployed the GIFT Survey Authoring System (SAS) and GIFT Course Authoring Tools (CAT) through our Apache Tomcat web application server. Using nginx to serve the existing SMF and act as a proxy to the GIFT instance on the same server, gave the participants the experience of a seamless, consolidated system with Single Sign On (SSO) for each subsystem. The experimental test bed was hosted on a dedicated server off site from the research location. Each participant received login credentials and used a separate work station in their lab to access the test bed through the internet from a standard browser.

The researchers guided participants through standard tasks involved in creating learning content. The participants were encouraged to comment on the experience and compare and contrast it to the tools and processes that they typically use as ISDs and SMEs. The session was videotaped to allow for detailed analysis afterward. The researchers described the system to the participants as an experimental learning content authoring system for the Army and that the long-term goal was to grow the system into a powerful tool that is useful to them (and other users) in creating adaptive learning experiences that are easy to update. The researchers also noted that having their formative feedback at an early stage would help guide development in the direction that's most useful to users. Their data collection experience was designed to simulate a collaboration to create the course. So, each participant was asked to create a different scenario and then we had them work together to tie it all together into a complete course.

Data Collection 1 Results

Each of our recommendations has its basis in the time-tested and research-proven principles of UI and User Experience (UX) professions. Our recommendations are meant to help move GIFT closer to its goal of being useful to SMEs who want to author effective courses on their own. The Nielsen/Norman Group of UI/UX professionals defines useful as the result of utility and usability (Nielsen, 2016). Utility speaks to the extent that the system has the features the user wants and needs. Usability can be described as having 5 criteria: (1) easy to learn to use; (2) user can complete tasks quickly; (3) user can remember how to use it after being away from it for a while; (4) errors the user makes are few and easily rectified; and (5) the system is enjoyable to use.

Recommendation 1: Sell the utility, immediately. Users found that the system contained a large number of steps compared to other systems they had used to build adaptive training or surveys. Some of those steps were unclear in meaning or purpose. The naming conventions used are not consistent with what SMEs would name the features, buttons, and other controls. As a result, they expended a great deal of mental effort (cognitive tolls) to work in the system. Although the researchers explained the long term purpose of the system (to creative adaptive training suited to each individual), the perceived benefits of the system were not sufficient to motivate the users to want to continue using the system in its current state. For all of these reasons, we recommend an early intervention of “selling the utility” – making the benefits of the system so clear that new users will be motivated to expend the needed effort to understand and master the system.

We recommend the system provide a short but impactful explainer video that helps users understand the system and what’s in it for them. Specific questions that should be answered include: (a) What is Adaptive Learning?; (b) Why should I use Adaptive Learning with my learners?; (c) What is GIFT? And, why is it better than my other options?; (d) How have others similar to me used it (compelling real success stories/visuals)?; and (e) How do I use GIFT to create Adaptive Learning?

The military has a long-standing tradition of rigorous ISD, which follows a standard ADDIE model (analysis, design, development, implementation, evaluation) of activities. We can reasonably expect a SME to have extensive knowledge of the content being taught. Based on their experience, they may also bring knowledge of the audience (having been a trainer) and the related organizational goals that lead to the SME being asked to share their knowledge. However, there are significant knowledge gaps in ISD for most SMEs. To achieve the long term goal of an independent SME creating effective training, the system must provide the education and support needed by the SME.

Recommendation 2: Use the process and vocabulary native to the SME. The current process flow and vocabulary used in the system is not reflective of how most SMEs think or work. As a result, they are burning significant brain power simply trying to understand the system rather than feeling the reinforcement of accomplishing their goals. To illustrate both of these concepts, we examined a short process – adding a question to an assessment – comparing how SMEs typically do it with how SMEs attempt to do it in GIFT.

For this very short sub-process of the larger course creation process, we can compare the GIFT experience versus the typical SME experience using the scorecard shown in Table 1.

Table 1. Cognitive Load Comparison

Measure	*GIFT Experience*	*Typical SME Experience*
Steps	20 (steps 7-9 repeat 3X)	9 or less*
Cognitive Load	High	Low
Time	Slow	Medium
Other	Process incomplete. Feedback to be added using additional steps, time and cognitive load in another part of GIFT.	* Ability to up-load can make process even shorter.

* Ability to upload can make the process even shorter.

Recommendation 3: Incorporate extensive yet lean, on-demand contextual support for SMEs. We recommend two approaches to provide support. First, provide SMEs some fast and simple support when they first arrive. This help should display automatically the first time the user experiences a screen.

Afterward, it should be available for the user to display on demand. Second, offer mouseover-based help for each control, vocabulary term or other element that the SME might not be familiar with. The example in Figure 3 shows that a vocabulary improvement has been made – changing the word Transition to Content, and then providing a mouseover that explains what particular types of content are and alerting the user if they will need to use another part of the system to create that content before trying to use it here.

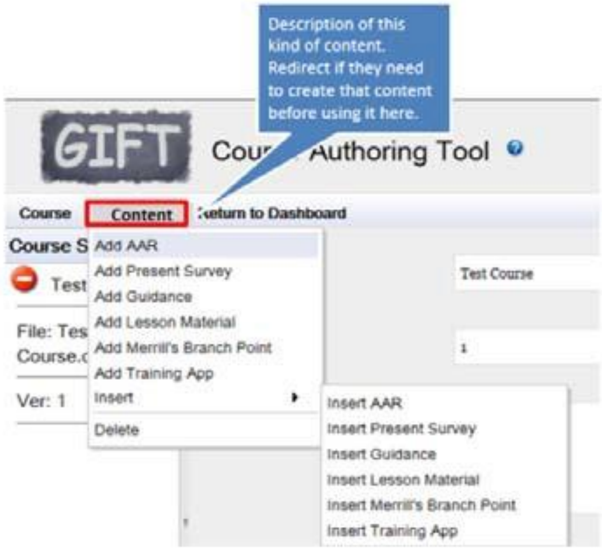


Figure 7. Mouseover Help Example

Data Collection 2 Procedure

For the second data collection, the SMF was expanded to include course topics and actual course materials, accessible from the "training" tab. Once launched, the course was played through the GIFT framework. In GIFT, a course is a series of transitions, which might include surveys, learning materials, and training applications. To enable a training application to play lessons comprised of web-based content, we implemented a new gateway interoperability module. Unlike standard web-based lessons, however, any element of the content can be selected and commented upon (Figure 4).

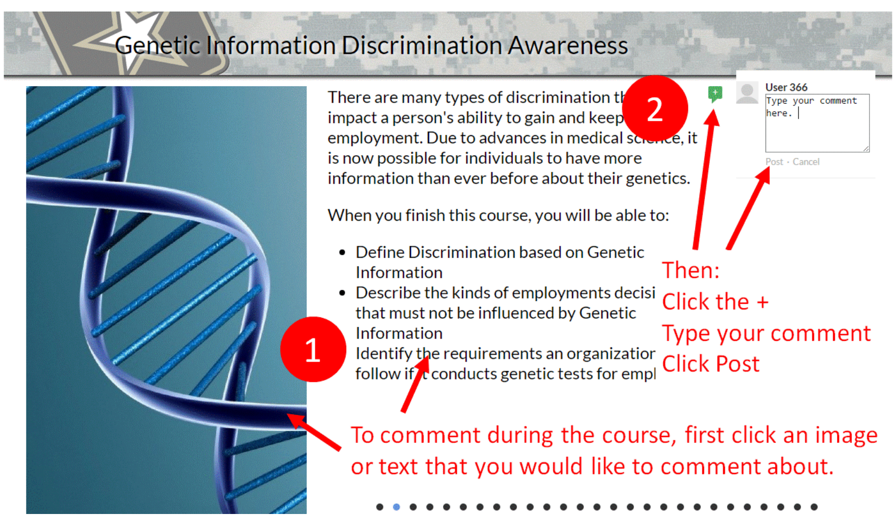


Figure 8. Granular Feedback within Course

Showing those comments in close proximity to the lesson content could negatively impact the flow of the course for future learners; so instead, the comments automatically appear as a new conversation thread under the feedback tab (Figure 5) of the surrounding topic page for this course. We added similar social media commenting capability to other GIFT transitions including surveys and learning materials.

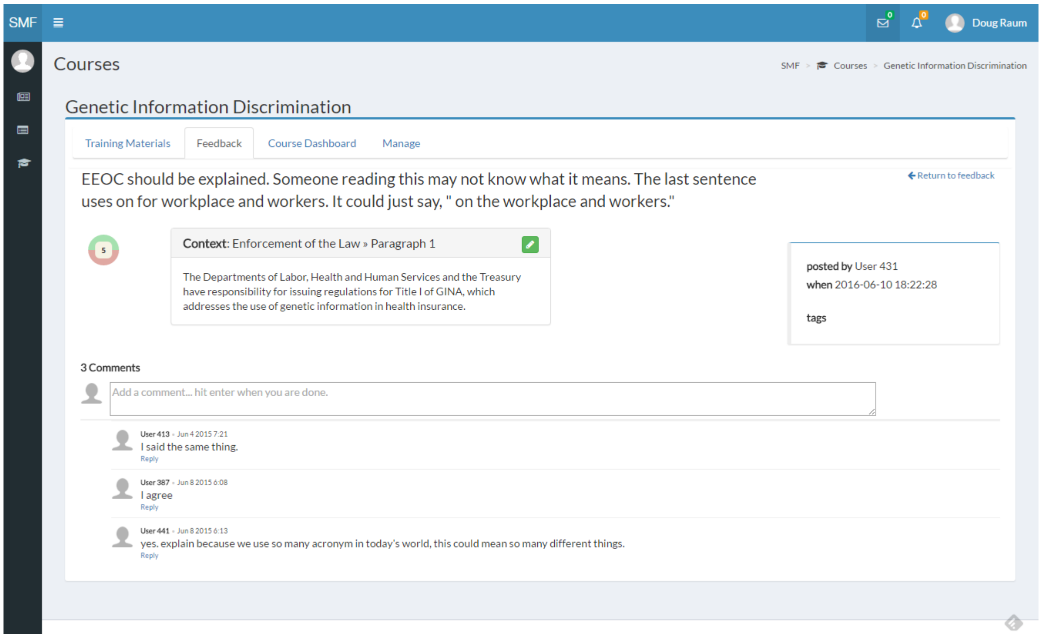


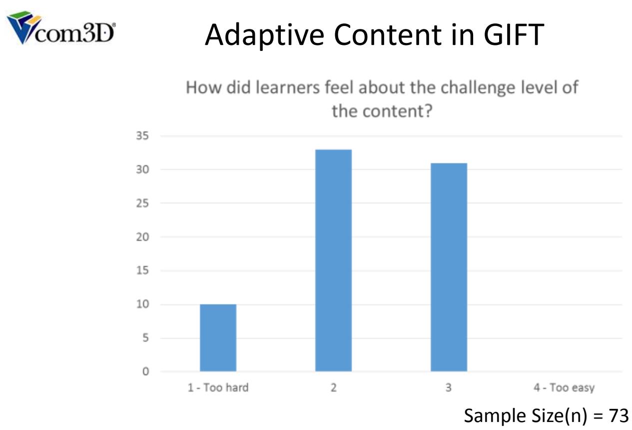
Figure 9. Learner Voting and Discussion on Course Feedback Page

The course material was developed by Vcom3D for specific use in the experimental research and reviewed by ISDs for relevance to the target student participants. The content was then prepared for playback with the web-based training application and other GIFT transitions. As part of course development, we created two paths through the course – one for novice learners and one for people more familiar with the material. Based on pre-test scores (using GIFT's survey engine), the learners were presented with content matching their level of knowledge. This allowed us to make use of GIFT's adaptivity in a simple way, but highlighting the potential of the tools. In addition, a pre-test survey was used to collect demographic data. This demographic data was used to present a different look and feel based on the learner's branch of military service.

Data Collection 2 Results

The data collection involved 73 students taking an online course through our modified GIFT instance, providing granular feedback on the course content, and commenting on the feedback of other students through the SMF. During the data collection event, multiple sessions of approximately 20 student participants accessed the experimental test bed from work stations in their lab through the internet using a standard browser and credentials provided by the researchers. Participants were asked to navigate to a particular topic and take the course associated with that topic. Participants were encouraged to generate questions or feedback on any content they encounter. After completion of the course, participants reviewed their comments on the topic page and also saw the comments of other participants. They were able to up vote and down vote the questions, answers, and feedback generated by others as well as contribute to the discussions. Participants in subsequent sessions reviewed the accumulated contributions of all preceding participants. At the end of each session, the participants completed a survey to provide feedback on their experience.

During the sessions, we received hundreds of original and follow-on comments from participants. Analysis of results showed that learners do experience problems with learning content in general; they liked the personalized look of their course (based on their service branch); and they felt the challenge level of the adaptive content was appropriate (Figure 6). They found the ability to comment within the course and within the SMF to be intuitive and easy.



Mean	2.29
Median	2
Mode	2

Figure 10. Learner opinion of adaptive content

Data Collection 3 Procedure

The third phase of research will explore techniques and algorithms for analyzing the user-generated content, surfacing the most relevant comments and activity, and relaying them to the most relevant stakeholder. For this data collection with content authors and content owners, the user management section of the SMF will evolve to display a "User Digest" specific to each user and their role in the system. An Activity section will highlight the latest contributions by the user. Back-end data analytics will look at factors such as up votes, down votes, and general activity to prioritize the feedback most relevant to this user. The goal is to highlight trending and actionable issues pertaining to course content owned by this user. Participants will then evaluate the efficacy of the system in surfacing errors, identifying gaps, suggesting content, and reducing ISD work-load.

After reviewing student feedback, participants will then be encouraged to use the updated SMF-based tools to update and re-publish the course content, with a goal of determining the effectiveness of rapidly turning learner feedback into actionable content updates and making those course improvements immediately available to new learners.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

At the end of the third phase of the current research, we will have investigated the efficacy of crowd-sourced and crowd-vetted content for applying field knowledge to improve learning content, while reducing instructor workload and turn-around time. However, we believe that social media can provide additional benefits to the learning environment, and to GIFT in particular, by (1) harnessing crowd inputs for the creation and refinement of a domain model, or the body of knowledge for a topic, and (2) mining social media data to enhance an individual's learner profile (or personal history of learning, demographics, and achievements). We have also identified the need to make the user experience more intuitive to its intended end-users (SMEs). At the end of the current research, we will make recommendations for these additional means for applying social media to the integrated learning environment. Additional areas of research that could be explored include: (1) harnessing crowd inputs into the creation and refinement of a domain model, or the body of knowledge for a topic; (2) mining social media data to enhance an individual's learner profile (or personal history of learning, demographics, and achievements): and (3) developing the user experience to be immediately intuitive to its intended end-users (subject-matter experts in the field).

REFERENCES

Advanced Distributed Learning. Experience API: Research Summary. Retrieved May 13, 2013 from http://www.adlnet.gov/tla/experience-api

Nielsen, J. Usability 101: Introduction to Usability. Retrieved March 15, 2016 from http://www.nngroup.com/articles/usability-101-introduction-to-usability/

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Sottilare, R., Holden, H., Goldberg, B., & Brawner, K. (2013). The Generalized Intelligent Framework for Tutoring (GIFT). In Best, C., Galanis, G., Kerry, J. and Sottilare, R. (Eds.) *Fundamental Issues in Defense Simulation & Training*. Ashgate Publishing.

US Army TRADOC (2011). The US Army Learning Concept for 2015. Retrieved November 15, 2012 from http://www.tradoc.army.mil/tpubs/pams/tp525-8-2.pdf.

ABOUT THE AUTHORS

Mr. Rodney Long *is a science and technology manager at ARL, Human Research and Engineering Directorate, Advanced Training and Simulation Division in Orlando, FL. He has a wide range of simulation and training experience that spans 28 years in the Department of Defense (DOD) and has a Bachelor's degree in computer engineering from the University of South Carolina and Master's degree in industrial engineering from the University of Central Florida*

Dr. Irene Boland *is the managing director of the Learning Development Institute, Inc. Previously, she was the Learning Development Director at Vcom3D. She has over 12 years of hands-on experience in applying emerging education science to real-world issues faced by enterprise organizations. She leverages her PhD in education in combination with proven social media technology to create innovative methods for accelerated learning and knowledge transfer. Her expertise enables clients to solve performance issues, maximize use of resources, and improve profitability.*

Mr. Doug Raum *is a software developer at Vcom3D, an Orlando-based company that develops innovative mobile, Web, and game-based learning. He has over 19 years in the information technology field in a variety of positions, with 12 years of software development and a focus on enterprise web application development, as well as a further background in systems, security, and infrastructure design/administration.*

Mr. Dan Silverglate *is Director of Software and Graphics at Vcom3D. He has over 15 years of experience in digital media production and software development. At Vcom3D, he leads the development of serious games for education and training featuring lifelike virtual humans. He holds a Bachelor's of Art in film studies from the University of Florida, where he graduated with high honors and was inducted in the Phi Beta Kappa Honorary Scholastic*

Society, and a Bachelor's of Science in computer science from the University of Central Florida, where he graduated Magna cum Laude.

***Dr. Ed Sims** is Chief Technology Officer of Vcom3D. Since 1997, his work has focused on the application of web and game-based technology for simulation, training and education. Prior to co-founding Vcom3D, he was Technical Director for Lockheed Martin Information Systems Company. He holds a Bachelor's degree in mathematics from the College of William and Mary and Master of Science and PhD degrees in systems engineering from Rensselaer Polytechnic Institute. He has been awarded five patents in the areas of real-time modeling and simulation of human behavior.*

The Hidden Challenges of Team Tutor Development

Desmond Bonner[1], Anna Slavina[1], Anastacia MacAllister[1], Joseph Holub[1], Stephen Gilbert[1], Anne M. Sinatra[2], Michael Dorneich[1], Eliot Winer[1]
[1]Iowa State University, [2]U.S. Army Research Laboratory

INTRODUCTION

This paper describes the unexpected challenges of team tutor development such as the task and logistics. Previously, a research team from Iowa State University (ISU) working with the U.S. Army Research Laboratory (ARL) developed the reconnaissance (Recon) task for simple team tutoring with the Generalized Intelligent Framework for Tutoring (GIFT) (Bonner et al., 2015; Gilbert et al., 2015). Considerations were included for the testing environment such as audio-based team interactions, initialization of the scenario simultaneously, and the inclusion of eyetracking and screen capture technology. Throughout the process of tutor development, several computational challenges have been encountered such as the implementation of team rules, determination of the appropriate amount of feedback, and the use of participants' behavior history as input to the tutor. Our descriptions of these challenges should forewarn future developers of team tutors. We also suggest enhancements to GIFT to aid this process.

Surveillance Task Development

The surveillance task was developed by using Virtual Battle Space 2.0 (VBS2) (see Figure 11). Surveillance was chosen as the military subject matter due to its scalability in large or small team environments. The task's purpose was to serve as a testbed for examining different dimensions of feedback (Bonner et al., 2015). In the task, two learners operate avatars atop a roof and are responsible for surveillance of the entire area. This surveillance consists of completing four subtasks: 1) scan their individual area, 2) identify opposing force avatars (OPFOR), 3) transfer responsibility for tracking OPFOR to a teammate, and 4) acknowledge a transfer from a teammate.

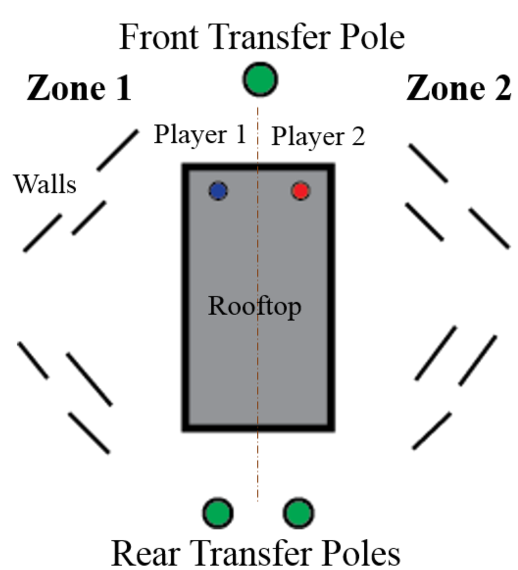


Figure 11: Surveillance Scenario Layout

Currently, the scope of the work has focused on the subject of feedback. Feedback is provided based on subtask performance (Table 1). Several GIFT conditions were developed which are further explained in Bonner et al. (2015) and Walton et al. (2015a). The conditions are also informed by work detailed in Walton et al.'s (2015b) work with the *Team Multiple Errands Task (TMET),* which deals with a team-based shopping task.

Table 2: Subtasks Performed by Learners

Sub Task	**Task Description**
Scan	The learner rotates the viewpoint within their 180 degree sector continuously throughout the task
Identify	The learner presses a key whenever spotting a new OPFOR avatar.
Transfer (notify)	When an OPFOR avatar is close to moving into a teammate's assigned sector, the learner must inform the team member.
Transfer (acknowledge)	The learner must acknowledge transfer of responsibility for the incoming OPFOR from the teammate who initiated the transfer process.
Measures	**Capture Method Description**
Degree of Scan	The position of the learner's viewpoint must pan the full 180 degrees of their assigned sector every 10 seconds; if not, feedback will be sent to them via GIFT and recorded via log files
ID Button Press	Button press logs will show if the ID key was pressed within 10 seconds of each individual OPFOR appearance
Transfers	Button press logs will show if the transfer key was pressed at the correct OPFOR distance from the transfer poles
Acknowledgments	Button press logs will show if the transfer was acknowledged within 10 seconds

In the surveillance task, the Scanning subtask is the first and most common task performed. Scanning is measured by how much of the environment was seen by the learner over a given amount of time. This can be measured through mouse movement and panning across the screen. It primarily serves as an individual task to make sure that the learner is consistently surveying their assigned area.

The Identify subtask also serves as an individual task. Identifying targets was operationalized as pressing the key associated with identification on the keyboard whenever an OPFOR is spotted. This is the most important of the tasks as it serves as a basis for the others. Participants scan for the purpose of identifying OPFOR. An OPFOR cannot be transferred if it is not first identified.

Transferring and Acknowledging are individual tasks which when paired, constitute a team task. Transferring was operationalized as one team member indicating to another team member where an OPFOR is going to appear. There were two transfer points on opposite sides of the environment (the one-pole area and the two-pole area, shown in Figure 11). Transfer performance was measured in terms of where the OPFOR was when the transfer was initiated. The learner must transfer when the OPFOR is an appropriate distance from the pole boundary: not too close and not too far. The acknowledgement of transfer was operationalized as a button press in response to a transfer initiation. Acknowledgement performance was measured by calculating the time that elapsed between the transfer and acknowledge button presses.

Currently, learners complete the surveillance task as distributed teams: each participant is located in a separate office of a secure laboratory as depicted in Figure 12. The task is completed on desktop computers with wireless headsets, a wrist electrodermal activity (EDA) sensor, and a speakerphone for communication with their team. Additionally, a separate laptop is included for participants to complete the accompanying electronic surveys. A participant designated as Player 1 has a desktop that comes equipped with an eyetracker.

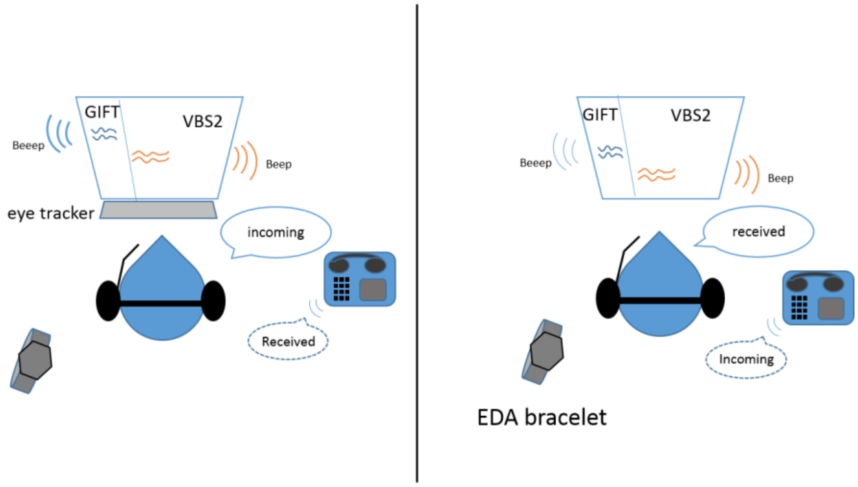


Figure 12: Participant Environment

Prior to arrival, participants complete an informed consent document and a pre-experiment survey. Once they arrive on the day of their session and the teammates meet, they complete a team familiarity survey and proceed to their assigned location as Player 1 or Player 2. Both undergo EDA calibration and training. Player 1 also is calibrated for the eye tracker before the study begins. Within the study, participants complete the task four times and complete two surveys after each trial. Each trial takes 5 minutes to complete, and within a trial, the tasks become difficult as more and more OPFOR emerge. The task is designed to be difficult to complete perfectly. Finally, participants complete a post-experiment survey at the conclusion of the session.

Surveillance Task Testing

To effectively develop the system, a user centered approach was adopted (see Figure 13) (Nielson, 1993). This consisted of gathering requirements needed, designing based on the requirements, implementing the design, evaluating with pilot tests, and iterating. First, requirements were gathered from research described in ISU's previous GIFT publications and ARL's extensive work on team tutoring (Bonner et al., 2015a; Walton et al., 2015a; Sottilare, Holden, Brawner & Goldberg, 2011). From this, the surveillance task was mapped out.

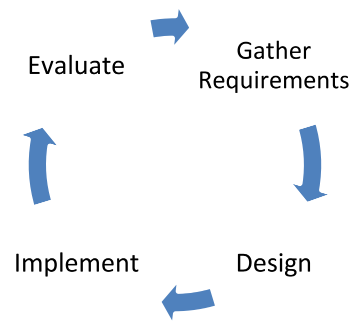


Figure 13: Iterative Design Process

Fourteen pilot tests have been conducted to test the scenario, study environment, and participants' interactions. During these tests, flaws were found which ranged from simple issues such as ergonomics in button placements to the frequency of feedback. For example, in early pilot test, researchers were able to determine an ergonomic improvement to the keyboard key assignments so that participants could make all key presses easily with the left hand while controlling the mouse with the right.

DEVELOPMENT CHALLENGES

Logistic Challenges

Audio: An experimental setup was created in which two teammates, would each have their own laptop with the simulation, and could communicate by voice. This sounded easy enough, but experimentation revealed the audio signals from one teammate's simulation would interfere with the cues of the other teammate's simulation. Wearing headphones for computer audio prevented participants from hearing each other. The participants were separated to help with interference and provide a more realistic training situation, since distributed teams might want to train together. Participants were placed in two different rooms, each with a laptop running VBS2, and made a Google Hangouts voice call between the two rooms. However, as piloting continued, it was clear that this did not solve the problem. Each participant needed to be able to hear the computer audio because it contained information about the simulation as well as periodic beeps from GIFT when feedback was presented. Each person also needed to hear communication from the teammate. But if there was an open audio channel, the audio from the computer of one teammate would go through that channel and interfere with the audio of the teammates' computer. In an initial pilot, with Google Hangouts running as well as the simulation, for example, each teammate heard audio from his or her own simulation, audio from the teammate's computer simulation, and the voice of the teammate. The set up was changed from an audio channel on the computer to a speakerphone call between the two rooms. Headphones were used on one ear only for the computer audio. Therefore one ear of each participant heard the computer, and the other ear heard the audio signal from the speakerphone. This workaround was sufficient for our task, but it seems a little bit clunky, and it is possible that with more complicated team tasks, it may be more difficult to set up audio logistics.

Screen Capture: An additional logistic complexity was capturing a screen recording of each participant's screen while also running GIFT and the simulation on the computer. Screen capturing software such as Camtasia can be CPU-intensive, and running it simultaneously with the complex VBS2 simulation required an especially powerful computer. To fully utilize video and voice recording without overloading the PCs, two Google Nexus cellular phones are placed by the speakerphone in each room to record audio. Currently, Camtasia is used to record Player 2's screen while eye-tracking software records Player 1's.

Simultaneous Start: Another complexity with team tutoring is starting the participants at the same time, so that the systems logging time-stamped data about their behavior have the same time start zero point. To mitigate errors, a three-pronged approach was implemented. First, code was built into the VBS2 scenario so that both participants press the enter key to initialize the scenario. When the scenario first loads, both participants are provided with these instructions via text. To coordinate, the experimenter uses the intercom system to make sure both participants are ready and then instructing both to press the key at the same time via a three second countdown. Lastly, code was added to GIFT so that the GIFT server does not start offering feedback until the client scenario is started. Otherwise, while setting up the task, the GIFT server would be active before the VBS2 scenario, telling learners they are not scanning while the VBS2 scenario had not yet started.

Teammate Communication: Initially participants were to communicate verbally using only a small set of prescribed military-style phrases such as “I have movement in my sector” and “Be advised, two OPFOR transferring to your zone”. This approach would lead to clear communication, and would also allow for easier data coding of communication later, e.g., “Participant 1 said Phrase 3.” However, in pilots, it was discovered that the task was stressful enough that participants did not have the cognitive load to keep to the unnatural prescribed phrases. The mostly undergraduate, non-military participants could no doubt learn the phrases if a significant portion of the experimental time was allocated to training them, but it was decided that the extra time during the study was not worth the potentially cleaner communication data. Thus, they were allowed to communicate openly.

Early pilot participants with military experience did prefer the provided statements over open communication as they were used to the terminology. When allowed to perform open communication, they added to the military phrases for specificity such as close/medium/far when indicating zone transfers.

Piloting Feedback Pre-GIFT Using Wizard of Oz Approach

Because it took multiple weeks to develop the appropriate condition code and feedback rules within GIFT for the Surveillance Tutor, that time was used to conduct pilots of the team experience with the simulation. An aim of the research was to experiment with the feedback statements and feedback timing that was being considering before coding them into GIFT. To pilot the feedback, Google Hangout was used, creating a chat window on screen that was shared by each participant and the researchers. Using this tool, the researchers could stand over the shoulder of the participants and type feedback to the participant as if the tutor were doing so, a Wizard-of-Oz prototyping technique (Kelley, 1984). It was useful to have a textfile open on the researcher’s screen with all the possible feedback statements that were drafted. This method worked well for exploring whether the feedback statements were understandable and possibly too long to fully attend to during the task. However, since the human researchers were playing the role of the tutor, and the actual tutor would respond to a diverse set of encoded conditions in the future, it was difficult to simulate the exact timing and frequency of the future feedback. It was also difficult to simulate rare conditions, i.e., when the participant’s behavior meets two sets of feedback conditions simultaneously, and the tutor might possibly issue two unrelated feedback statements at close to the same time.

It was also difficult to simulate team feedback in our distributed context, because participants were in two different rooms, so even though one researcher was standing over the shoulder of each participant, it was sometimes difficult for them to assess the team’s performance overall since each researcher could see only one participant directly. There was no way for the two researchers to coordinate quickly enough (e.g., “I’ll give the next team feedback, so you don’t have to”). The best approach here was for one researcher to be in charge of team feedback, but that researcher then had to rely on indirect information about the other teammate’s performance. In the future, a GIFT live testing feature might enable similar testing, with feedback statements assigned to function keys that the researcher could press to make the tutor issue a feedback statement. The live testing feature could automatically log all feedback statements made and give the researcher a report afterward, noting data such as “Feedback 1 used 45 times (50% of all feedback), Feedback 2 used 20 times (22%)…” etc.

Eye Tracking Pilots: Too Much Feedback?

One of the challenges with issuing feedback to the participants was that the simulation itself had communication messages that appeared regularly about what was happening within the simulation, e.g., “Target identified” or “Target Transferring at 1 Pole”. These were textual messages that appeared in the simulation window at the lower left and sometimes were accompanied by audio cues. At the same time, feedback in the GIFT window would also appear in the side window, sometimes accompanied by audio cues (Figure 14). It was important for us to figure out whether participants could pay attention to both sets of messages, and whether they understood the different roles of those messages. The messages within the simulation were simply status updates: here’s what’s going on right now. But the messages in the GIFT feedback window were meta-messages, urging behavioral change or encouraging participants.

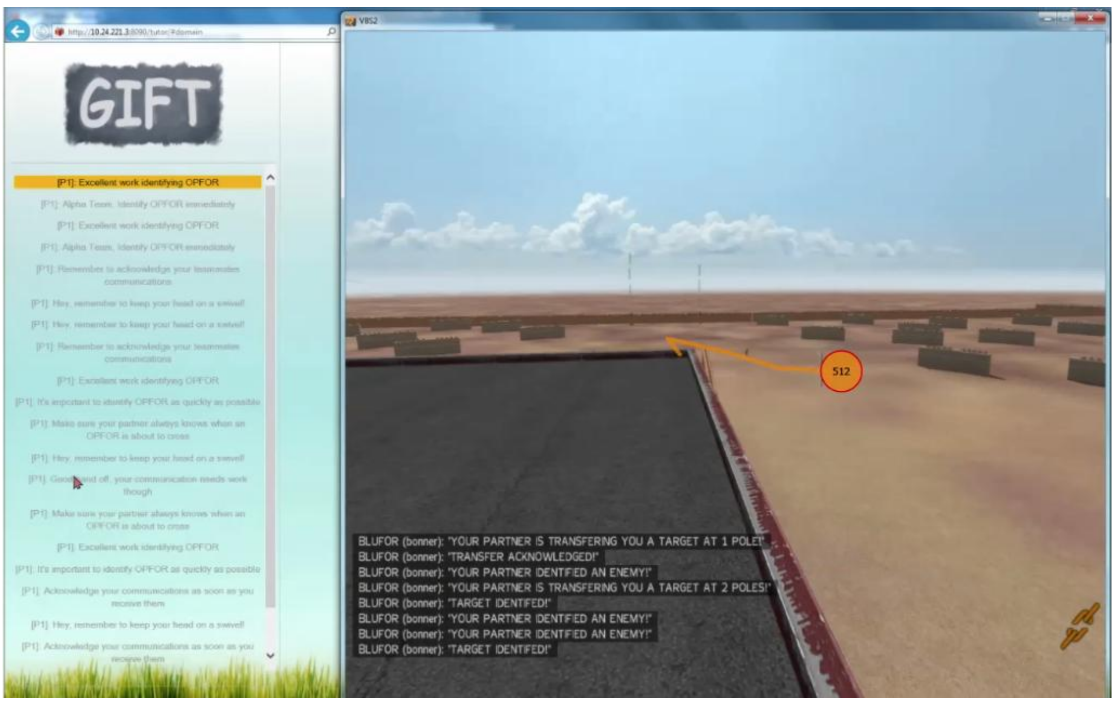


Figure 14: Too much feedback. This picture shows a pilot experiment using eyetracking. After less than a minute of the task, many feedback messages accumulate at left and status updates appear in the scenario window, while participants mostly ignored all written communication and focused on the task. Future pilot experiments decreased feedback and increased font size and readability of messages to lower cognitive load.

It was shown that participants in pilots could attend to both sets of messages if they were short and relatively infrequent. Pilot eyetracking data revealed that when feedback messages were long, and they accumulated down the screen (15+ messages), participants no longer attended the feedback. In later pilots less frequent feedback was explored, as well as presentation variables such as font size, color contrast, and the maximum number of messages to display.

Team Feedback: Public or Private?

In a team setting, feedback given mid-task can generally be directed to team members in four basic ways:

1) **Private**: The individual teammate receives feedback privately about his or her performance, e.g., "Player 1, you need to…"

2) **Public Anonymous**: The entire team receives feedback about each individual's performance anonymously, e.g., after Player 1 makes a mistake, all members receive, "Team, you need to …"

3) **Public Identified**: This could be called the shaming approach. The entire team receives feedback about each individual's performance by name, e.g., after Player 1 makes a mistake, all members receive, "Player 1, you need to…"

4) **Team Feedback**: This feedback is about team performance overall (rather than an individual's performance) and is sent to all team members, e.g., "Team, your frequency of communication needs to decrease."

GIFT currently doesn't support these four modes by default, but for our system it was configured to support modes 1, 2, and 4. One pilot participant was more responsive to public identified feedback, reminded of positive experiences on sports teams. Meanwhile, the partner disagreed, felt publically shamed, and preferred private feedback. Similarly, survey responses across multiple pilots suggest varying efficacy for feedback mode based on previous experiences with teams. Previous research on these approaches yields mixed results (Walton et al., 2015a), suggesting that the best approach may vary by team member's team skills and by the task at hand.

Experimental Design Challenges with Team Tutors

Common challenges for running robust objective clinical studies on team performance are 1) controlling for a similar team experience across multiple consecutive within subjects trials (i.e., handling teams' learning curves and ensuring the task is not repetitive), and 2) controlling for the influence of the team dynamics of particular participant teams (familiarity of members, varying team skills, personalities, etc.). Both of these factors lead generally to a high amount of variance in data collected, which means that studies must include larger sample populations and very simple experimental designs (typically one independent variable) to achieve statistical power. In the surveillance task the first challenge was approached by using multiple similar but different scenarios with the same number of OPFOR. Also, team were discouraged teams from discussing strategy between trials. Doing so might lead to non-linear improvements in the learning curve of the task. To address the second challenge, extensive questions were asked about team experience and team preferences in our surveys, hoping to use those data to factor out impact on our dependent measures during analysis.

A third challenge in team tutor studies, if one is interested in evaluating whether the feedback helped performance, for example, is that different teams may receive dramatically more feedback than others, based on their baseline performance. It is important that the dependent variables, as much as possible, do not depend on the amount or exposure to feedback. In the Surveillance task, after piloting and noticing this challenge, it is this efforts aim to have infrequent but impactful feedback.

Computational Challenges

Challenge 1: Assessment Inheritance

When authoring any intelligent tutoring system, the question arises of the granularity of assessment. In algebra, one might ask, "Does the student know how to solve for a variable?" (higher level) or one might ask, "Does the student know how to divide both sides by the coefficient of x to isolate x?" (lower level). In a simulation tutor, one might assess, "Did the trainee reach the checkpoint on time?" (higher level), or one might assess, "Did the trainee march from A to B, crawl prone from B to C, and then jog from C to D, and arrive at D on time?" (lower level). In these examples, the more general assessment could be derived from the more specific. If you know how the trainee marched, crawled, and jogged, you can answer the question of whether the checkpoint was reached on time. We suggest that this ability to derive higher level assessments is analogous to computational inheritance between parent and child classes. The child assessment (more specific, grounded in concrete behavioral markers) is derived from a parent assessment (higher level, focused more on overall performance).

The challenge in a team tutoring context arises because there are high level team measures that one would like to identify that seem to have no child assessment from which to draw information. In the surveillance task, for example, it is important that the team be good at identifying OPFOR. Identifying can be a team measure, because it is beneficial and informative to compare whether Team A vs. Team B is better at identifying. That team measure for identifying (parent assessment) is likely a simple function of individuals' identifying performance (child assessment), e.g., a weighted average, so team identification is not difficult to assess. However, with the team construct of backup behavior (to what extent does one teammate notice that the other teammate needs help temporarily and pitch in to offer support), another parent assessment, there is no child assessment at the individual level that one can use to provide detail for the team measure. Instead, the team measure must be separately assessed using its own behavioral markers which are likely separate from what is being measured for individual performance.

In GIFT, for the surveillance task, a DKF file exists for each player and a separate DKF file exists for the team, so that there can be individual assessments and team assessments as needed. This approach does not scale well, because for a team of five members, if you want to assess individuals, interactions of pairs of teammates, trios of teammates, quartets of teammates, and the entire team of five, you would need 30 DKFs. If you just wanted to assess the individuals and the team, you would need six DKFs. In either case, it would be attractive if GIFT facilitated the assessment inheritance concept so that a team DKF measure could be a parent of the individual DKFs when appropriate, so that code would not have to be redundant within DKFs.

Challenge 2: Length of Assessment Window

When team members attempt a task, they are immediately assessed. That moment is designated as an event level assessment. In GIFT's standard assessment framework, an event is evaluated as Below Expectation, At Expectation, or Above Expectation. However, for the experiment to have more granularity for the assessment was desired as well as in what would cause the individual’s overall state to change. Therefore rules were written based on an individual’s assessed events that would allow us more flexibility in state changes, and provide an overall state of Below Expectation, At Expectation or Above Expectation. For example, it might be that if an individual has five consecutive Above Expectation events, then the individual's overall state for that task can be set at Above Expectation.

This approach to assessing an individual's state (or a team's state) based on a historical moving window of assessment events is a key concept that feels natural to want to use within assessment, but is not currently built into GIFT. In the surveillance task, different assessment windows were initially specified for different tasks and different states. E.g., for an individual to drop from At Expectation to Below Expectation in the Identify task, the individual must miss three consecutive OPFOR (fail three times at the Identify task, earning a Below Expectation event on each one). To move back up to At Expectation, the individual must achieve five At Expectation events. These thresholds of three and five are arbitrary, however, and the thresholds are different in different tasks. They are continuously adjusted throughout the pilot process.

Challenge 3: Team Feedback Rules

Independent of the intelligent tutoring software platform to be used (GIFT in this case), authoring team rules can be difficult because the complexity of anticipating the multiple possible team interactions. Also, a team's performance is not always a linear sum of the individuals' performances. The details of these challenges arise from the behavioral markers chosen to measure performance.

It is challenging to design effective team rules by themselves in GIFT without including individual rules. Since the transfer and acknowledge tasks are evaluated together, it is not necessarily an effective team evaluation of actions, but instead a series of individual actions completed in sequence.

The team is graded on whether the individuals did their part. For example, a good sports team can pass or score a point, but it is not a rich picture of teamwork. A trainer or coach is able to evaluate how an individual on a team completed tasks to reach a team goal, but can also determine how the team can work together to be more effective. Maybe one player needs to slow down to meet the needs of the others despite individually performing well. Even though Player 2 is always on time with their transfers, maybe Player 1 is at times overwhelmed and late acknowledging due to workload.

An example of straightforward authored individual rule is below. This rule that indicates that if an individual is currently Above Expectation, but based on their recent past performance is considered At Expectation for the majority of assessments, then they should be shifted to the At Expectation state and provided the appropriate feedback.

```
//Scanning rule for Individual:
//If individual state = Above and majority of events in window are At,
//then drop to At state and get At feedback.

IF state_scan = "Above" AND scan_score(scan_window) > (drop_threshold/2)
THEN:
 state_scan = "At"
 give scan_at_state feedback
```

Team rules are more complex than individual rules, and rely on the current state assessment with individual team members, as well as those team members' recent actions (event assessment). A team rule example can be found below. In our case it is accounting for the state of two members. In the case of a larger team the rule would be more complex. These rules assess the overall state of team communication by examining the transfer and acknowledging the states of the team members.

```
// Communication rule for Team:
```

```
// If either player has Above state for transferring, and
// the other player has Above state for acknowledging, then
// set team's communication state to "Above."
If player(any).transfer_state = "Above" and
   player(other).acknowledge_state = "Above" THEN:
       Team_communication_state = "Above"
       Give team_communication_above_feedback
```

Recommendations

Based on these challenges, we have several recommendations for future versions of GIFT. GIFT should allow for more extensive logging of events within the learner's interface. Currently, GIFT logs all events within GIFT (e.g., messages), but it requires custom software development to log button presses or mouse events in the third-party software during the GIFT session. For example, in our surveillance task, we developed code to record VBS2 button presses and turn them into GIFT messages so that they could be logged by GIFT. In our scanning condition, it would have been beneficial to have a way to track each time the learner moved the mouse instead of the position of the camera, but that was beyond the scope of our effort. It would also be helpful to have the GIFT feedback more tightly integrated with the software platform to provide feedback directly in the learner's field of view instead of being located off to the left. As previously mentioned, the inclusion of a live test feature would help pilot testing. Additionally, a plugin for multiple open source platforms (e.g., Unity) would allow for wider use of the tutor. Finally, team tutoring should expand to larger teams with more complex roles (Bonner et al. 2015). The DKFs should be reorganized to prevent redundancy with regard to team and individual assessment. Beyond military tasks, other avenues such as education and corporate teams could be leveraged.

CONCLUSIONS

Several challenges have been encountered in our two years of developing with GIFT but have been able to overcome a majority of them. While GIFT is a robust system, there are additions that could be made to improve its functionality for team tutoring. Team tutoring is inherently difficult to design for, particularly in a domain-independent framework. There needs to be flexibility in the approaches that are taken to construct assessments for both individuals and teams, as different domains, varying team sizes, and varying interdependencies of responsibilities may be present depending on the specific task to be taught. Feedback types and timing are very important to team tutoring, and this should be taken into consideration when authoring for teams. The work demonstrates lessons learned and ways that GIFT was utilized when developing a team Surveillance tutor. It may be helpful to expand GIFT's capabilities for assisting team tutoring design as work was included to incorporate features which are not currently present. While designing team intelligent tutoring systems is a hard problem, it is one that is achievable.

ACKNOWLEDGMENTS

The research described herein has been sponsored by the U.S. Army Research Laboratory - Human Research & Engineering Directorate (ARL-HRED). Statements and opinions expressed in this paper do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred. Also, this work would not be possible without contributions from Joan H. Johnston.

REFERENCES

Bonner, D., Walton, J., Dorneich, M. C., Gilbert, S. B., Winer, E. & Sottilare, R. A. (2015). The Development of a Testbed to Assess an Intelligent Tutoring System for Teams. In *Workshop on Developing a Generalized Intelligent Framework for Tutoring (GIFT): Informing Design through a Community of Practice* (p. 9).

Gilbert, S., Winer, E., Holub, J., Richardson, T., Dorneich, M. & Hoffman, M. Characteristics of a Multi-User Tutoring Architecture. In *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)* (p. 39).

Kelley, J.F. (1984). An iterative design methodology for user-friendly natural language office information applications. *ACM Trans. Inf. Syst.*, *2* (1) , 26-41.

Nielsen, J. (1993). "Iterative User Interface Design". *IEEE Computer* vol.26 no.11 pp 32-41.

Sottilare, R. A., Holden, H. K., Brawner, K. W. & Goldberg, B. S. (2011).*Challenges and emerging concepts in the development of adaptive, computer-based tutoring systems for team training*. ARMY RESEARCH LAB ORLANDO FL HUMAN RESEARCH AND ENGINEERING DIRECTORATE.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Walton, J., Dorneich, M. C., Gilbert, S., Bonner, D., Winer, E. & Ray, C. (2015a, February). Modality and Timing of Team Feedback: Implications for GIFT. In *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym2)* (p. 199).

Walton, J., Bonner, D., Walker, K., Mater, S., Dorneich, M., Gilbert, S., & West, R. (2015b, February). The Team Multiple Errands Test: A Platform to Evaluate Distributed Teams. In *Proceedings of the 18th ACM Conference Companion on Computer Supported Cooperative Work & Social Computing*(pp. 247-250). ACM.

ABOUT THE AUTHORS

Mr. Desmond Bonner *is a graduate student in the Human-Computer Interaction and Industrial Engineering program at Iowa State University's Virtual Reality Applications Center. His background is in Graphic Design. His work focuses on using games for learning to increase interest in Science, Technology, Engineering, and Mathematics (STEM) fields as well as leadership within small teams.*

Ms. Anna Slavina *is a graduate student in Psychology, Education, and Human-Computer Interaction at Iowa State University's Virtual Reality Applications Center. Her background is in psychology. Her research interests include the effects of technology on memory, intelligent tutoring systems, and visually induced motion sickness.*

Ms. Anastacia MacAllister *is a graduate student in Mechanical Engineering and Human-Computer Interaction at Iowa State University's Virtual Reality Applications Center. She is working on developing Augmented Reality work instructions for complex assembly and intelligent team tutoring systems.*

Mr. Joseph Holub, *is a graduate research assistant at the Virtual Reality Applications Center (VRAC) at Iowa State University where he is finishing his Ph.D. in human computer interaction (HCI) and computer engineering. He has worked on projects for path planning of unmanned aerial vehicles, live virtual constructive training of dismounted soldier, and visualization of large data using contextual self-organizing maps. Currently, he is working on building tools for augmented reality assembly in manufacturing as well as the GIFT team training architecture. His Ph.D. research is on visualizing functional imaging data on multiple hardware platforms.*

Dr. Stephen Gilbert *is an associate director of the Virtual Reality Applications Center and assistant professor of Industrial and Manufacturing Systems Engineering at Iowa State University. His research interests focus on technology to advance cognition, including interface design, intelligent tutoring systems, and cognitive engineering. He is a member of IEEE and ACM and works closely with industry and federal agencies on research contracts. He is currently PI on a project supporting the U.S. Army Research Laboratory STTC in future training technologies for teams.*

Dr. Anne M. Sinatra *is an Adaptive Tutoring Scientist at the Army Research Laboratory's SFC Smith Center. Her background is in Cognitive and Human Factors Psychology. She conducts adaptive training research as a member of the Learning in Intelligent Tutoring Environments (LITE) Lab and works on the Generalized Intelligent Framework for Tutoring (GIFT) project.*

Dr. Michael Dorneich *is associate professor of Industrial and Manufacturing Systems Engineering and a faculty affiliate of the human computer interaction (HCI) graduate program at Iowa State University. Dr. Dorneich's research interests focus on creating joint human-machine systems that enable people to be effective in the complex and often stressful environments found in aviation, robotic, learning, and space applications. Dr. Dorneich has over 19 years' experience developing adaptive systems which can provide assistance tailored to the user's current cognitive state, situation, and environment.*

Dr. Eliot Winer *is an associate director of the Virtual Reality Applications Center and associate professor of Mechanical Engineering and Electrical and Computer Engineering at Iowa State University. He is currently co-leading an effort to develop a next-generation mixed-reality virtual and constructive training environment for the U.S. Army. Dr. Winer has over 15 years of experience working in virtual reality and 3D computer graphics technologies on sponsored projects for the Department of Defense, Air Force Office of Scientific Research, Department of the Army, National Science Foundation, Department of Agriculture, Boeing, and John Deere.*

THEME II: GIFT FEATURE DEVELOPMENT

The GIFT 2016 Community Report

Scott Ososky[1, 2], Keith Brawner[1]
[1]U.S. Army Research Laboratory, [2]Oak Ridge Associated Universities (ORAU)

INTRODUCTION

The first version of the Generalized Intelligent Framework for Tutoring (GIFT) was released to the public in May 2012 (Sottilare, Brawner, Goldberg & Holden). One year later, the first symposium of the GIFT user community was held at the Artificial Intelligence and Education conference in Memphis, Tennessee. Since then, the GIFT development team has continued to gather feedback from the community regarding recommendations on how the GIFT project can continue to meet the needs of the user community and beyond. This paper continues the conversation with the GIFT user community in two important ways. First, it invites and encourages members of the GIFT user community to continue to share their feedback, research findings, and technology innovations with the development team and with one another in order to strengthen the power, usability, and flexibility of the GIFT project. This year, the title of the current paper has been updated to reflect the emphasis on the GIFT community. Second, as a follow up to the "GIFT 2015 Report Card and State of the Project" (Brawner & Ososky, 2015), this paper briefly describes how the GIFT development team is addressing features requested in previous GIFT Symposium meetings and serves as documentation for the next major project direction.

The research and technology innovation efforts presented in the current document include those that are informed by the GIFT user community, and only represent a *fraction* of the overall research, development, and implementation work associated with GIFT. We invite the reader to review the other chapters in this volume, publications on GIFTTutoring.org, and other references described below, to get a sense of the total body of work on the GIFT project. Major themes in this current, 2016 GIFT community discussion include user experience improvements for authors and researchers, functionality supporting experimentation with GIFT, and the alpha release of GIFT Cloud as an enabling technology.

WELCOME

First, to new GIFT users and new members to the GIFT community, welcome. There are a number of recommended resources that will help orient new members to the GIFT project beginning with the original GIFT description paper (Sottilare, Brawner, Goldberg, & Holden, 2012) and the new GIFT Quick Start Guide (Ososky, 2016). The GIFT user community is also invited to ask questions and share experiences and feedback on our forums (https://gifttutoring.org/projects/gift/boards). The forums are actively monitored by a small team of developers, in addition to a series of Government project managers. The forums are a reliable way to interact with the development team and other members of the GIFT community. The forums, at the time of this writing, have over 600 postings and responses.

This past year, a series of research outlines were published that describe the larger scope of the adaptive training research initiatives on *domain modeling* (Sottilare, Sinatra, Boyce, & Graesser, 2015), *instructional management* (Goldberg, Sinatra, Sottilare, Moss, & Graesser, 2015), *authoring tools* (Ososky, Sottilare, Brawner, Long, & Graesser, 2015), *learner modeling* (Goodwin, Johnston, et al., 2015), and *training effectiveness* (Johnston et al., 2015), plus an upcoming outline on *architectural research*. Additionally, GIFT community developers and power-users are encouraged to read the documentation provided with each downloadable release of GIFT (located within the GIFT install folder at GIFT\docs\index.html). That documentation will also be available online, at GIFTTutoring.org, in the near future.

COMMUNITY-REQUESTED FEATURES AND ARCHITECTURE

In combination with suggestions gathered from the forums, there have been many suggested improvements in the previous meetings of the GIFT community (Sottilare, 2014; Sottilare & Holden, 2013; Sottilare & Sinatra, 2015). Recommendations have been documented in the associated proceedings, while being actively addressed by the development groups. **Table 3** summarizes the features and functionality requested during the previous years' GIFT Symposiums and Design Recommendation on Intelligent Tutoring Systems book series as completely as practical, with a summary of the degree to which each request has been, or will be addressed. **Table 3** is generally divided in accordance with the features which are 1) *Existing*: currently available at time of writing this paper from the GIFT website, 2) *Emerging:* have had significant work applied, are possibly available upon request, and are scheduled to be widely available within the next two releases, or 3) *Future:* is in planning to be available in the long-term. Additionally, items characterized as existing does not indicate that development in those areas has stopped, please see individual sections for status of work on each of the efforts.

Table 3. List of requested functionality, source of requests, and status of responses

Section	Functionality	Existing	Emerging	Future
0	Make further use of GAMETE, incorporate dialogue, incorporate dialogue-mining	X		
0	Metacognitive framework, strategy identification	X		
0	Web-based, service-based	X		
0	Easier access to GIFT on the web	X		
0	Ability to rapidly create and run experiments	X		
0	Learnable, efficient, authoring UX though authoring tool enhancements		X	
0	Conduct effectiveness evaluation on authoring techniques		X	
0	Tools for course content creation with external applications		X	
0	Unity platform and Physics Playground integration		X	
0	Sensor-based domain assessments		X	
0	Team tutoring architecture		X	
0	Modular reinforcement learning, data-driven processes for mining		X	
0	Fine-grained experience application programming interface (xAPI) tracking, better-informed learning record store (LRS), analytics for prediction		X	
0	Transition prediction of states			X
0	Use the xAPI, expand its use in instructional tactics, integrate it with other systems			X

EXISTING FEATURES AND TECHNOLOGIES

Make further use of GAMETE, incorporate dialogue, incorporate dialogue-mining

In the previous year's community report, it was planned to make further use of the Game-based Architecture for Mentor-Enhanced Training Environments (GAMETE) project for rapidly connecting tutoring systems with simulated training environments. This functionality was requested in a number of works (Cai, Graesser, Hu & Nye, 2015; Engimann et al., 2014; Rus, Maharjan & Banjade, 2015). GAMETE, as a project, has ended, but the intellectual property of the project has been transferred free of cost to the Government, and, by extension, the GIFT project. At the end of the GAMETE, the platform was in a noticeable number of separate components, interfaces, specifications, and functionality. Much of this functionality is back-end functionality which has been absorbed into GIFT in a number of unspecified ways. However, the most noticeable, front-end, components provide easy way to author finite conversations with the tutor (**Figure 15**). This includes a student interface for having the conversations, an authoring interface to create responses for the student, and a basic manner of authoring assessment. This interface has been standardized, incorporated into GIFT as a new authoring tool, and enables the creation of student- or tutor- driven dialogue based on actions and assessments. This new tool addresses other projects that were seeking to incorporate student dialogues into the tutoring process.

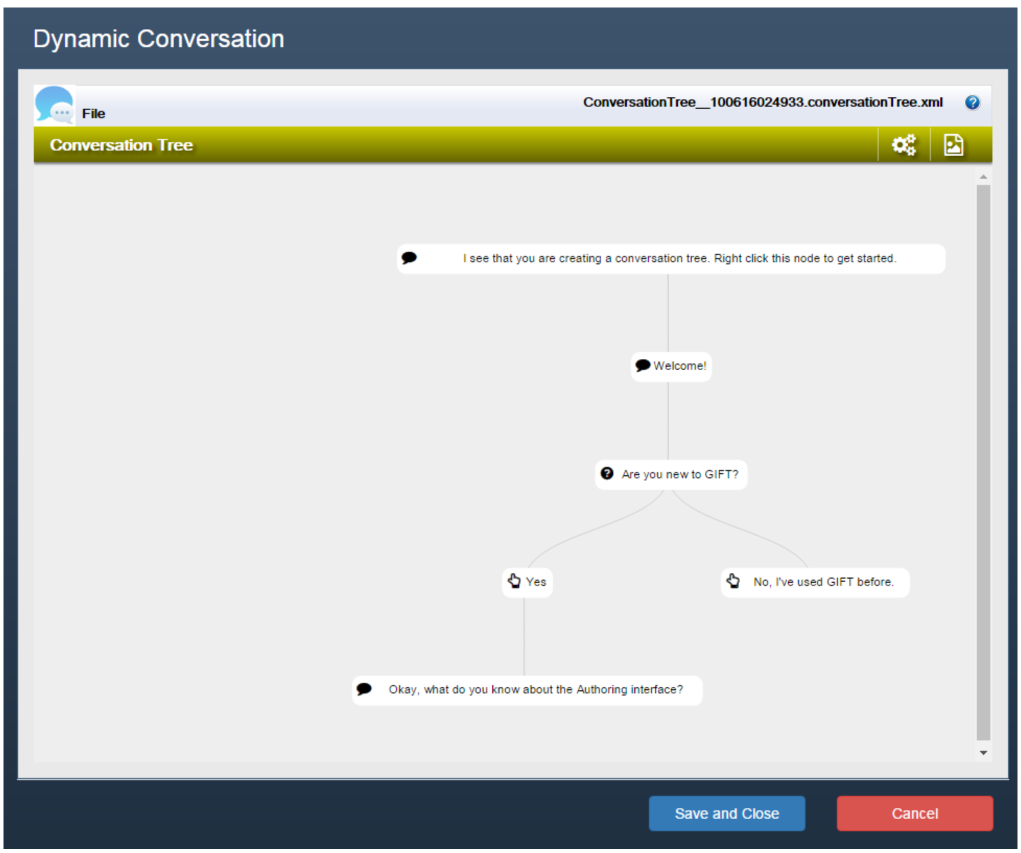


Figure 15 - Authoring interface for conversation trees. Conversation Trees are now a type of Survey in the Authoring system.

Metacognitive framework, strategy identification

There is currently no explicit representation for "metacognitive framework". However, many conversations have occurred to try to determine a way to measure, record, store, and reason with metacognitive and process data in order to address the requirements for instructional framework construction (Segedy, Kinnebrew, Goldberg, Sottilare & Biswas, 2015). It was determined that an expansion to the existing framework was not required, but that a new data attribute value could be added to the learner state in order to represent knowledge of metacognition. That value is updated each turn (in a turn-based simulator, clock-cycle in a real-time simulator), based on the actions that the student took, and is automatically stored via GIFT log message and xAPI long-term profile, respectively. That value is able to be reasoned with through the Domain, Learner, and Pedagogical modules, which pass information and updates to the value. The near-term goal is to be able to relate metacognitive experiences across domains of instruction, eventually being able to broadly relate domains of instruction (e.g., tutor feedback, "read before acting, just as you did in [Domain 1]").

Web-based, service-based architecture

There has long been a community need and request for making GIFT usable over the open internet. A major task was undertaken to apply all of the GIFT interoperable technology to be usable over the Hypertext Transfer Protocol (HTTP) protocol. This includes interfacing with simulation technology that resides on the desktop computer through the use of downloadable *Gateway* interop Java applets. These applets download automatically, contain digitally signed certificates, and are procedurally generated based on the external applications available in a specifically authored course. The use of web technology to interface with a PC, and potentially complex, simulations and simulators is a powerful feature which is automatically configured for authors without further modification, as per previous research requests (Brawner & Ososky, 2015; Nye & Morrison, 2013; Rus et al., 2013).

GIFT Cloud is the web-based version of GIFT, enabled by that development effort. Now that GIFT operates on the web through web protocol standards, there is an opportunity for modules to be separate from the functional portion of the network where they currently operate. In short, modules are free to operate as separable services, present on the web, for any who choose use them. Further, efforts in an agent-based architecture are tasked with expanding the availability of the service-based functionality behind these modules, and expanding it to simplify the addition of further services. Each additional release of GIFT has increased reliance on browser- and web-based tools in response to community demand.

Easier access to GIFT on the web

GIFT Cloud allows users to create, manage, and access course content within an application, described in the previous section. Unlike previous versions of GIFT, GIFT Cloud (located at cloud.gifttutoring.org) does not require the user to download or install any software on their computer. Authors have access to a private workspace within the application, where their course files and other media are stored. Thus, it is possible to start creating a course on one computer and resume work on the course later from a different computer. GIFT Cloud is currently in *alpha*, with additional enhancements planned in the coming months.

User-created courses are secured via GIFT Account, which can be created on GIFT Tutoring at no cost. The GIFT Account provides access to restricted sections of GIFT Tutoring (including downloadable versions of GIFT) in addition to the web-based GIFT Cloud. The GIFT Account also allows users to take courses and view course history within GIFT Cloud. More information is available on GIFT Cloud within the companion Quick Start Guide (Ososky, 2016), as well as in a usability-focused overview document (Ososky, Brawner, Goldberg & Sottilare, 2016). This important and significant release provides the foundation for further UX improvements, such as specialized tools for experimentation, and rapid prototyping of more intuitive course-authoring interfaces.

Ability to rapidly create and run experiments

Research helps to power innovation on the GIFT platform. The release of GIFT Cloud also introduced a specialized interface specifically for users who want to conduct research using GIFT, in response to the request for rapid ability to run experiments (Sinatra, 2015). The new My Experiments section within GIFT Cloud currently provides the following functions: The user interface (UI) creates research versions of existing courses, which are locked for editing to ensure experimental control. In order to protect the confidentiality of study participants, the UI creates a unique study link that does not require the use of a GIFT Account to access the course. The study link can also be activated or de-activated by the researcher in order to comply with a data collection schedule. Finally, the My Experiments interface provides the tools to generate customized data output files to download to a local computer for further analysis. The development team plans to continue to expand the native functionality of the My Experiments tools to accommodate different research design and compliance requirements.

EMERGING FEATURES AND TECHNOLOGIES

Learnable, efficient, authoring UX though authoring tool enhancements

GIFT Cloud allows the GIFT development team to rapidly update the platform in response to feedback and end-user goals. The team is already developing new experiences for the authoring tools that will help to make course creation more intuitive and efficient. These experiences are intended to align more closely with the end-user's mental model of course authoring, rather than based on a representation of the system architecture, as described in Ososky et al. (2015). The GIFT Authoring Tool (GAT) updates that will be coming later this year will focus on three areas of improvement (Ososky, 2016, May). First, a new flowchart-style visual course authoring interface will provide a more natural way for authors to sequence course objects and visualize the structure of their tutors. The second area of focus is consistency within interfaces and terminology, respectively: A new survey authoring UI is being designed to be more tightly integrated with the GAT UI, making it easier and more efficient to access and create survey content from within the GAT. System-level terminology will also be updated with user-friendly language, to help make GIFT more learnable for novice and intermediate GIFT authors. Finally, a comprehensive redesign of GIFT's help and support framework is underway that will provide more specific and on-demand help to authors within the GAT, at the point of need. These improvements will be made available with the release of GIFT 2016-1, with future work anticipated to build upon them.

Conduct effectiveness evaluation on authoring techniques

Given the web-based GIFT Cloud, we also gain the ability to automatically record the actions that authors take in order to make adaptive courses, as requested in previous papers (Brawner & Ososky, 2015; MacLellan, Wiese, Matsuda & Koedinger, 2014). Anonymized data can be collected to determine the adaptive tutoring features/functions that are used to create courses. The GIFT team can leverage these data to continue to improve the usability of the authoring tools. Additionally, the data regarding the effectiveness of the course to teach its desired subject can be made available (e.g., effectiveness of content, performance, or adaptation). While the data are not yet collected in a systemic fashion, usage data are planned to be collected in the future, via embedded analytics tools. Note that we do not, nor plan to, look at actual data collected from students (e.g., demographic, performance) without explicit permission.

Tools for course content creation with external applications

One of the most advanced functions within GIFT is dynamic real-time assessment coupled with an external application, typically for training purposes. To that end, the GIFT team is developing tools that help authors to more rapidly create GIFT content using their external applications. Currently, authoring dynamic real-time assessment content takes place exclusively within the GIFT authoring tools (GAT), and that interface does not have any direct communication with the external application. For instance, the user must manually input the names and/or locations of objects associated within their training applications into GIFT in order for GIFT to track these elements during course runtime. An initial version of a new tool, “GIFT Wrap,” is being developed, which will help to connect GIFT with external applications in order to make authoring assessments with application scenarios more efficient. GIFT Wrap is scheduled to be released with GIFT version 2016-1, with future work anticipated to improve its applicability and functionality.

Unity platform and Physics Playground integration

The Unity platform is one of the external applications that are of interest to the GIFT user community. A version of GIFT that incorporates the Unity environment was previously requested (Brawner & Ososky, 2015; Ray & Gilbert, 2013). A demonstration of a Unity-based environment was presented at the 3rd GIFT Users Symposium (Zhao, Ventura, Nye & Hu, 2015), and development effort to support Unity in GIFT has continued. At the time of this writing, it is anticipated that GIFT-2016-2X will support the ability to incorporate Unity-based environments in GIFT courses in both the GIFT Local (desktop) and GIFT Cloud instances, and include an example Physics Playground course, demonstrating that functionality. Interested parties should be aware of the efforts of the University of Memphis and Aptima teams who are working with Unity; a version of the software that incorporates Unity functionality is currently available upon request.

Sensor-based domain assessments

One of the suggestions from the previous GIFT Symposiums is the ability to use sensor to assess, not only learner state, but also assess domain performance (Goldberg & Amburn, 2015). This is especially relevant to a domain where physical performance is the critical learned skill, such as in psychomotor training domains. This capability has been developed and is currently in use in data collection efforts. The software changes required to support sensor-based domain assessments are available upon request. They are expected to be available in future releases after the experimentation and validation efforts are completed.

Team tutoring architecture

Burke and Gilbert have each stressed that team tutoring is important for the long-term training of military persons (Burke, Feitosa & Salas, 2015; Gilbert et al., 2015). Military people train in small groups significantly more often than they train individually, and team training is correspondingly more important in this context. In order to effectively train teams, the team state, team pedagogy, and team domain information must all be represented in the same way as for an individual. As a concrete example, a team of two observers communicating with each other may be provided a performance assessment based on the effectiveness of their communication, a learner state based on their communication frequency, and have assigned pedagogy reflecting encouragement for clear communication according to a common communication standard. This type of problem is currently addressed in a development branch of GIFT, which makes use of team modules for each of the various GIFT modules. This branch is available upon request, but has not yet been empirically validated. Experiments with those team modules are currently in planning.

Modular reinforcement learning, data-driven processes for mining

Other papers in this symposium proceeding are addressing the needs of data-driven process for mining (Rowe et al., 2015). Examples of these programs are in the After Action Review (AAR) efforts from Aptima, the Markov Decision Process framework efforts from North Carolina State University, and others. The general methods for creating policies and data mining are currently available upon request and are expected to be incorporated in the late 2016 or early 2017 GIFT release. More on these subjects can be read about in the respective papers, published in the same proceedings as this work.

Fine-grained xAPI tracking, better-informed LRS, analytics for prediction

The use of finer-grained xAPI tracking is in development, consistent with recommendations (Goodwin, Murphy & Hruska, 2015). There is currently an effort to incorporate web-based training experiences, simulated experiences, live experiences, and actual performance on task in order to track the impact that differing instructional interventions make on the long term. Increased xAPI tracking ability is needed in order to make this distinction and is being integrated into GIFT. The second part of this effort, the ability to make sense of this type of data, is also being added in the form of an online analytics platform. Generally speaking, these efforts require significantly more resources than other portions of the current task list (Table 1) and are taking correspondingly longer to implement. Further work is being performed in this area through potential integration with the Advanced Distributed Learning (ADL) Total Learning Architecture (TLA).

FUTURE FEATURES AND TECHNOLOGIES

Transition prediction of states

Last year's version of this community report mentioned that it was planned to make use of the "predicted" value of learner states, as part of feature requests from Rowe and Defalco (Brawner & Ososky, 2015; DeFalco & Baker, 2013; Rowe, Lobene & Sabourin, 2013). This is still currently planned with multiple projects beginning to make use of it. The recent completion of the final study using GIFT with Teacher's College has made use of sensor-based current-state affect detection and game-based current-state affect detection. Those affective state detections were used to inform instruction, delivering only instruction, which was shown to be valuable, having shown value from a previous study. The prediction value for "next state" is still anticipated to be used and will be addressed in the Vanderbilt metacognitive effort, the North Carolina State University tutorial planning effort, or the University of Southern California agent-driven framework effort. The exact effort that addresses this first is left to the individual efforts to decide and the reader can keep abreast the research through following the associated research groups.

Use the xAPI, expand its use in instructional tactics, integrate it with other systems

xAPI ("The xAPI Overview," 2016) is being used more extensively in GIFT than previously, as requested in last year's work (Brawner & Ososky, 2015; Poeppelman, Hruska, Long & Amburn, 2014). GIFT still provides the ability to log final scoring into a LRS, and is compatible with most, if not all, major LRS vendors. Near-term plans include the integration with LearnSphere, and an effort to expand DataShop (Koedinger et al., 2010) to encompass an increased variety of tutoring data. The combination of these two efforts generally promotes the flexibility of the system. Additionally, future xAPI use is intended for cross-environment tracking, and through potential integration with the ADL TLA.

CONCLUSION

Community commitment and future research directions

The GIFT team understands that there are potentially many solutions available to those in need of instructional content, and that the cost associated with switching to and learning a new platform are not trivial (Ososky, 2016). The GIFT development team remains committed to proactively improving the usability and flexibility of GIFT for its broad community of users. Features and improvements described earlier in this document are intended to cultivate a positive user experience for the GIFT community of tutor authors and intelligent tutoring researchers. One of the major themes of the past year has been GIFT usability, as a function of learnability and efficiency. As with previous improvements to GIFT's content creation tools, the current planned enhancements create a roadmap for future work related to designing the UX of GIFT.

Future research should build upon the GIFT Wrap project to support greater number of external applications as an efficient and flexible authoring solution. Future work on the researcher tools should include the ability to export study data in different formats, and provide additional options for balancing and/or randomizing experimental condition assignment. Continued work on the GAT should be focused on optimizing standardizing authoring interfaces for each course object type, plus exploring methods to best organize collections of courses and then manage those projects in a collaborative authoring context. Finally, GIFT support should be expanded to include enhanced content in the form of authoring templates and tutorial videos. The move toward a web-based version of GIFT as GIFT Cloud represented a significant step in being able to more rapidly test and implement these features for use by the community.

Shape the future of GIFT

GIFT is intended to provide members of the training, educational, and research communities with the tools and technology needed to efficiently create, manage, and deliver adaptive tutoring content, through leveraging a flexible and extendable framework. GIFT will be continuously improved and developed for the foreseeable future. The members of the GIFT community have a valuable opportunity to help shape how these and other features are designed and implemented into GIFT. The GIFT development team encourages members of the GIFT community to continue to communicate feedback, issues, suggestions, and results (of research) in order to help us provide the useful tools, powerful technologies, and positive user experiences that will make adaptive tutoring technology accessible to the broadest possible audience.

REFERENCES

Brawner, K. & Ososky, S. (2015). The GIFT 2015 report card and the state of the project. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 137.

Burke, C. S., Feitosa, J. & Salas, E. (2015). The Unpacking of Team Models in GIFT. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 49.

Cai, Z., Graesser, A. C., Hu, X. & Nye, B. D. (2015). CSAL AutoTutor: Integrating Rich Media with AutoTutor. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 169.

DeFalco, J. A. & Baker, R. S. (2013). Detection and Transition Analysis of Engagement and Affect in a Simulation-based Combat Medic Training Environment. *AIED 2013 Workshops Proceedings Volume 7*, 88.

Engimann, J., Santarelli, T., Zachary, W., Hu, X., Cai, Z., Mall, H. & Goldberg, B. (2014). *Game-based Architecture for Mentor-Enhanced Training Environments (GAMETE)*. GIFTSym2, Pittsburgh, PA.

Gilbert, S., Winer, E., Holub, J., Richardson, T., Dorneich, M. & Hoffman, M. (2015). Characteristics of a Multi-User Tutoring Architecture. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 39.

Goldberg, B. & Amburn, C. (2015). The Application of GIFT in a Psychomotor Domain of Instruction: A Marksmanship Use Case. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 115.

Goldberg, B., Sinatra, A., Sottilare, R., Moss, J. & Graesser, A. (2015). Instructional Management for Adaptive Training and Education in Support of the US Army Learning Model-Research Outline: DTIC Document.

Goodwin, G., Johnston, J., Sottilare, R., Brawner, K., Sinatra, A. & Graesser, A. (2015). Individual Learner and Team Modeling for Adaptive Training and Education in Support of the US Army Learning Model-Research Outline: DTIC Document.

Goodwin, G., Murphy, J. S. & Hruska, M. (2015). Developing Persistent, Interoperable Learner Models in GIFT. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 27.

Johnston, J., Goodwin, G., Moss, J., Sottilare, R., Ososky, S., Cruz, D. & Graesser, A. (2015). Effectiveness Evaluation Tools and Methods for Adaptive Training and Education in Support of the US Army Learning Model-Research Outline: DTIC Document.

Koedinger, K. R., Baker, R., Cunningham, K., Skogsholm, A., Leber, B. & Stamper, J. (2010). A data repository for the EDM community: The PSLC DataShop. *Handbook of Educational Data Mining*, 43-55.

MacLellan, C. J., Wiese, E. S., Matsuda, N. & Koedinger, K. R. (2014). *SimStudent: Authoring Expert Models by Tutoring*. GIFTSym2, Pittsburgh, PA.

Nye, B. D. & Morrison, D. (2013). Toward a Generalized Framework for Intelligent Teaching and Learning Systems. *Artificial Intelligence in Education (AIED)*.

Ososky, S. (2016). *Generalized Intelligent Framework for Tutoring (GIFT) Cloud / Virtual Open Campus quick start guide*. (ARL-CR-0796). Orlando, FL: US Army Research Laboratory.

Ososky, S. (2016, May). Designing the user experience of the GIFT Cloud authoring tools. *Proceedings of the 4th Annual GIFT Users Symposium*.

Ososky, S. (2016). Practical requirements for ITS authoring tools from a user experience perspective. In D. D. Schmorrow & C. M. Fidopiastis (Eds.), *Foundations of Augmented Cognition*: Manuscript submitted for publication.

Ososky, S., Brawner, K., Goldberg, B., & Sottilare, R. A. (2016). GIFT Cloud: Improving usability of adaptive tutor authoring tools within a web-based application. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*.

Ososky, S., Sottilare, R., Brawner, K., Long, R., & Graesser, A. (2015). *Authoring Tools and Methods for Adaptive Training and Education in Support of the Army Learning Model: Research Outline*. (ARL-SR-0339). Orlando, FL: U.S. Army Research Laboratory.

Poeppelman, T., Hruska, M., Long, R., & Amburn, C. (2014). *Interoperable Performance Assessment for Individuals and Teams Using Experience API*. GIFTSym2, Pittsburgh, PA.

Ray, C., & Gilbert, S. (2013). Bringing Authoring Tools for Intelligent Tutoring Systems and Serious Games Closer Together: Integrating GIFT with the Unity Game Engine. *AIED 2013 Workshops Proceedings Volume 7*, 27.

Rowe, J., Lobene, E. V., & Sabourin, J. (2013). Run-Time Affect Modeling in a Serious Game with the Generalized Intelligent Framework for Tutoring. *AIED 2013 Workshops Proceedings Volume 7*, 95.

Rowe, J., Mott, B., Lester, J., Pokorny, B., Peng, W., & Goldberg, B. (2015). Toward a Modular Reinforcement Learning Framework for Tutorial Planning in GIFT. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 89.

Rus, V., Maharjan, N., & Banjade, R. (2015). Unsupervised Discovery of Tutorial Dialogue Modes in Human-to-Human Tutorial Data. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 63.

Rus, V., Niraula, N., Lintean, M., Banjade, R., Stefanescu, D., & Baggett, W. (2013). Recommendations For The Generalized Intelligent Framework for Tutoring Based On The Development Of The DeepTutor Tutoring Service. *Artificial Intelligence in Education (AIED)*.

Segedy, J. R., Kinnebrew, J. S., Goldberg, B. S., Sottilare, R., & Biswas, G. (2015). Using GIFT to Model and Support Students' Metacognition in the UrbanSim Open-Ended Learning Environment. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 13.

Sinatra, A. M. (2015). The Instructor's Guide to GIFT: Recommendations for using GIFT In and Out of the Classroom. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 149.

Sottilare, R. (2014). *Proceedings of the 2nd GIFT Users Symposium*. Pittsburgh, PA: Army Research Laboratory.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Sottilare, R., & Holden, H. K. (2013). Recommendations for Authoring, Instructional Strategies and Analysis for Intelligent Tutoring Systems (ITS): Towards the Development of a Generalized Intelligent Framework for Tutoring (GIFT). *Artificial Intelligence in Education 2013 1*.

Sottilare, R., Sinatra, A., Boyce, M., & Graesser, A. (2015). Domain Modeling for Adaptive Training and Education in Support of the US Army Learning Model-Research Outline: DTIC Document.

Sottilare, R., & Sinatra, A. M. (2015). *Proceedings of the 3rd GIFT Users Symposium*. Orlando, FL: Army Research Laboratory.

The xAPI Overview. (2016). Retrieved from https://www.adlnet.gov/adl-research/performance-tracking-analysis/experience-api/

Zhao, W., Ventura, M., Nye, B. D., & Hu, X. (2015). GIFT-Powered NewtonianTalk. *Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3)*, 3.

ABOUT THE AUTHORS

Dr. Keith Brawner *is a researcher for the U. S. Army Research Laboratory's Human Research & Engineering Directorate (ARL-HRED), and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT). He has 10 years of experience within U.S. Army and Navy acquisition, development, and research agencies. He holds a Masters and PhD degree in Computer Engineering with a focus on Intelligent Systems and Machine Learning from the University of Central Florida. His current research is in ITS architectures and cognitive architectures. He manages research in adaptive training, semi/fully automated user tools for adaptive training content, and architectural programs towards next-generation training.*

Dr. Scott Ososky *is a Postdoctoral Research Scientist within the US Army Research Laboratory's Human Research and Engineering Directorate (ARL-HRED). His current research examines mental models of adaptive tutor authoring, including user experience issues related to development tools and interfaces within the adaptive tutor authoring workflow. He has also published numerous works regarding human interaction with near-futuristic, intelligent robotic teammates. Dr. Ososky received his Ph.D. and M.S. in Modeling & Simulation, as well as a B.S. in Management Information Systems, from the University of Central Florida.*

Using GIFT Wrap to Author Domain Assessment Models with Native Training Applications

Michael Hoffman[1], Chris Markuck[1], Benjamin Goldberg[2]
[1]Dignitas Technologies, [2]U.S. Army Research Lab

INTRODUCTION

The Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Goldberg, Brawner & Holden, 2012) is a framework and tool set for the creation of intelligent and adaptive tutoring systems (Brawner, 2012; Sottilare, Goldberg, Brawner & Holden, 2012). Since its inception, GIFT has become a standard for authoring, deploying, managing, and evaluating Intelligent tutoring system (ITS) technologies. In that time GIFT has pursued best practices for automated instruction, course authoring and sound instructional strategies, as well as to facilitate ongoing ITS research. With GIFT, users can create tutors with domain agnostic tools that vary from simple content delivery to adaptive and individualized learner experiences. To date, GIFT has already been used across various domains from existing external simulations, serious games, and computer-based training environments to teach physics, train military tasks and tactics, and solve cognitive problems.

One of the goals of GIFT is to create tutors that can be used across domains and training applications. Single domain tutors can often excel in providing a rich authoring experience and learning environment by limiting the choices available. Understandably, it is difficult to design an intuitive authoring tool that is flexible enough to support almost any domain. Part of the generalizability of GIFT is the ability to interact with and assess the learner's skill in an, often times existing, external training application (e.g. Bohemia Interactive's Virtual Battlespace, Newton's playground). This presents an additional authoring challenge. In eLearning and computer based training systems (CBTS), the author is concerned with content delivery and presentation, therefore the authoring tools are tailored around that learning experience. While in a single domain tutoring system, each training domain uses a different training applications and therefore the authoring tools favor a single domain. GIFT must provide the capability of both content delivery as well as assessing learner actions against a real-time domain assessment model in any training environment. It is this requirement and the unknown of what future domains and training systems will be integrated that necessitates the need for authoring tools that are generalized and easily extended.

A generalized tool to build tutors helps to facilitate authoring across domains, however it decreases the likelihood that the authoring experience will be well guided with context sensitive help when necessary. For example a course designer creating a course on counter insurgency and another creating a course on crowd control would benefit from course creation interfaces tailored for each domain to help guide them through the authoring process. However, since it is currently inconceivable to develop an authoring environment for all domains and user roles, an alternative solution is needed that can provide the domain familiarity while still offering a generalized approach. This paper provides an insight into ongoing development on the next generation GIFT authoring tool called GIFT Wrap. GIFT Wrap aims to merge the abstract nature of authoring domain assessment models with the native training applications that subject-matter experts are familiar with. This connection will diminish the learning curve associated with authoring for an ITS.

COURSE AUTHORING

The GIFT tool suite includes support for authoring of course flow, domain assessment models, pedagogy, sensor configurations, surveys, and the use of third-party tools such as the AutoTutor Script Authoring Tool and Student Information Models for Intelligent Learning Environments (SIMILE) Workbench. The core GIFT authoring tools create a series of extensible markup language (XML) files. Those XML files are used during course execution to populate and configure the various GIFT modules, which, in turn, enables logic that manages and assess the learner's progress. By defining course configurations in this manner, GIFT supports authoring courses using tools both developed by the core GIFT team as well as by other third parties. This encourages a solution where authors can not only use familiar interfaces and mental models but also take advantage of tools that are tailored for certain domains and applications. In the ideal case, GIFT users would enhance authoring functions by extending the current baseline tool to meet their specific need. Historically, however, users have shown an overwhelming interest in using the existing authoring tools rather than building on top of them. Therefore, to support course creation and development requirements, a series of authoring tools were developed over GIFT's lifetime. Each generation improved upon the existing authoring experience in some manner. While this paper does provide an introduction to some of the GIFT authoring tools, a more detailed explanation of each tool can be found in Hoffman & Ragusa (2014). Before describing the development of GIFT Wrap it is helpful to understand the previous authoring tools in order to elicit appreciation for GIFT Wrap by exposing benefits and short-comings of earlier efforts.

Desktop XML Editor

The first GIFT authoring tools were developed simply because the XML file structure was becoming too complex to author in a text editor. At the time, we were developing several key pieces of the architecture in GIFT that not only lowered the priority of developing a user-friendly authoring tool but would have made it difficult to define the requirements for such a user interface. Therefore, we needed a tool set that provided a graphical user interface for authoring that also included improved element validation, tips, and hints for complex elements, and most importantly, an interface that would require minimal software development as the schemas were continually being improved upon.

After some consideration we integrated the Java library called XAmple XML Editor. XAmple provided the necessary features of dynamically adapting a graphical authoring user interface based on schema changes. In addition, the tool offered the ability to extend the interface with custom selection logic for any XML element, which is useful, for example, when trying to select a survey object that was authored in the separate survey system. Within just two weeks, we built several authoring tools for GIFT. Each tool gave the author the ability to create one of the GIFT XML configuration files required for an executable lesson. One example of a GIFT XML authoring tool is the domain knowledge file (DKF) authoring tool (DAT), which can be seen in Figure 16. The DAT allows DKFs to be created. DKFs contain the domain assessment model and instructional strategy information for a scenario within a GIFT course. In this figure, a portion of the XML tree structure driven by the underlying DKF XML schema is expanded to show the authoring experience when trying to edit an element that is a distant descendant in the inherent XML hierarchy.

These XML authoring tools were very helpful at the time and numerous courses and experiments were created using these interfaces. In addition they provided the agility, flexibility and engineering usefulness software developers appreciate. However they lacked the usability features that would satisfy the needs of the entire GIFT community from subject matter experts to experts in course design and software developer to instructional system designer. Moreover, GIFT was moving towards web-based user interfaces and services in order to make the framework more accessible, which meant that this suite of desktop based tools required a web interface counterpart.

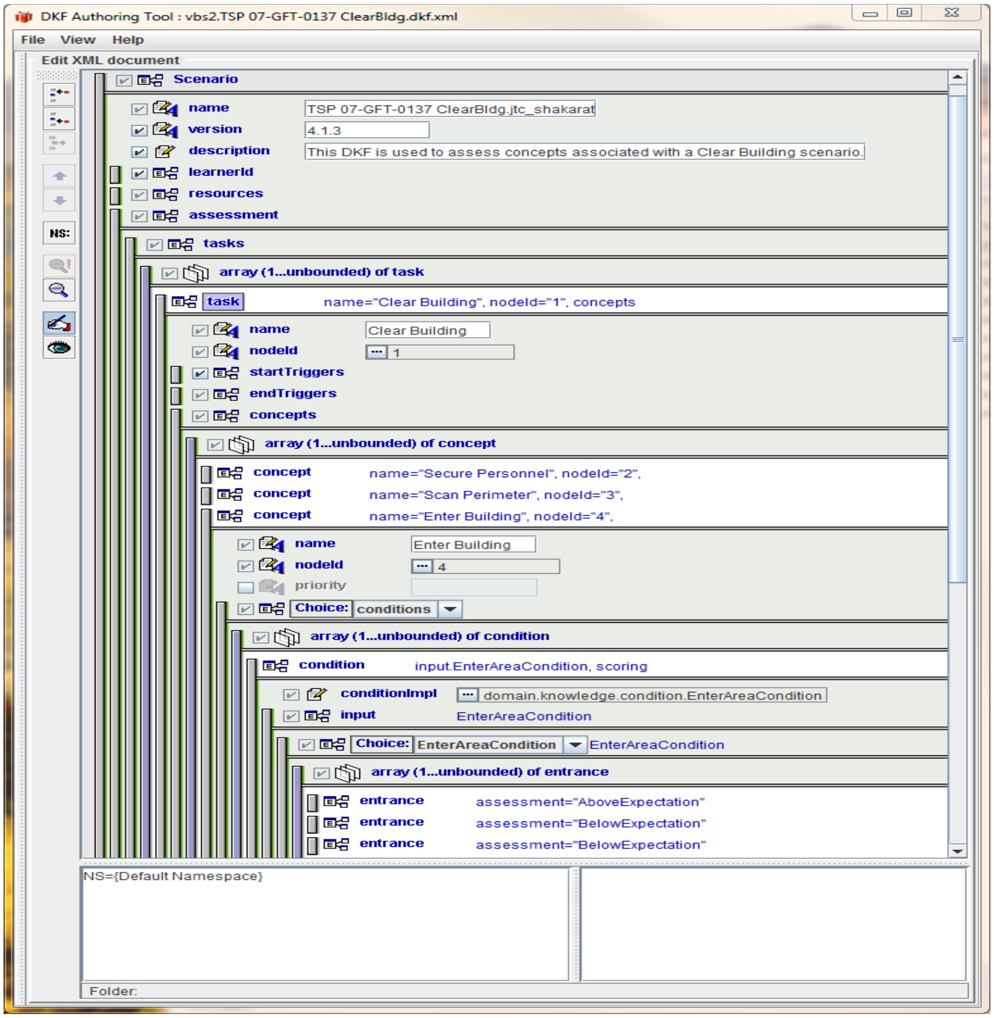


Figure 16. The Domain Knowledge File (DKF) desktop authoring tool showing the Clear Building DKF contents. The XML tree is expanded to show the complicated nesting of information that is likely to happen.

1st Generation Web Course Authoring

Having gathered lessons learned from developing and using the XML desktop-based authoring tools along with input from the GIFT community, the first web-based authoring tool was created called the GIFT Authoring Tool (GAT). The initial version of the GAT satisfied several usability requirements (but not all) and was substantially easier to use than the desktop XML authoring tools. As one can see in Figure 17, the interface greatly improved the presentation of the underlying data model by hiding the XML structure from the author that was shown in Figure 16. It also attempted to present related elements within the same view.

Though an improvement, this suite of tools fell short of the long-term goals. The interface didn't provide context sensitive help, adapt to user roles, or hide unnecessary complexity. There was no attempt at translating the XML named elements to author from software engineering terms into course designer jargon which would have lowered the expertise required to author a GIFT course. Furthermore, the authoring tool didn't address user authentication or support for multiple concurrent users.

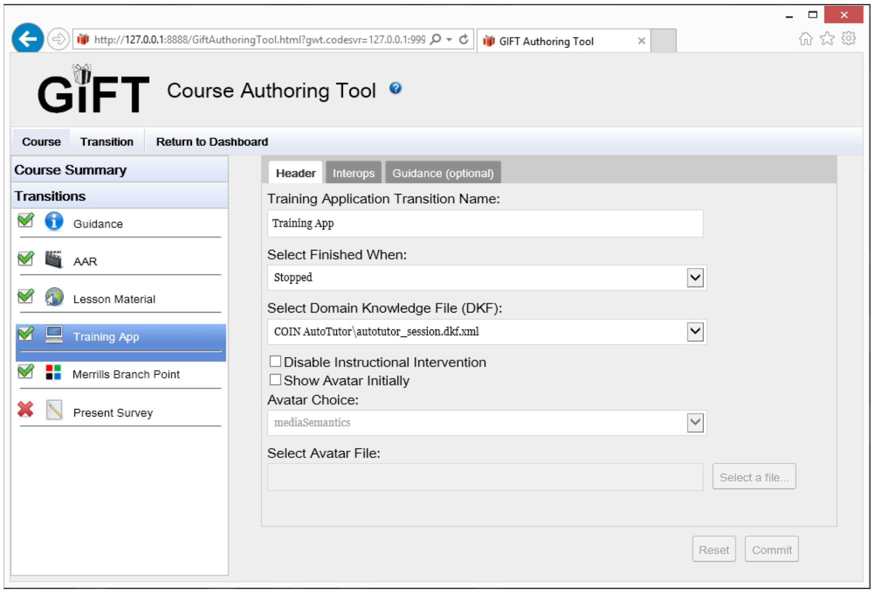


Figure 17. The web-based course authoring tool showing an instance of a Training Application course transition. (Notice how this interface is more user-friendly and graphically oriented versus the XML-based authoring tools used by GIFT developers.)

2nd Generation Web Course Authoring

The requirements for this generation of the GAT were driven by the need to host a GIFT instance on a server while allowing a user to fully author a GIFT course. With this version, the GIFT community no longer would have to download and install GIFT on their personal computers. They could experience what GIFT had to offer by merely visiting a website and exploring the new GIFT dashboard. This version, depicted in **Figure 18**, improved upon the following features:

- Ability to author sensor/learner/pedagogical configurations for a course in a web-based interface
- Organized objects relevant to a course into a course folder
- Allowed users to manage workspaces in GIFT rather than in Windows File Explorer
- User authentication
- Supported multiple concurrent authors on a per file basis
- Read only courses and surveys
- Replaced engineering terms in the authoring user interface with laymen's terms and phrases
- Provide context sensitive help and input validation at various levels
- Created several wizards to help guide the authoring process
- Improved usability of authoring in GIFT
- Web-based course import and export capabilities
- Better issue reporting to the user in the web environment versus the refer users to the GIFT log files
- Deliver courses as part of a user study / experiment

Even though these improvements significantly enhanced authoring in GIFT, the implementation still required an abundant understanding of GIFT before creating even the simplest of courses. For example, most GIFT users don't need to know what a DKF is but rather understand that GIFT provides the ability to author assessment models and those models are used to assess the learner. Furthermore a GIFT user doesn't need to realize that a GIFT course requires a survey context to be defined that encapsulates the surveys that can be presented in that course and instead that GIFT allows surveys to be managed in a course.

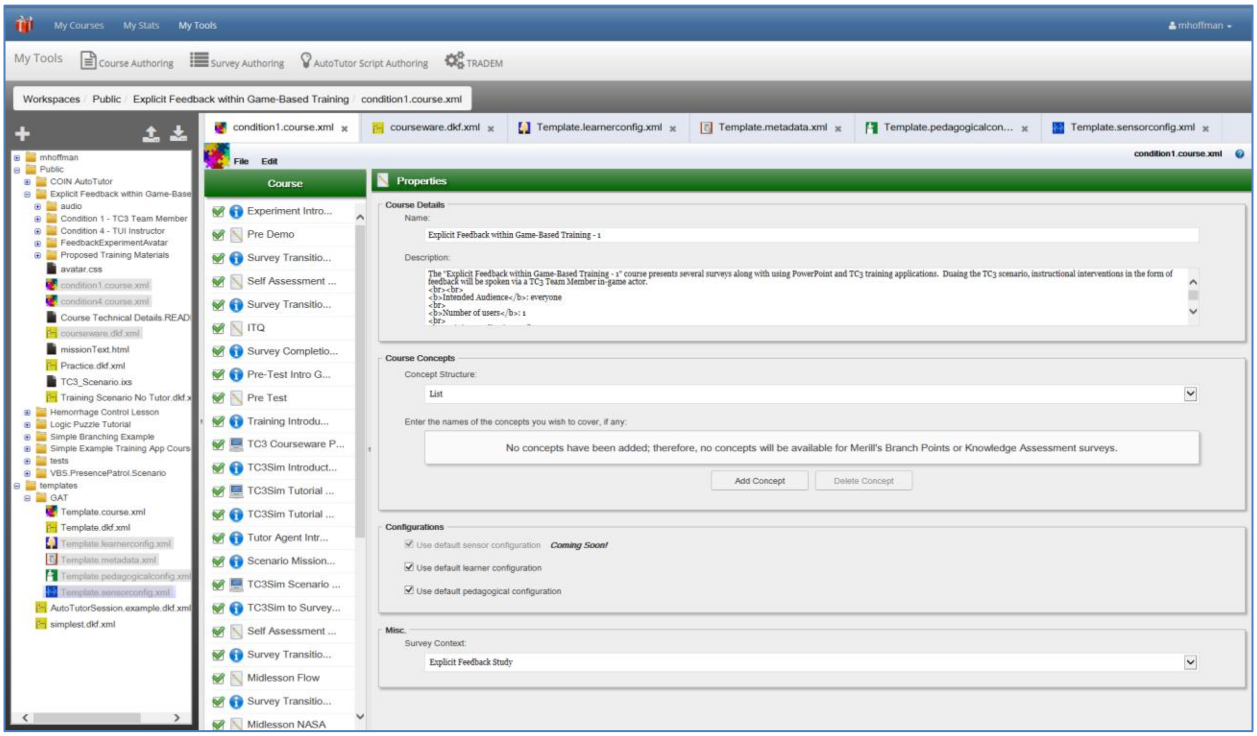


Figure 18. The improved course folder based course authoring tool showing an instance of a course. The tabbed panel across the top middle of the page depicts how an author could edit multiple GIFT files across course folders.

3rd Generation Web Course Authoring

Currently under development, the third generation of web-based GIFT authoring tools builds upon the existing interface and aims to increase the ease of authoring simple GIFT courses. One of the major changes will be the shift from a file explorer view that forced users to understanding the differences between the various GIFT file types, to a simpler layout centered around course flow. In addition, the new layout will increase the screen real-estate available for authoring course objects, which is currently an issue on smaller resolution monitors like laptops. The new interface will allow authors to insert course objects from a toolbar into a course flow workspace providing a key visualization component that is missing in the current authoring tool. By visualizing and interacting with course objects instead of a file system, the author will be able to quickly layout a course and understand, at a high level, the learner's experience when taking the course. From there, authors can edit each course object, manage course properties and media, and eventually preview the authored course.

Another major improvement to the authoring tool will be the overhaul of the interface used to create surveys. The key concerns with the current survey authoring tool is that it is not tightly integrated with course authoring, does not easily traverse the hundreds of existing questions and surveys, and demands an overwhelming exploration of the interface to create even a common quiz or test. With the next generation of course authoring, users will be guided through the survey authoring experience in a way that easily facilitates creating tests to assess knowledge (e.g., test) or collect user information (e.g., self-assessment manikin). Additionally, selecting from existing surveys will be changed to prioritize established surveys (e.g., task load index) and personally created surveys over surveys created for public courses (e.g., Hemorrhage Control Test) or by other GIFT users.

While each version of the GIFT course authoring tool addressed critical issues with creating courses for a generalized ITS framework, there still exists a fundamental issue of providing an easy-to-use interface for creating domain assessment models around interactions in an external training application. To date, GIFT requires course authors, including subject-matter experts, to encode assessment conditions and instructional strategies in a DKF. The DKF is the hardest configuration file to understand due to the nature of its ability to contain a generalized representation of any domain for any training application in addition to defining simple pedagogical rules and instructional tactics (e.g., feedback) to deliver.

Another barrier to authoring assessment rules for a training application is that there is no direct communication between GIFT authoring and the external training application. This disconnect forces authors to manually extract information about objects, locations, actions, and events from the training application (e.g., VBS) and insert that information into GIFT. For example, when creating the GIFT Presence Patrol course, there were several assessment conditions that required coordinates from the environment in the VBS scenario. To obtain these coordinates, the author had to first understand the type of data being provided by VBS when the learner was interacting in the training environment because that coordinate type (e.g., local/internal, geocentric, geodetic) would need to be compared to the values being authored in the conditions. Next, there needed to be a way to capture these coordinate values at specific locations of interest. In VBS, this was done by walking the terrain during course authoring until the actor in the game reached a specific location. At that point in time, network traffic was captured until the appropriate message was received and decoded. Finally the author could copy and paste the coordinate value into the GIFT authoring tool. Although this experience will most likely differ across training applications, there needs to be an easier and more natural way for authors to inject external training application scenario information into GIFT competency models for assessments.

AUTHORING WITH GIFT WRAP

The focus of the existing suite of GIFT course authoring tools has been to provide a single location where an author can create and deliver courses for learners and ITS research purposes (e.g., user studies). Although the experience is becoming easier for users of all types, the toughest task will always be extracting authoritative knowledge from subject-matter experts in a way that is convenient and instinctive in a computer setting while circumventing the prerequisite of understanding course design principles and complex ITS authoring interfaces. GIFT is developing a solution called GIFT Wrap whereby experts, teachers, and trainers can all create tutors while leveraging existing training applications they are familiar with. Essentially GIFT Wrap is the first attempt at redefining the assessment authoring experience by providing a means for an author to build tutoring content while interacting with content creation tools associated with a specific (training) application.

GIFT Wrap is an application that is separate from the external training application. The reason for this stems mainly from not knowing which training applications might be integrated with GIFT in the future in addition to understanding that some systems are proprietary or closed and injecting GIFT functionality is not possible. Although GIFT Wrap and the training application are two distinct user interfaces, both applications are ideally started on the same system in order to have the windows collocated in some manner. This is important so that an author using GIFT Wrap has the ability to author both the training scenario and the assessment rules at the same time.

Once both GIFT Wrap and the training application are running, the user can connect the two applications. This will enable communication for authoring purposes. In the initial version, GIFT Wrap will support authoring "checks on learning" surveys meant to assess a learner's comprehension on some activity being presented in the external training environment. After selecting to create a check on learning object, the author is allowed to author the question text to present to the learner. Next, responses to the question are authored. Question responses can be in the form of simple text or selecting an object from the connected training application. By choosing to use an object from the training application, the author is creating an assessment experience where the learner can also select that object from within that environment during course execution. For example, the object could be an actor in a game, in which case the learner could select the visual representation of that actor in the environment in a way that is natural such as by looking at the actor and selecting with the computer's mouse. This is one of the most important features of GIFT Wrap and can be seen in Figure 19. Both the author and learner use the same native training application to create and evaluate assessments respectively. In the past, GIFT would require the learner to answer the question through the tutor user interface (TUI) webpage and not the training application window. Furthermore, when authors wanted to associate a question's choice with an object from a training scenario, they are forced to create a representation of that object, either text or image based, manually in the GIFT survey authoring system. Now the interaction requires a simple selection from a list during authoring.

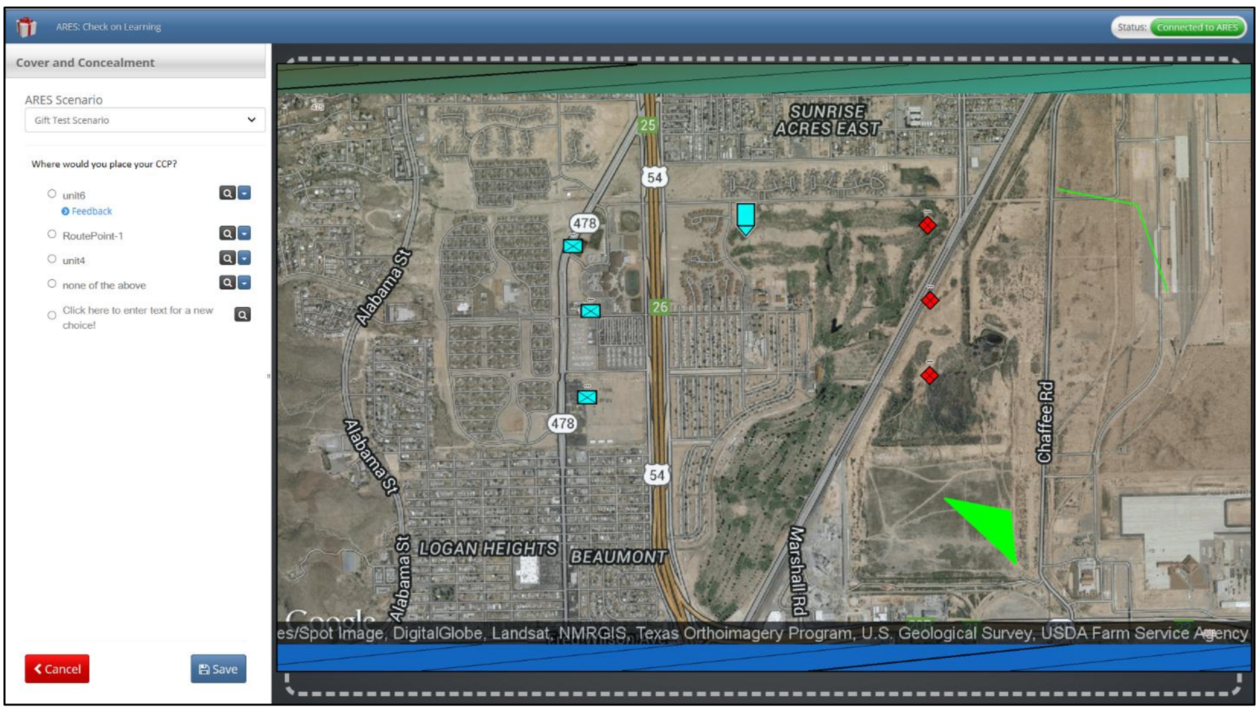


Figure 19. An example GIFT Wrap authoring experience showing how an author could create a check on Learning GIFT course object (on the left-hand side) alongside the native training application (Augmented Reality Sandtable [ARES] on the right-hand side)

Another key feature offered by this interaction between GIFT Wrap and the external training application is the ability to support scenario creation at the same time as authoring a check on learning. Essentially, the training scenario doesn't have to be complete before starting the GIFT Wrap experience. The author is free to create objects natively in the training application at any point using that application's scenario editor. Once an object is created, it will be available to select in the GIFT Wrap window. After a response to an assessment question is authored, the user has the option of providing feedback for that response. This feedback will be presented to the learner alongside the question during GIFT course execution. After creating a check on learning course object, the process can be repeated for each assessment needed during scenario execution. As tutoring elements are created using GIFT Wrap, a series of files are being created in a course folder unbeknown to the author. At some point, the author will need to publish that course folder so that it is available to the learner to take through the GIFT dashboard, thereby closing the loop on course creation using the GIFT Wrap application.

The first external training application being integrated with GIFT Wrap is the Augmented Reality Sandtable (ARES). ARES will be used to help define requirements for GIFT Wrap and serve as an example for other developers to extend GIFT Wrap. In addition the integration intends to support ongoing research of applying tutoring to ARES scenarios.

ARES

Sand tables are low-tech support tools that have been a prominent figure in battle planning since the Stone Age (Smith, 2009). Throughout history, there is evidence that leaders have replicated battlespaces in some form of a sand table to visualize units, soldiers, and other friend and foe military personnel. Today sand tables are still effectively used to model the terrain of a particular battlespace for purposes of tactical planning and training. Leaders use sand tables to compare alternative courses of action during mission planning and rehearsal to evaluate effectiveness of maneuvers and fires, and use this information to determine how best to proceed (Smith, 2009). Sand table exercises (STEXs) have long been recognized as a formal, effective means to conduct tactical training, using a sand table, with an emphasis on cognitive skill development (e.g., spatial orientation) and tactical decision making. A typical sand table used in military schoolhouses consists of four legs supporting a large (e.g., 4 ft x6 ft) tray of sand deep enough to build terrain, and other abstract elements meant to represent different types of terrain and military personnel. ARES is designed to merge the simplicity and low cost of a traditional sand table with multimedia capabilities, including commercial off-the-shelf products such as a projector, monitor, laptop/tablet, and gestural sensors (i.e., Microsoft Kinect), to optimize training in tactical decision making as shown in depicted in Figure 5. This essentially digitizes the terrain reasoning experience users have with the sand table, thereby offering a new set of enhanced training capabilities normally seen in television and movies. ARES has been fielded to several school houses for evaluation and is generating interest from other organizations. It is because of the recent evolution and the importance of this commonly used training environment that ARES is ideal for integrating with GIFT.

Figure 20. An example ARES scenario showing entity locations, terrain features and tactical overlays. The ARES system also provides a tablet interface to facilitate authoring.

In 2015, there was a brief effort that integrated GIFT with ARES. That instance enabled GIFT to display an ARES scenario on the sand table and present the learner with an assessment question on the TUI that related to the image. Each of the choices for the assessment question related to an object in the scenario. Therefore, the learner would first analyze the situation on the sand and then choose the object in GIFT that answered the question. The course presented a series of these situations in an effort to capture the learner's knowledge level on military tactics. One of the major difficulties in creating this experience was authoring the GIFT course. The author was required to have an in depth understanding of how to create the GIFT DKF to sequence events between GIFT and ARES. Moreover, the assessment questions and references to ARES scenarios and scenario objects had to be entered manually into the GIFT authoring tool rather than within ARES or some overlay authoring tool. The first GIFT Wrap instance with ARES will be addressing lessons learned from this integration effort and diminish the knowledge required to create sand table tutors for authors of all experience levels. As with building GIFT, the generalizability of GIFT Wrap will present various challenges.

Challenges

While developing GIFT, the focus has been on providing a generalized framework for course creation, research, and software development in a way that is supportive to all knowledge levels. The same consideration will be used when engineering the first version of GIFT Wrap. One of the key challenges shared between both GIFT and GIFT Wrap is the ability to communicate with external training applications (e.g., PowerPoint, VBS). In GIFT, this is handled in the gateway module via logic called interop plugins. There is an interop plugin instance for each type of training application with which GIFT has been integrated. When the gateway module is started for a course on a learner's computer, the necessary interop plugins are configured for any applications required during that course. This configuration logic opens the control and communication between GIFT and the external training application. Most of the communication taking place in the gateway module is done using the network layer of the learner's computer. Currently, the network configuration is managed by a configuration file, which is normally only edited when a problem occurs. For GIFT Wrap, these parameters should be exposed in a user interface to improve usability.

The main objective of GIFT Wrap is to facilitate a native authoring experience as much as possible. To achieve this, GIFT Wrap will most likely have to expand upon the communication capabilities of interop plugins that are currently structured around taking a course rather than authoring. Any training application to be integrated with GIFT Wrap will need to provide scenario authoring information that can be associated with domain assessment models in GIFT. This information will then be leveraged during course execution in a way that is seamless to both the author and learner. For example, if the author would like to assess the learner on the best objective rally point in ARES by having the learner select the appropriate location in the ARES environment and then the ARES scenario authoring tool integrated with GIFT Wrap would need to provide the author with list of selectable objects (e.g., units, tactical graphics, basic geometry). However, a mere flat list of objects found in the training environment can be too burdensome to search through when there are hundreds or even thousands of possible objects. Therefore, GIFT Wrap needs to support the ability for connected training applications to provide a more sophisticated mechanism by which the author can quickly find and select the desired object. For systems that are more conducive to changes in their user interface, allowing authors to select scenario objects from within the training application scenario editor will further bolster the intuitiveness of creating a GIFT course using GIFT Wrap. Another requirement for most systems being integrated will be the ability for GIFT Wrap to control that application at some level. For example, the GIFT Wrap user may choose to edit a GIFT course object that assesses an ARES scenario. In that case, GIFT Wrap will need the ability to change what map is displayed on the table. Without this ability, the author would have to synchronize the two authoring interfaces separately increasing the opportunity of errors and misconceptions.

Included in this effort are useful changes to the GIFT runtime. In GIFT's current state, learners are required to answer assessment questions through the GIFT TUI using a mouse and keyboard. That logic will need to be improved upon to allow the learner to select an object in the external training application when that question is presented. The training application will then need to provide that information to the GIFT gateway module. The TUI will then show the user's selection and submit the question's answer for assessment. Another change to GIFT is the ability to present feedback for an assessment question. The feedback can be authored either in GIFT Wrap or the GIFT course authoring tool and will be presented in a structured review during course execution.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

Following the initial ARES use case, the next iteration on GIFT Wrap will involve the VBS training environment. This will extend the authoring features in the GIFT Wrap tool to incorporate avatar and virtual agent/object movements to inform assessment logic. The domain of land navigation will serve as the use case, with scenarios and terrain models already in place to support development. Initial assessment models will involve situational awareness prompts triggered by production rules built around the various scenario objects present in the environment. The goal is to build these assessment rules within the VBS mission editor, where those configurations are used to auto populate DKF fields required for run-time execution. This will include defining a task concept, building a set of measures around that concept that associate with performance states, identifying the various state transitions that can occur within a single concept, and associating domain-level tactics and feedback with all available performance transitions. An additional goal is to incorporate both scenario-derived performance assessments with survey-based assessments to provide two levels of interaction to gauge knowledge and skill comprehension. This approach will also target building in scenario adaptations/branches that can be managed by GIFT's pedagogical model.

Another avenue sought after is extending GIFT-delivered training into real-world environments, with land navigation serving as an excellent use case to support this capability. With the incorporation of wearable sensing technologies and cellular networks that support data sharing, the assessments and logic built within VBS should translate well to a live land navigation course. This will require investigating how best to address the authoring requirements in GIFT Wrap to support this use case and what external technologies and training applications must be incorporated. This includes pairing GIFT with a variety of sensor and cellular products along with identifying how to visualize the environment to support easy authoring (e.g., integrating with Google Maps to build scenario objects based on real-world terrain). Another issue to address is handling large amounts of data that must be shared over a cellular network. This will involve determining GIFT processes that should be managed locally on a cellular device versus modules that operate on an instance of a GIFT server.

The biggest challenge when authoring a GIFT course is establishing a user-friendly way to build assessment models. Those models are where the micro-adaptive run-time power of GIFT is configured. It is that piece that is arguably the most relevant part for those concerned with scenario-based training exercises. GIFT Wrap is the first attempt at redefining the assessment authoring experience by bridging it with scenario/mission editing tools provided by training applications.

Admittedly, these tools have only began to reveal what it takes to author an adaptive course using a generalized framework like GIFT. GIFT is continuously evolving to support ITS research. As new capabilities are implemented, course creators will need easy to use and intuitive authoring tools. In the future, course authoring will have to consider, among other things, team tutoring, additional training applications, simple rule based branching, agents, and collaboration including publishing and role-based user interface tailored experiences.

REFERENCES

Brawner, K., Holden, H., Goldberg, B., & Sottilare, R. (2012, January). Recommendations for Modern Tools to Author Tutoring Systems. In The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)(Vol. 2012, No. 1). National Training Systems Association.

Hoffman, M. & Ragusa, C. (2014). Unwrapping GIFT: A Primer on Authoring Tools for the Generalized Intelligent Framework for Tutoring. In Proceedings of the Second Annual GIFT Users Symposium (GIFTSym2)

Smith, R. (2009). The long history of gaming in military training. *Simulation & Gaming*, *41*(1), 6-19.

Sottilare, R. A., Goldberg, B. S., Brawner, K. W., & Holden, H. K. (2012, January). A Modular Framework to Support the Authoring and Assessment of Adaptive Computer-Based Tutoring Systems (CBTS). In The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)(Vol. 2012, No. 1). National Training Systems Association.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

ABOUT THE AUTHORS

Mr. Michael Hoffman *is a senior software engineer at Dignitas Technologies. He is the lead engineer on GIFT and has been responsible for ensuring that the development of GIFT meets the evolving customer requirements in addition to supporting both intelligent tutoring for computer based training and intelligent tutoring technology research of the growing user community. Michael manages and contributes support for the GIFT community through various mediums including the GIFT portal (www.GIFTTutoring.org), annual GIFT Symposium conferences and technical exchanges with ARL.*

Mr. Chris Markuck *is the Primary Principal Investigator and Technical Lead for the ARES project. He has over 12 years of experience with a wide variety of software applications, all involving modeling and simulation, some for the Army and some for commercial game applications.*

Dr. Benjamin Goldberg *is an adaptive training scientist at the Army Research Laboratory's Human Research and Engineering Directorate. He leads research focused on instructional management within ARL's Learning in Intelligent Tutoring Environments (LITE) Lab and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT).*

Extending GIFT with a Reinforcement Learning-Based Framework for Generalized Tutorial Planning

Jonathan Rowe[1], Megan Frankosky[1], Bradford Mott[1], James Lester[1], Bob Pokorny[2], Wilbur Peng[2], and Benjamin Goldberg[3]
North Carolina State University[1], Intelligent Automation, Inc.[2], U.S. Army Research Laboratory[3]

INTRODUCTION

Planning tutorial actions is a critical component of intelligent tutoring systems (ITSs). Tutorial planners determine how pedagogical tasks and scaffolding are tailored to learners at run-time. Devising tutorial planners that generalize across students, learning environments, and domains is an important challenge for the field. These challenges are amplified by the increasing complexity of advanced learning technologies, such as simulations (Johnson 2010; Kim et al., 2009; Mislevy et al., 2014) and digital games (Shute, Ventura & Kim, 2013; Rowe & Lester, 2015; Baker, Clarke-Midura & Ocumpaugh, in press). Data-driven methods for devising tutorial planners, such as reinforcement learning, show especial promise for addressing these challenges, due to their capacity to automatically induce pedagogical models from large datasets characterizing student behavior and learning outcomes. Further, they introduce the possibility of devising ITSs that automatically refine and improve their pedagogical strategies over time.

Reinforcement learning has been the subject of growing interest in the ITS community over the past several years (Barnes & Stamper, 2008; Beck, Woolf & Beal, 2000; Chi, VanLehn & Litman, 2010; Rowe and Lester, 2015). This work has emphasized probabilistic models of student behavior, as opposed to explicit models of cognitive states. For example, Chi, VanLehn and Litman (2010) used MDPs to model tutorial dialogues, devising pedagogical tactics directly from student data in the Cordillera physics tutor. Rowe and Lester (2015) utilized a related approach to dynamically tailor story events in a narrative-centered learning environment for middle school science. Barnes and Stamper (2008) modeled students' logic proof sequences as MDPs in order to automatically generate context-appropriate hints. Rafferty utilized inverse reinforcement learning techniques to draw inferences about student goals and misconceptions based upon logs of students' freeform math problem-solving actions. Complementary work investigating partially observable Markov decision processes (POMDPs) to model tutorial planning has been explored, yielding novel approaches for compactly representing planner state representations (Mandel et al., 2014; Folsom-Kovarik, Sukthankar & Schatz, 2011).

In this paper, we describe our recent work on a modular reinforcement learning framework for tutorial planning in the Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden 2012). We focus on inducing tutorial planning models directly from learner data to support a broad range of tutorial interventions, which share a generalized encoding of instructional strategies and tactics across multiple learning environments. This work is part of a collaborative project between North Carolina State University (NCSU), Intelligent Automation, Inc. (IAI), and the U.S. Army Research Laboratory (ARL) to investigate generalizable data-driven tutorial planning that operates across multiple training environments. We are investigating modular reinforcement learning-based tutorial planning in the domain of counterinsurgency and stability operations (COIN) training, with a focus on adaptive hypermedia and simulation-based learning environments. We describe the initial design and development of a generalized tutorial planner, whose design is inspired by Chi's ICAP framework (2009), which differentiates between passive, active, constructive, and interactive forms of learning. In addition, we describe a pilot study that was conducted with university ROTC cadets, which was designed to test the impact of ICAP-inspired tutorial interventions on learning outcomes during COIN training. We conclude with a discussion of design recommendations for GIFT to facilitate reinforcement learning-based tutorial planning induced from simulated- and human-student data.

DESIGN OF INSTRUCTIONAL STRATEGIES AND TACTICS

There is a longstanding research literature on the design and effectiveness of alternate instructional methods across a range of learning environments, including advanced learning technologies. To inform the design of scaffolding interventions for our project, we conducted a brief literature review on the effectiveness of instructional strategies and tactics for learning. This includes reviews of John Hattie's work synthesizing over 800 meta-analyses about the factors that influence learning (Hattie, 2008), Durlach and Spain's Framework for Instructional Technology (Durlach and Spain, 2014), Chi et al.'s work on reinforcement learning-based tutorial planning in dialogue systems (Chi, VanLehn, Litman, and Jordan, 2011), Woolf and McDonald's early work on instructional strategies and tactics in tutoring discourse (Woolf, 1984), and Chi's work on the ICAP framework (Chi, 2009). We have also begun to examine evidence about instructional interventions from the TARGET (Training Aide: Research and Guidance for Effective Training) web tool. Based upon this literature review, the project team decided to focus on the *feedback* and *support* categories of Durlach and Spain's framework (2014), as well as the *constructive*, *active*, and *passive* forms of instructional activities described by Michelene Chi (2009). Utilizing these complementary taxonomies of instructional methods, we designed and developed a set of instructional techniques and strategies that were compatible with a pair of training environments for COIN operations: the UrbanSim Primer hypermedia-based learning environment and UrbanSim simulation-based learning environment.

URBANSIM PRIMER Hypermedia-Based Learning Environment

The UrbanSim Primer is a hypermedia-based learning environment that provides direct instruction on complex counterinsurgency and stability operations (Figure 1). Developed by the USC Institute for Creative Technologies, the UrbanSim Primer presents hyperlinked video, audio, text, and diagrams on a range of doctrinal concepts of counterinsurgency, including the importance of population support, crucial strategies such as Clear-Hold-Build, resources and processes for intelligence gathering, and issues in successful execution of COIN operations. The Primer also provides preliminary instruction on usage of the UrbanSim simulation. The Primer's course material is organized in terms of 7 lessons, and there are three versions of the material which vary in duration and comprehensiveness: Complete (~2.5 hours to complete), Partial (~1.5 hours to complete), and Limited (~1 hour to complete). In this project, we utilize a subset of the Limited version of the Primer course material, which includes excerpts from Lessons 1, 2, 3, 4, 5, and 7.

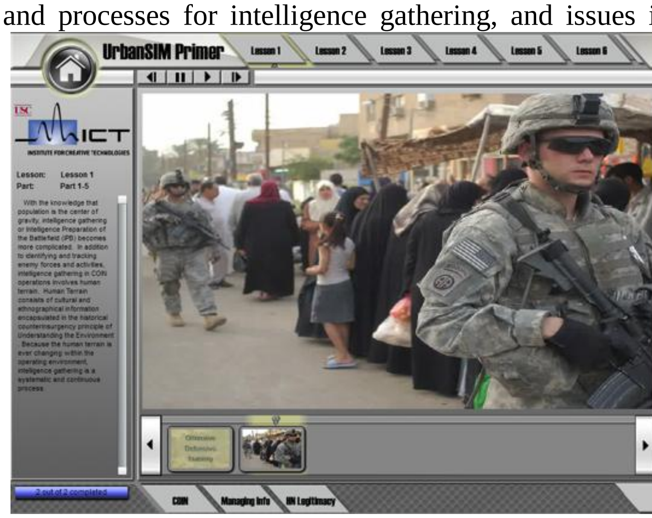


Figure 21. UrbanSim Primer hypermedia-based learning environment.

The UrbanSim Primer is traditionally delivered as a web-based application using Adobe Flash. In order to facilitate its integration with GIFT, we extracted the Primer's video content into a set of WMV video files and embedded them into Microsoft PowerPoint shows. This enabled GIFT to launch and manage the UrbanSim Primer, interleaving embedded assessments and instructional interventions with Primer content, as well as automatic logging of learner actions within PowerPoint (e.g., clicking to advance a slide, over-dwelling on a slide).

URBANSIM Simulation-Based Learning Environment

UrbanSim is an open-ended simulation-based virtual training environment for counterinsurgency and stability operations (Figure 2). In UrbanSim, learners act as a battalion commander whose mission is to maximize civilian support for the host nation government (McAlinden et al., 2009). Training experiences using UrbanSim resemble computer game play interactions with turn-based strategy games. On each turn, the learner assigns actions for 11 Battalion resources, such as “E Company, A platoon patrols the Malmoud Quarter” or “G Company, B platoon recruits policemen in the Northern Area.” Trainees’ actions, and consequences to their actions, are simulated using an underlying social-cultural behavior engine that determines how the host city’s inhabitants respond to different situations. During each turn, UrbanSim presents (1) situation reports, such as “the Mayor is pleased with the increased electrical power available to citizens,” (2) significant events, such as “an IED exploded at the Gas Station on Hwy2,” and (3) civilian support for the host nation government, visually rendered from an overhead view using game engine technologies. The decision space in UrbanSim is enormous; an analysis by Brian Vogt (2012) estimated that there are approximately $5x10^{27}$ different action sequences that learners can perform in the simulation environment. This complexity points to the need for an assessment framework that can robustly handle the broad range of possible learner action in UrbanSim, as well as the promise of data-driven pedagogy for delivering context-sensitive scaffolding and feedback in different learner states.

Figure 2. UrbanSim simulation-based training environment

In prior work, the IAI team applied a cognitive task analysis method to identify performance patterns of trainees that should become targets for scaffolding and remediation. The task analysis involved taking performance data of real learners, reformatting it into a human-readable style, and asking subject-matter experts to (1) assign scores reflecting learners’ overall proficiency and (2) critique learners’ actions. The subject-matter experts’ critiques addressed a broad range of topics, including what structures were repaired, whom U.S. forces held meetings with, and what security actions were taken against specific threats. Analysts, with the assistance of subject-matter experts, characterized these comments into a small set of scoring rules. Learners’ actions either followed good practice or violated good practice. When learner performance complied with good practice, points were assigned to the learner. When learners violated good practice, points were deducted. Pokorny, Haynes, and Gott (2010) reported that this task analysis method yielded scores with excellent psychometric properties. The scores from experts and from automated scoring systems were valid, as they correlated with time of service in the job, as well as reliable, as experts’ scores correlated well with other experts’ scores.

In this project, we are focusing on two of the performance categories that were identified in the cognitive task analysis from the previous paragraph: (1) *security* and (2) *meetings with host-nation leaders*. Specifically, we are designing instructional strategies and tactics that can be delivered during UrbanSim training, and that are compatible with the *security* and *meetings with host-nation leaders* dimensions of learner performance. We describe the design of these instructional methods in the next section.

GENERALIZABLE INSTRUCTIONAL METHODS FOR COIN TRAINING

We selected four types of instructional techniques for modeling and delivery by GIFT during COIN training.[2] These instructional techniques were selected to enable a common encoding of pedagogical actions across both the UrbanSim and UrbanSim Primer learning environments. The instructional techniques include (1) single-topic coaching, (2) multi-concept review, (3) immediate feedback on unproductive learning behaviors, and (4) no feedback. These instructional techniques are delivered to learners during training using GIFT's Tutor User Interface (TUI). The techniques can be implemented using a broad range of context-sensitive instructional strategies and tactics in either UrbanSim or the UrbanSim Primer. However, at time of writing only the UrbanSim versions of these instructional methods are fully operational.

In this project, single-topic coaching consists of a text-based feedback message about a specific dimension of learner performance in either the *security* or *meetings with host-nation leaders* performance areas of COIN training. These feedback messages can be delivered at the end of any turn in UrbanSim, or alternatively, at the end of an UrbanSim Primer unit, which is presented in the form of a PowerPoint show. The messages are designed to address areas of learner performance that are below expectation along a particular dimension of COIN content understanding. The messages identify a performance dimension that the learner is having difficulty with, and provide a brief excerpt from the *U.S. Army/Marine Corps Counterinsurgency Field Manual* related to the topic. An example of a single-topic coaching intervention is shown in Figure 3.

> Your security actions are too lax given the current risk. Risk depends on the number of current insurgent groups and how active they are.
>
> Read the following excerpt from the *U.S. Army/Marine Corps Counterinsurgency Field Manual* below.
>
> "At the beginning of a COIN operation, military actions may appear predominant as security forces conduct operations to secure the populace and kill or capture insurgents. However, political objectives must guide the military's approach. Commanders must, for example, consider how

Figure 3. Example of single-topic coaching feedback message.

Multi-concept reviews are similar to single-topic coaching, except that they address multiple dimensions of COIN performance simultaneously. They do not necessarily focus only on concepts on which learners are performing below expectation; they review a range of performance dimensions relevant to successful execution of COIN operations. Multi-concept reviews interleave summaries of effective COIN operational practice and excerpts from relevant U.S. Army field manuals, such as the *Commander's Handbook for Strategic Communication and Communication Strategy*. Similar to single-topic coaching, these messages are delivered the end of turns in UrbanSim, or the conclusion of units in the UrbanSim Primer. However, multi-concept reviews cannot occur after any training turn during training. They do not occur during the initial two turns of UrbanSim. Multi-concept reviews are reserved for the third turn, or later, within UrbanSim, in order to ensure that learners have had the opportunity to demonstrate their proficiency across multiple dimensions of

[2] We refer to these pedagogical methods as *instructional techniques,* because it is the nearest term in the GIFT literature to what we mean in this paper. However, it should be noted that the GIFT taxonomy of instructional techniques, strategies, and tactics connotes specific assertions about how different sources of information influence pedagogical actions within GIFT. We do not use the terms *instructional technique* or *instructional strategy* to imply a particular architecture for modeling pedagogical methods in our project; the specific implementation of our instructional methods within GIFT is the subject of ongoing work.

performance. In addition, multi-concept reviews cannot occur on the same turn as a single-topic coaching intervention.

Feedback on unproductive learning behaviors focuses on addressing egregious or inefficient actions performed by learners that have little or no relevance to the learning task within UrbanSim or the UrbanSim Primer. An example of an egregious action includes killing an important civilian leader within the Al Hamra host city, a type of learner behavior that has alternatively been described as *without thinking fastidiously* (Wixon et al., 2012) or *off-task* (Rowe et al., 2009) in the ITS community. Inefficient learner actions include under dwelling and over dwelling on training content or tasks, such as a single turn within UrbanSim or a particular slide in the UrbanSim Primer. Feedback on unproductive learning behaviors can occur after any turn. In addition, feedback on unproductive learning behaviors can occur on the same turn as another instructional intervention, such as single-topic coaching or multi-concept review.

For each of these three instructional techniques, we identified a shared set of pedagogical strategy options that drive the techniques' implementation in a context-sensitive manner. The design of the pedagogical strategies was based upon the ICAP framework (Chi, 2009), which distinguishes between (1) interactive, (2) constructive, (3) active, and (4) passive forms of instructional activities. Because this project does not focus on "interactive" forms of instructional methods, which typically refer to tutorial dialogues, we devised instructional strategies consistent with constructive, active, and passive forms of each technique. In other words, each of the three instructional techniques in this project—single-topic coaching, multi-concept reviews, and unproductive learning behaviors—can be implemented in the form of a passive, active, or constructive intervention.

The passive form of an instructional technique consists solely of a text-based message that participants read prior to continuing with their training. Passive instructional strategies do not require a particular response from the learner beyond clicking a button at the conclusion of the feedback message, but they are efficient and enable learners to promptly return to hypermedia or simulation-based training. The single-topic coaching message in Figure 3 is an example of a passive intervention. The active form of an instructional technique expands upon the passive strategy by prompting learners to highlight key parts of feedback or review messages to identify their most important elements. After the learner completes her highlight, she is presented with an expert highlight of the same instructional message in order to facilitate critical evaluation of her own active learning performance. The constructive form of an instructional technique expands further by prompting learners to briefly summarize, in their own words, the most important parts of the feedback or review message. After the learner finishes writing her summary, she is presented with an expert summary in order to facilitate her own evaluation of her learning performance. In this manner, each of the three instructional strategies encourages varying levels of cognitive engagement with scaffolding during COIN training, while remaining general enough to remain applicable to each form of instructional technique in the project.

The specific instructional tactics utilized to implement each instructional strategy and technique are pre-authored and selected in a context-sensitive manner according to hand-authored rules driven by the assessment framework developed by IAI. These pre-authored messages address particular domain-specific aspects of performance, while enacting particular combinations of instructional techniques and strategies. Run-time decisions about which instructional techniques and strategies to deploy during training with UrbanSim and the UrbanSim Primer will ultimately be controlled by a data-driven tutorial planner, which we outline in the next section.

MODULAR REINFORCEMENT LEARNING FRAMEWORK FOR TUTORIAL PLANNING

We will formalize tutorial planning as an instance of modular reinforcement learning. Modular reinforcement learning is a multi-goal extension of classical single-agent reinforcement learning, which involves decomposing an agent planning task into multiple distinct sub-problems, which are solved separately and combined at execution time (Bhat, Isbell, & Mateas, 2006; Karlsson, 1997). Modular reinforcement learning tasks are formally defined in terms of *N* concurrent Markov decision processes (MDPs), $M = \{\mathrm{M_i}\}_1^{\mathrm{N}}$, where each M_i = $(\mathrm{S_i}, \mathrm{A_i}, \mathrm{P_i}, \mathrm{R_i})$ corresponds to a sub-problem in the composite reinforcement learning task. Each agent M_i has its own state sub-space S_i, action set A_i, probabilistic state transition model P_i, and reward model R_i. The solution to a modular reinforcement learning problem is a set of *N* policies, $\pi^*=\{\pi_i^*\}_1^{\mathrm{N}}$, where π_i is the optimal policy for the constituent MDP M_i. Whenever two policies π_i and π_j with $i \neq j$ recommend different actions in the same state, an arbitration procedure must be applied.

Tutorial planning in virtual training environments is naturally represented as a modular reinforcement learning problem. Different types of tutorial decisions are modeled separately as MDPs. For each MDP, state consists of the learner's state and history as well as the learning environment conditions; actions represent the pedagogical decisions the planner can perform; a probabilistic state transition model encodes how learners, and the learning environment, respond to the planner's tutorial decisions; and a reward model encapsulates measures of trainees' learning outcomes, which the tutorial planner seeks to optimize. The solution to a modular reinforcement-learning problem is a set of policies, or mappings between states and tutorial actions, that govern how the tutorial planner scaffolds trainees' learning. If two policies conflict, externally defined arbitration procedures specify which policy prevails.

By decomposing tutorial planning into multiple sub-problems, we can reduce the complexity of reinforcement learning by reframing the task in terms of several smaller, concurrent Markov decision processes. To perform this decomposition, we employ the concept of an adaptable event sequence (AES), an abstraction for a series of one or more instructionally related events that, once triggered, can unfold in several different ways within the learning environment (Rowe and Lester, 2015). In this project, we will utilize two nested levels of AESs. A set of high-level AESs will control decisions about which instructional technique to deploy at particular points of training, such as single-topic coaching, multi-topic review, feedback on unproductive learning behaviors, or no intervention. A set of lower-level AESs will control decisions about which instructional strategy to be utilized to implement the technique, such as feedback requiring a constructive learner response, or feedback requiring an active learner response, or feedback permitting a passive learner response. Distinct MDPs will be utilized to encode each of these AESs for each learning environment (i.e., UrbanSim and UrbanSim Primer). The instructional tactics that implement each strategy will be chosen based upon the concept that is scored lowest by the assessment framework.

PILOT STUDY

We conducted a pilot study to test our initial instructional interventions with a group of cadets from North Carolina State University's ROTC program. The pilot study was intended to pilot test the instructional interventions and content knowledge assessments that had been designed during the project's first year. In addition, we sought to collect data on learner responses to the instructional interventions, which would drive initial efforts to utilize reinforcement learning for inducing tutorial planning policies.

There were 23 undergraduate seniors (20 male, 3 female) who participated in the pilot study. The cadets had minimal prior experience with counterinsurgency and stability operations training, and no prior experience using UrbanSim or the UrbanSim Primer. The study took place over the course of three weeks during cadets' regular ROTC laboratory class period. All study activities were completed on laptop computers. All participants used the same version of the software and followed the same study procedure; there were no experimental conditions. Learners completed all study activities independently.

During the first week of the pilot study, participants attended two study sessions. During the first session, participants completed a brief demographic questionnaire, as well as a pre-test on COIN content knowledge. The overall session lasted approximately 20 minutes. The pre-test was delivered and managed by GIFT. The pre-test consisted of 20 multiple-choice items that were devised to align with course material from the UrbanSim Primer. The questions spanned a broad range of COIN concepts and difficulty levels, and they included both fact-level and application-level questions. The items were designed to balance between different question formats, including traditional questions, gap-fill questions, and cloze questions. Prior to administering the pre-test, it was reviewed by a COIN subject-matter expert for content validity.

During the second session, participants viewed the UrbanSim Primer, which was delivered in the form of a PowerPoint presentation. Embedding the UrbanSim Primer's video content in PowerPoint enabled GIFT to manage and log student interactions with the hypermedia-based learning environment[3]. After participants finished the Primer, they completed a second content knowledge assessment, or mid-test, on COIN. The mid-test consisted of 20 multiple-choice items that paralleled the pre-test, but utilized distinct language and question styles. After the mid-test, participants completed a brief interactive tutorial on UrbanSim. The tutorial introduced participants to the controls and user interface of the UrbanSim simulation-based learning environment. The second session lasted approximately 2 hours.

The following week, learners attended a third study session where they completed two scenarios in UrbanSim: *Al Hamra* and *Al Hamra2*. The two scenarios took place in a fictional Iraqi city, and they each involved distinct training events and mission requirements. The first scenario, Al Hamra, was intended to serve as a pre-test of COIN practice in UrbanSim. Thus, learners did not receive instructional interventions during the training scenario (e.g., single-topic coaching, multi-topic review, or feedback on unproductive learning behavior). The second scenario, AlHamra2, was intended to serve as a pilot test of the instructional interventions for UrbanSim. The instructional interventions were shown in a web browser adjacent to the UrbanSim application window. Learners could switch their focus between the two software applications, alternating between interacting with UrbanSim and reviewing instructional interventions in the browser after each turn.[4] Instructional interventions were delivered according to a policy that combined hand-authored rules (e.g., multi-topic reviews could only occur after the third turn of UrbanSim training) and stochastic decision making (e.g., select between constructive, active, or passive interventions according to a uniform random policy). The instructional interventions were not yet delivered according to policies induced using reinforcement learning techniques. After receiving an instructional intervention, learners were given the opportunity to rate the helpfulness of the intervention on a scale of 1 to 5. Learners interacted with each UrbanSim scenario for 8 turns, after which they exited the software and started a new scenario or finished their participation in the day's study session. The study session lasted approximately 2 hours.

---

[3] The pilot study version of the UrbanSim Primer did not yet have support for our instructional interventions, due to development schedule constraints.

[4] A software error during the second study session constrained the set of instructional interventions to include only a particular multi-topic review question for many participants. This reduced the breadth of instructional interventions that were sampled during UrbanSim training, and may have reduced learners' COIN training performance. The issue was resolved prior to the fourth study session.

During the third week, learners attended a fourth study session where they completed another two scenarios in UrbanSim: *Al Hamra3* and *Al Hamra*. As in the earlier session, the two scenarios took place in the same fictional Iraqi city but involved different training events and mission requirements. The final scenario was identical to the first one completed during Week 2 of the study. During the AlHamra3 scenario, learners received instructional intervention messages in an adjacent web browser. As in the prior study session, instructional interventions were delivered according to a policy that combined hand-authored rules and stochastic decision making. In addition, learners were prompted to rate the helpfulness of each intervention on a scale of 1 to 5. Learners received no instructional interventions during the final *Al Hamra* scenario; the final scenario was intended to serve as a post-test of COIN practice in UrbanSim. As in the prior week, learners interacted with each UrbanSim scenario for 8 turns. After completing both scenarios in UrbanSim, participants completed a COIN content knowledge post-test. The post-test consisted of 20 multiple-choice items, which paralleled the pre-test and mid-test but utilized different language and question styles for each item. After the post-test, participants concluded by leaving the study room.

At the time of writing, data analysis from the pilot study is ongoing. Preliminary results suggest that learners did achieve significant content learning gains from pre-test (M=11.4, SD=1.63) to mid-test (M=14.8, SD=1.82), as well as from pre-test to post-test (M=13.1, SD=2.15), $F(2, 38) = 25.81$, $p < .001$. However, the delay between the UrbanSim Primer and post-test yielded slightly reduced learning gains relative to the mid-test. This is unsurprising, given that the multiple-choice tests assessed COIN factual knowledge; participants primarily utilized UrbanSim between the mid-test and post-test, which focused on COIN practice. Empirical data on cadets' learning gains will inform iterative refinements to the design of our COIN instruction and content knowledge assessments. In addition, we intend to utilize UrbanSim performance data, as well as data on learner ratings of the instructional interventions, in order to develop *simulated students*, which will enable generation of synthetic training episodes for inducing initial tutorial planning policies using modular reinforcement learning. We will explore several different approaches to devising simulated students and reward schemes as we induce tutorial policies. We will analyze the resulting tutorial policies using learner-centered design as a theoretical lens, and integrate the most promising policies with GIFT during Year 2 in order to test them with a new cohort of learners.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

Data-driven approaches to tutorial planning, such as reinforcement learning, show significant promise for devising effective models of instructional techniques and strategies for complex domains and learning environments. In this paper, we have described recent work from a research collaboration between North Carolina State University, Intelligent Automation, Inc., and the U.S. Army Research Laboratory on a modular reinforcement learning framework for generalized data-driven tutorial planning with GIFT. We are investigating tutorial planning in the domain of counterinsurgency and stability operations (COIN) training, with a focus on adaptive hypermedia and simulation-based learning environments. We have devised a shared set of instructional techniques and strategies that are applicable to multiple learning environments, and which are inspired by Durlach and Spain's Framework for Instructional Technology (2014) and Chi's ICAP framework (2009). Specifically, we are developing instructional scaffolding that consists of context-sensitive feedback and support delivered in the form of passive, active, constructive learning interventions. In order to pilot test these interventions, we conducted a pilot study with university ROTC cadets, which was designed to test the impact of ICAP-inspired tutorial interventions on learning outcomes during COIN training. Data analysis from this study is ongoing, and will inform the iterative refinement of the instructional interventions and COIN knowledge assessments, as well as the creation of simulated students for generating synthetic training episodes for the modular reinforcement learning framework.

Several extensions to GIFT can be recommended in order to enable this line of research, some of which are under active planning and development. Currently, GIFT has limited support for encoding instructional techniques, strategies, and tactics in terms of modular policies. Devising an agent-based architecture for encoding pedagogical methods would facilitate the integration of modular reinforcement learning features with GIFT. In addition, integrating support for arbitrating potential conflicts between competing policies will be an important related feature after modular tutorial planning policies are supported. Incorporating support for parameterized stochastic control of pedagogical actions will be important for reinforcement learning in GIFT. Enabling a tutorial planner to explore the decision space of tutorial actions is a critical part of data-driven methods for inducing instructional models. Finally, elaborating upon GIFT's taxonomy of instructional techniques, strategies, and tactics to provide guidance on how different sources of information, including domain-specific information, might inform high-level instructional decisions within the GIFT architecture would be valuable.

REFERENCES

Baker, R., Clarke-Midura, J., Ocumpaugh, J. (in press) Towards General Models of Effective Science Inquiry in Virtual Performance Assessments. To appear in *Journal of Computer Assisted Learning*, Special Issue on Learning Analytics in Massively Multiuser Games, Virtual Environments and Courses.

Chi, M.T.H. (2009). Active-Constructive-Interactive: A conceptual framework of differentiating learning activities. *Topics in Cognitive Science*, *1*(1), 73-105.

Chi**,** M., VanLehn, K, Litman, D. & Jordan, P. (2011). An evaluation of pedagogical tutorial tactics for a natural language tutoring system: A reinforcement learning approach. *International Journal of Artificial Intelligence in Education*, *21*(1-2), pp. 83-113.

Durlach, P. & Spain, R. (2014). Framework for Instructional Technology: Methods of Implementing Adaptive Training and Education (ARI Technical Report No. 1335). Fort Belvoir, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Hattie, J. (2013). *Visible learning: A synthesis of over 800 meta-analyses relating to achievement*. New York: Routledge.

Kim, J., Hill, R. W., Durlach, P., Lane, H. C., Forbell, E., Core, M., Marsella, S., Pynadath, D., & Hart, J. (2009). BiLAT: A game-based learning environment for practicing negotiation in a cultural context. *International Journal of Artificial Intelligence in Education*, *19*(3), 289-308.

Mandel, T., Liu, Y. E., Levine, S., Brunskill, E., & Popovic, Z. (2014). Offline policy evaluation across representations with applications to educational games. *Proceedings of the 13th International Conference on Autonomous Agents and Multi-Agent Systems* (AAMAS), pp. 1077-1084, Paris, France.

Mislevy, R. J., Oranje, A., Bauer, M. I., von Davier, A., Hao, J., Corrigan, S., Hoffman, E., DiCerbo, K., & John, M. (2014). *Pyschometric Considerations in Game-Based Assessments*. Retrieved from https://www.ets.org/research/policy_research_reports/publications/white_paper/2014/jrrx

Rowe, J., McQuiggan, S., Robison, J., & Lester, J. (2009). Off-Task Behavior in Narrative-Centered Learning Environments. *Proceedings of the 14th International Conference on Artificial Intelligence in Education*, pp. 99-106, Brighton, U.K.

Shute, V. J., Ventura, M., & Kim, Y. J. (2013). Assessment and Learning of Qualitative Physics in Newton's Playground. *The Journal of Educational Research*, *106*(6), 423–430.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Vogt, B. (2012). *A Methodology to Assess UrbanSim Scenarios* (Master's Thesis). Retrieved from https://calhoun.nps.edu/bitstream/handle/10945/17472/12Sep_Vogt_Brian.pdf

Wixon, M., Baker, R.S.J.d., Gobert, J., Ocumpaugh, J., Bachmann, M. (2012). WTF? Detecting Students who are Conducting Inquiry Without Thinking Fastidiously. *Proceedings of the 20th International Conference on User Modeling, Adaptation and Personalization (UMAP 2012)*, 286-298, Montreal, Canada.

Woolf, B. & McDonald, D. (1984). Context-dependent transitions in tutoring discourse. *Proceedings of the Fourth National Conference on Artificial Intelligence (AAAI),* pp. 355–361, Austin, TX.

ABOUT THE AUTHORS

Dr. Jonathan Rowe *is a Research Scientist in the Center for Educational Informatics at North Carolina State University. He received the Ph.D. and M.S. degrees in Computer Science from North Carolina State University, and the B.S. degree in Computer Science from Lafayette College. His research is in the areas of artificial intelligence and human-computer interaction for advanced learning technologies, with an emphasis on game-based learning environments, intelligent tutoring systems, user modeling, educational data mining, and computational models of interactive narrative.*

Ms. Megan Frankosky *is a Research Psychologist in the Center for Educational Informatics at North Carolina State University. She received the M.S. degree in Human Factors Psychology and Applied Cognition from North Carolina State University. Her research centers on understanding how knowledge of student interactions with technology and corresponding cognitive and affective states can be applied to facilitate the learning process. She is particularly interested in game-based learning, adaptive learning technologies, user experience research, human-computer interaction, user modeling, and learning analytics.*

Dr. Bradford Mott *is a Senior Research Scientist in the Center for Educational Informatics at North Carolina State University. Prior to joining North Carolina State University, he served as Technical Director at Emergent Game Technologies where he created cross-platform middleware solutions for video game development, including solutions for the PlayStation 3, Wii, and Xbox 360. Dr. Mott received his Ph.D. in Computer Science from North Carolina State University in 2006, where his research focused on intelligent game-based learning environments. His current research interests include computer games, computational models of interactive narrative, and intelligent game-based learning environments.*

Dr. James Lester *is a Distinguished Professor of Computer Science at North Carolina State University, where he is Director of the Center for Educational Informatics. His research centers on transforming education with technology-rich learning environments. With a focus on adaptive learning technologies, his research spans intelligent tutoring systems, game-based learning environments, affective computing, and tutorial dialogue. The adaptive learning environments he and his colleagues develop have been used by thousands of students in K-12 classrooms. He received his Ph.D. in Computer Science from the University of Texas at Austin in 1994. He is a Fellow of the Association for the Advancement of Artificial Intelligence (AAAI).*

Dr. Bob Pokorny *is Director of Education and Training Technology Division at Intelligent Automation, Inc. He earned his Ph.D. in Experimental Psychology at University of Oregon in 1985, and completed a postdoctoral appointment at University of Texas at Austin in Artificial Intelligence. Bob's first position after completing graduate school was at the Air Force Research Laboratory, where he developed methodologies to efficiently create intelligent tutoring systems for a wide variety of Air Force jobs. At Intelligent Automation, Bob has led many cognitive science projects, including adaptive visualization training for equipment maintainers, and an expert system approach for scoring trainee performance in complex simulations.*

Dr. Wilbur Peng *received his B.S. in Electrical Engineering from Cornell University, Ithaca N.Y. in 1992, and received his PhD in Electrical Engineering from the University of Maryland at College Park in 2002. His dissertation research addressed the problem of representing and using expert knowledge the form of graph-based similarity structures. At IAI, Dr. Peng leads research and development projects in several areas, including cluster and distributed computing; autonomous aerial vehicle software control architecture; parallel discrete event* *simulation on multicore distributed computing clusters; network simulation, emulation and modeling; and semantic and ontological modeling methodologies, frameworks and applications.*

Dr. Benjamin Goldberg *is a member of the Learning in Intelligent Tutoring Environments (LITE) Lab at the U.S. Army Research Laboratory's (ARL) Human Research and Engineering Directorate (HRED), Simulation and Training Technology Center (STTC) in Orlando, FL. He has been conducting research in modeling and simulation for the past five years with a focus on adaptive learning and how to leverage artificial intelligence tools and methods for adaptive computer-based instruction. Currently, he is the LITE Lab's lead scientist on instructional strategy research within adaptive training environments. Dr. Goldberg is a Ph.D. graduate from the University of Central Florida in the program of Modeling & Simulation.*

Integrating an Interoperable Competency Model with GIFT using the Experience API (xAPI)

Gregory Goodwin[1], Ashley Medford[2], Gabriele Diaz[2], Jennifer Murphy[3], Maria Twitchell, Mary Jean Blink[4]
[1]U.S. Army Research Laboratory, [2]Problem Solutions, [3]Quantum Improvements Consulting, [4]TutorGen

INTRODUCTION

A robust student model is key to successful intelligent tutoring. Traditionally, student models are specific to the intelligent tutoring system (ITS) in which they reside, and are not reusable across different educational technologies. As the U.S. Army works toward the goal of maintaining persistent representations of an individual learner and integrating Soldier training across multiple systems, student models should be flexible to support their use across multiple training platforms. Work on automating student model generation from systems could provide a means to extend the capability of the Generalized Intelligent Framework for Tutoring (GIFT) and allow an ecosystem of ITSs and tools to create common value.

ITSs are capable of both micro- and macro-adaptation (VanLehn, 2005, 2011). Micro-adaptation refers to the tutor-student interaction occurring in real time as training content is being delivered. It depends heavily on measures of the learner taken at the time of training delivery. Macro-adaptation refers to adaptation that occurs at the level of the lesson or course. An individual's overall learning path is adapted to meet their specific needs. For example, changes to the overall difficulty of the material being presented or even the selection of the most appropriate lessons or courses occurs for that learner. This outer loop adaptation depends more heavily on measures of student proficiencies, aptitudes, and goals.

The representation of student traits, states, knowledge, skill, or ability within an ITS is referred to as a student, or learner, model (Sottilare & Gilbert, 2011;Sottilare, Goldberg, Brawner & Holden, 2012; Sottilare, Holden, Goldberg & Brawner, 2013). There are several types of learner models that can be implemented in an ITS. Measures of student performance within these systems may be specific to the content of the tutoring session or they may be measures that apply across a wide variety of content areas (Goodwin, et al., 2015).

Increasingly, student models are beginning to explore ways to represent and use more than just the cognitive measures of learners. Bloom (1956) developed a well-known framework for describing competencies that includes three components: cognitive, psychomotor, and affective. All three of these components have trait-like and state-like dimensions. Table 1 provides a taxonomy and examples all of these possible components of a learner model.

The learner model in GIFT is currently populated by learner input and actions accumulated during the learning session. Learners can provide answers to questions about their level of motivation, interest, and prior knowledge. Finally, as the learner is presented with the material, GIFT can collect a variety of cognitive, behavioral, and physiological measures from the learner. While this approach to populating the learner model is sufficient for data collections during experiments in which learners may only interact with GIFT during a single session, it would be valuable if these trait-like measures could be stored in a long-term learner model. A robust, persistent learner model could be used and updated by GIFT as well as other training systems. The top left quadrant, the trait-like domain-dependent measures, comprise what we are defining as a competency.

Table 1. Components of the Learner Model.

	Learner Measure Category	Trait-Like (macro-adaptation)	State-Like (micro-adaptation)
Domain Dependent	**Cognitive**	Relevant prior cognitive experience/knowledge/training	Comprehension of concepts presented in the training
	Psychomotor	Relevant prior psychomotor experience or training	Measures of skill improvement
	Affective	Fears, likes, goals, attitudes relevant to the training	Arousal and emotions in response to the training
Domain Independent	**Cognitive**	Intellect/aptitude, memory, meta-cognitive skills	Attention, cognitive workload
	Psychomotor	Physical strength, stamina, sensory acuity	Endurance and fatigue
	Affective	Personality traits, general test anxiety	Arousal, emotions resulting from factors independent of training

One approach to streamlining the data collection process is to create a learner model that is continuously updated as the learner gains experience and receives training. Leveraging existing data about learners to determine relevance and predict probability of success could greatly increase effectiveness of ITSs. One way of doing this is to use the experience application programming interface (xAPI), which provides an interoperable way to describe individuals and their interactions, or experiences, with systems.

Learner Data vs. Competency

It is important to realize that simply recording the training activities, experiences, assignments, accomplishments, etc. of learners is not sufficient to create a learner model with learner competencies. The purpose of a learner model is to have a representation of the learner's knowledge, skills, abilities, and competencies. These constructs are all inferred from more granular raw data of student activities.

Recording this raw performance data is the focus of efforts including specifications like the xAPI and the Human Performance Markup Language (HPML). The ability of these open standard data repositories to support comprehensive learner assessments, as would be needed for a learner model, have been demonstrated in projects like the Soldier Performance Planner (SP^2) or Pipeline. These efforts enable performance data to be captured and interacted with in interoperable ways. In turn, they represent a base of technologies and a user community that is currently using the xAPI to support learner assessments.

Currently, GIFT has a minimally functional capability in its Learning Management System (LMS) module to produce and consume simple xAPI data. A minimally functional macro-adaptive course filtering capability also exists. In recent research, we aimed to expand the creation and use of xAPI data by the GIFT architecture. While GIFT had an initial xAPI capability, we proposed a more focused and granular approach that would allow additional affordances in a learning ecosystem. This effort endeavored to conduct the foundational research to enhance the portability of persistent competency and student models in order to support their use across multiple training systems and ultimately, enable their application for more granular predictors of success within ITSs. To accomplish this goal, we worked to enhance the GIFT architecture to import and track externally generated competency models as well as to expand its creation and use of xAPI data. The integration of xAPI data into GIFT provided the potential support cross-platform student modeling for Army training and education. Further, it enabled the investigation of relevant research questions about how these complex learner models should best be leveraged.

In this paper, we describe our work to create such a learner model which we call an Interoperable Competency Model (ICM) that can be used by GIFT. This model incorporates xAPI performance data from training delivered both within and outside of GIFT, to model learner competencies. An ICM database exists outside of GIFT to facilitate the data being accessed, updated, and used by multiple training systems. In this way, GIFT can effectively cooperate with other training systems within a learning ecosystem to develop learner competencies. This might take the form of providing programmed or remedial training, providing diagnosis of learner problems, recommending training to learners and instructors, or providing predictions of skill acquisition and/or retention.

The use case selected for this project involves the use of marksmanship training data. This use case is well suited for this project because marksmanship training in the Army is currently conducted in a series of training events that include simulation, classroom, and a variety of live-fire events. Learner data are available at different levels of granularity across these different events. While these data could be captured across systems using the xAPI, the majority of marksmanship training technologies do not currently incorporate the xAPI. To collect these data, we integrated this research with ongoing efforts to develop interoperable metrics for marksmanship performance using a number of training technology systems. Under the Support for Training Effectiveness Assessment with Data Interoperability (STEADI) effort, performance metrics expressed in xAPI statements were developed for training effectiveness evaluation and to meet the needs of other audiences. Marksmanship data were collected using a subset of technology-based training systems. Integration of these research efforts provided data required to develop an ICM for marksmanship, enabling integration with GIFT as well as additional levels of analysis and experimentation. Although technologies do not currently exist to capture all marksmanship performance data in a single learner record store (LRS), for this use case we constructed a hypothetical database to demonstrate how such an interoperable competency model could work to develop interoperable metrics for marksmanship performance using a number of training technology systems.

METHODS AND RESULTS

Model Development

Trait-like basic rifle marksmanship assessments were identified and organized into three categories of measures: cognitive, psychomotor, and affective. In addition to these measures, our team identified measures of marksmanship performance that could serve as both state-like measures as well as outcome measures that could serve to validate this model going forward. These include qualification scores, range system data, sensor data, shot analysis, and guided instructor assessments. These measures were based on previous research involving marksmanship training (James & Dyer, 2011).

Table 2. Rifle Marksmanship Assessment Constructs

Component	***Construct***
Cognitive	General Cognitive Ability Marksmanship Domain Knowledge Openness to Experience Hunting Experience Videogame Experience
Psychomotor/Physical	Visual Acuity Handedness Eye Dominance Height Physical Fitness Sports Experience Musical Ability
Affective	Perceived Stress Resiliency/Hardiness Grit Self-Efficacy Initiative

In order to maximize plasticity and minimize reengineering requirements, an online xAPI registry was created to include the identified marksmanship measures. The online registry provides a central means to quickly update the core measures and metrics during future research efforts. Table 3 contains examples of the data fields found in the xAPI marksmanship registry.

Table 4. Sample Data Fields Identified in the xAPI Marksmanship Registry

Measure	**Description**	**Data Type**	**Data Range**	**Source**	**xAPI Property**
Team Sport Experience	Team sport participation frequency	Integer	1-5; not at all to very frequently	Demographics Questionnaire	Context
Eye Side Dominant	Eye dominance: left or right	String	(a) left eye, or (b) right eye	Demographics Questionnaire	Context
BRMT	Basic Rifle Marksmanship Test score	Integer	0 to 10	Basic rifle marksmanship testing score	Context

The xAPI registry was developed in a webpage to increase accessibility. The online registry was created using Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and JavaScript. The layout of the page was designated by CSS, while the actual information within the page, including the primary table with the marksmanship measures, was made using HTML. JavaScript was leveraged to enable particular behaviors for the site. Specifically, adding a script allowed the ability to click the “Example” button for a specific measure and seamlessly jump to the appropriate location to view the sample xAPI statement for that measure. Once clicked, the example was designed to appear at the top of the webpage. Figure 1 illustrates the layout of this website.

Team Sport Experience	Team sport participation frequency.	http://www.problemsolutions.net/xapi/contextextensions/team-sport-experience	Context	Example
Individual Sport Experience	Individual sport participation frequency.	http://www.problemsolutions.net/xapi/contextextensions/individual-sport-experience	Context	Example
Large Animal Hunting	Large game hunting frequency.	http://www.problemsolutions.net/xapi/contextextensions/large-animal-hunting	Context	Example
Small Animal Hunting	Small game hunting frequency.	http://www.problemsolutions.net/xapi/contextextensions/small-animal-hunting	Context	Example
Fishing	Fishing frequency.	http://www.problemsolutions.net/xapi/contextextensions/fishing	Context	Example
Outdoor Activities	Time spent outside frequency.	http://www.problemsolutions.net/xapi/contextextensions/outdoor-activities	Context	Example
Media Consumption	How much media is used on a regular basis.	http://www.problemsolutions.net/xapi/contextextensions/media-consumption	Context	Example

Figure 22. xAPI Marksmanship Registry Website

Competency Modeling

The marksmanship assessment constructs, encoded in the xAPI, enabled the collection of training data to develop an initial ICM. Interoperable competency models map these learner data to their respective assessment constructs within the model. In turn, the models are able to make individualized predictions about the learner's marksmanship competency.

For example, the Army scores marksmanship competency/proficiency in four categories:

1. Expert (38-40 hits; max = 40)
2. Sharpshooter (33-37 hits)
3. Marksman (26-32 hits)
4. Unqualified (25 or fewer hits)

The ICM could use learner measures to make predictions about competency using the Army standard. However, in a training environment, it would probably be more useful to be able to use learner measures to make predictions about performance in intermediate training events. For instance, an ICM might map performance in the simulator to predictions about performance during the subsequent period of live instruction.

The values in the model each have the following attributes:

1. Default (if no data for this factor)
2. Weight (relative contribution of the factor to the prediction)
3. Min (minimum value possible for this factor)
4. Max (maximum value possible for this factor).

As an example, the algorithm for translating these data into a prediction for a qualification score [0, 40] would be the following:

Weight x [(Score – min) / (Max – min)] = f.c. (factor contribution)

Alternatively, an ICM could be used to determine the types of mistakes students are making, which would enable GIFT or other systems to determine the types of interventions needed. Such an ICM might focus more on the variance in the recorded shots, or even the pattern in the shot groupings themselves in addition to trait-like measures. Figure 2 illustrates how these measures could be used to identify different training strategies.

Case 1	Case 2
Low written test score Type A shot pattern Low previous experience High aptitude Required intervention: start with the basics, but allow self-paced learning	Low written test score Type A shot pattern Low previous experience Low aptitude Required intervention: start with the basics, but require directed learning

Figure 2. Use of Learner Measures to Predict Training Strategies

GIFT Enhancements

It is important to keep in mind that the purpose of an ICM is to facilitate the sharing of learner competencies in a complex learning ecosystem. Students will come to GIFT having received training and experience outside of the GIFT environment. The following illustrates how GIFT would use the ICM to identify the learner's competencies.

An external, RESTful web service connected to a database contains ICM measures. The server receives specific ICM requests from GIFT when a user chooses a course. These requests are used to query its database for matching ICMs. The ICM RESTful server then sends a response to GIFT with the appropriate ICM, if available. This process is illustrated in Figure 3.

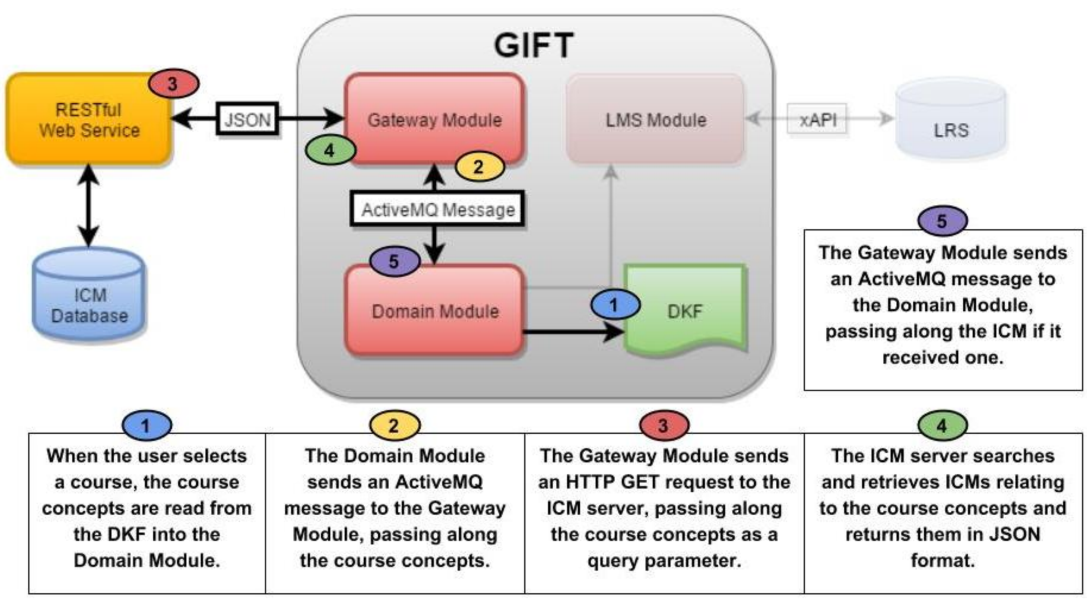


Figure 3. GIFT Integrated Architecture Flow: Steps 1 - 5

Once GIFT has the ICM, it uses that to query the learner data stored in the LRS and then uses the ICM to generate the appropriate assessment of the learner's competency. GIFT can then use that assessment to make macro-adaptive decisions about training.

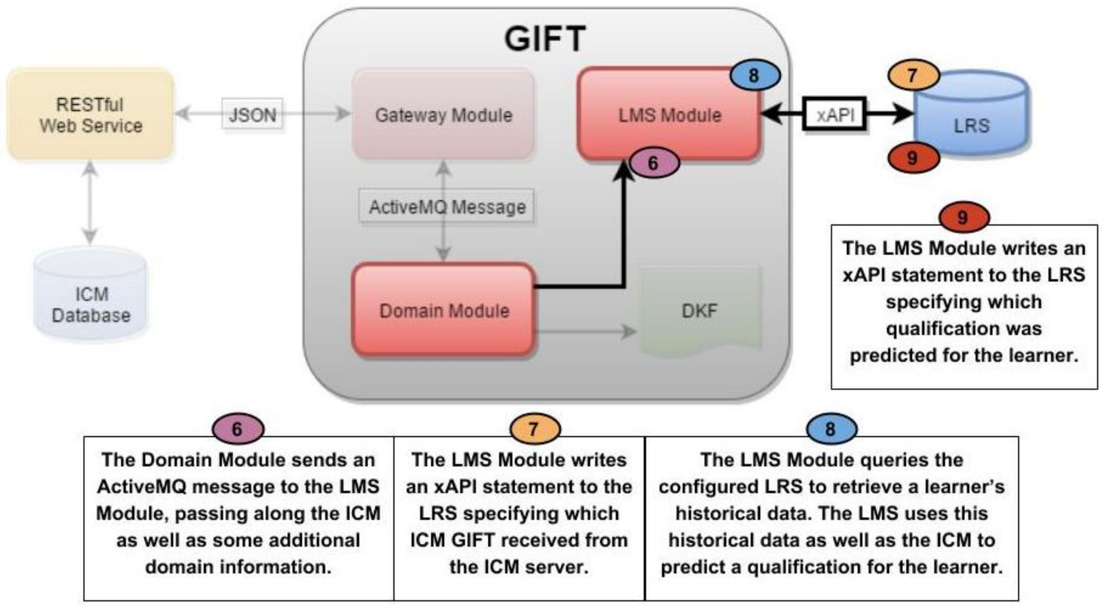


Figure 4. GIFT Integrated Architecture Flow: Steps 6 – 9

As the learner completes training in GIFT, those learner behaviors and assessments would be fed back into the LRS. That new performance data would then impact subsequent assessments of learner competencies. This is illustrated in Figure 5.


```
{
    "verb": {
        "id": "http://activitystrea.ms/schema/1.0/qualify"
    },
    "timestamp": "2015-12-10T15:39:17.413702Z",
    "object": {
        "definition": {
            "name": {
                "en-US": "Marksmanship Example Course"
            },
            "description": {
                "en-US": "Marksmanship Example Course"
            }
        },
        "id": "activityId:GIFT/Marksmanship+Example+Course%5CmarksmanshipCourse.course.xml",
        "objectType": "Activity"
    },
    "actor": {
        "mbox": "mailto:jsavage@problemsolutions.net",
        "objectType": "Agent"
    },
    "stored": "2015-12-10T15:39:17.413702Z",
    "result": {
        "response": "expert"
    },
```


Figure 5. Example of xAPI Statement Written to LRS by GIFT at Step 9

In sum, our research resulted in multiple enhancements to GIFT. An external RESTful web service, connected to a database, was developed to store ICMs. A prototype marksmanship ICM was integrated, and GIFT was configured to communicate with the ICM web service via Java-Script Object Notation (JSON). In addition, GIFT was enhanced to request and process external ICMs, using the marksmanship model and historical learner data captured with the xAPI, in order to produce a prediction of an individual learner's success or failure.

GIFT's xAPI functionality was augmented in order to allow the consumption and generation of more granular xAPI statements. As a result, GIFT is able to process and track ICMs via xAPI. Moreover, GIFT is afforded with focused/attenuated consumer functionality capable of filtering, sequencing, selecting, and interfacing external content and systems. Conversely, GIFT is now able to produce xAPI data usable by outside systems within a learning ecosystem.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

This work represents an important step towards expanding the ability of GIFT to function in a complex learning ecosystem. Individuals will always learn and gain experience in a variety of places and interoperable competency models will facilitate the efficiency and effectiveness of adaptive training systems like GIFT in such an environment. Although this is an important step, it is only the first towards demonstrating the value of this capability.

Lacking actual marksmanship data, we developed this ICM based on notional data that were stored in an LRS. In future work, we plan to use actual data from students going through the Army's rifle marksmanship training course. This will enable us to validate our model and refine our measures. By collecting these data, it will be possible to determine which performance measures play a significant role in identifying competency levels in students and which ones do not. Identifying these high value measures will improve the efficiency of the assessment process.

In addition to determining which measures are most predictive of competency levels, we will want to insure the complete ICM accounts for enough variance to reliably predict competency levels. Addressing this will require both an examination of the constructs we have measured as well as the level of granularity of the xAPI representations of those measures. This will be especially true of sensor data. Raw sensor data can be voluminous and are not suited for storage in an LRS. The xAPI statements in an LRS tend to be discrete statements that reflect critical features of the raw data (e.g., min, max values, area under the curve, etc.). Making sure that the right features are included as statements in the LRS so that learner competencies can be derived will require some experimentation.

As we refine and validate these xAPI statements, we will also need to improve and expand the xAPI marksmanship registry. This registry will serve as a valuable resource for the community. As new training aids, devices, simulators, etc., are developed, the registry will provide a means for standardizing xAPI statements in the marksmanship domain insuring interoperability across those systems and with existing ICMs.

REFERENCES

Bloom, B.S. (Ed.). Engelhart, M.D., Furst, E.J., Hill, W.H., Krathwohl, D.R. (1956). Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain. New York: David McKay Co Inc.

Goodwin, G., Johnston, J., Sottilare, R., Brawner, K., Sinatra, A., & Graesser, A. (2015). Individual learner and team modeling for adaptive training and education in support of the U.S. Army Learning Model: Research Outline. US Army Research Laboratory Special Report 0336. Aberdeen Proving Ground, MD.

James, D. R., & Dyer, J.L. (2011). Rifle marksmanship diagnostic and training guide. U.S. Army Research Institute for the Behavioral and Social Sciences. Research Product 2011-07. Arlington, VA.

Sottilare, R., & Gilbert, S. (2011). Considerations for tutoring, cognitive modeling, authoring and interaction design in serious games. Authoring Simulation and Game-based Intelligent Tutoring workshop at the Artificial Intelligence in Education Conference (AIED) 2011, Auckland, New Zealand, June 2011.

Sottilare, R., Goldberg, B., Brawner, K., & Holden, H. (2012). A modular framework to support the authoring and assessment of adaptive computer-based tutoring systems (CBTS). In Proceedings of the *Interservice/Industry Training Simulation & Education Conference*, Orlando, Florida, December 2012.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Sottilare, R., Holden, H., Goldberg, B., & Brawner, K. (2013). The Generalized Intelligent Framework for Tutoring (GIFT). In Best, C., Galanis, G., Kerry, J. and Sottilare, R. (Eds.) *Fundamental Issues in Defence Simulation & Training*. Ashgate Publishing.

VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., & Taylor, L., et al., (2005). The Andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence and Education*, 15(3), 147–204.

VanLehn, K. (2011): The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems, Educational Psychologist, 46:4, 197-221

ABOUT THE AUTHORS

Dr. Gregory Goodwin *is a senior research scientist at the Army Research Laboratory-Human Research and Engineering Directorate, Advanced Training and Simulation Division in Orlando, Florida. His research focuses on methods and tools to maximize the effectiveness of training technologies. After completing his Ph.D. at the State University of New York at Binghamton in 1994, Dr. Goodwin spent four years in post-doctoral training at the Columbia University College of Physicians and Surgeons, and at Duke University Medical Center. After working as a college professor, he took a position with the Army Research Institute (ARI) in 2005 and in 2014 he joined the Army Research Laboratory.*

Ms. Ashley Medford *is the Director of Government Services at Problem Solutions. She is an experienced program manager and consultant, specializing in the development and use of immersive technologies for education, training, analysis, and collaboration. Throughout her career, Ms. Medford has led various immersive learning technology initiatives for major clients, including the Air Force Research Laboratory, Naval Air Systems Command, Army Research Laboratory, National Aeronautics and Space Administration, Office of Naval Research, U.S. Special Operations Command, and U.S. Central Command. She is also a member of the e-Learning Guild, Association for Talent Development (ATD), and National Defense and Industrial Association (NDIA).*

Mr. Gabriel Diaz *is the Deputy Director of Government Services at Problem Solutions. He has 20 years of military experience in the Marine Corps. As an active duty officer he executed duties as a communications and data network planner and operator. He was the Modeling and Simulation Officer to support simulation technology in support of live experimentation for the Marine Corps Warfighting Lab. He holds a BBA in Management Information Systems from the University of New Mexico, and a MS in Modeling & Simulation from the Naval Postgraduate School. Gabriel supports I/ITSEC on the Human Systems Performance and Training committees.*

Dr. Jennifer Murphy *is the Chief Executive Officer of Quantum Improvements Consulting, LLC. She has over 11 years of military selection and training research experience. Dr. Murphy holds a Ph.D. in Cognitive/Experimental Psychology from the University of Georgia. She previously worked as Research Psychologist at the U.S. Army Research Institute for the Behavioral and Social Sciences (ARI). Her research focused on the development of technology-based selection and training measures for cognitive and perceptual skills. Dr. Murphy served as*

Director of Defense Solutions at Design Interactive, Inc., where she managed a portfolio of training and performance support efforts for defense clients.

Ms. Maria Twitchell *conducts analyses, design, and evaluation of training solutions. Mrs. Twitchell has designed and conducted methodologies for job analyses, behavioral and cognitive task analyses, formative and summative evaluations for various training and electronic job aids, and validated various training performance support systems. Maria has also designed and developed evaluative components for electronic performance support tools, online training assessment simulations, and training support packages. She has developed and validated knowledge-based tests, performance–based assessments, and various types of reaction surveys. Her experience includes consulting for small organizations and nonprofits on business developmental interventions pertaining to selection, performance, certifications, and training.*

Ms. Mary Jean Blink *has extensive experience designing custom and commercial software products. She has computer audit experience through Ernst & Young in the high technology and biotechnology industries. Her previous experience includes hardware, software, and development; network infrastructure; technology policy development; compliance documentation in highly regulated industries; and database administration. She continues as a respected authority in software development and is often invited to speak at software industry conferences and developer meetings. Ms. Blink has higher education teaching and administration experience, and is currently serving as COO for TutorGen, Inc., a Carnegie Mellon startup in the education technology field.*

An Adaptive AAR Capability for GIFT

Alan Carlin[1], Diane Kramer[1], Chris Nucci[1], Evan Oster[1], Jared Freeman[1], Keith Brawner[2]
[1]Aptima, Inc., [2]Army Research Laboratory[2]

INTRODUCTION

The Army Learning Concept (ALC) put forth the Army Learning Model (ALM), which embraces a learner-centric learning environment and requires learning to be tailored to the individual's experience and competence level (U.S. Department of the Army, 2011, p. 31). The ALM focuses on a Soldier-centered approach to learning and encourages learning to move beyond the classroom to a blended, more individualized model that incorporates the three pillars of leader development: institutional instruction, self-development, and operational experience. In such an environment, learning experiences are tailored to individual needs, creating more engaging and effective training.

To fully realize the intent of the ALM, technology must be developed that can effectively supplement and enhance instructors in classroom settings, while providing the necessary learning and feedback experiences during self-guided learning. One method for providing instruction that is tailored to the individual is the use of intelligent tutoring systems (ITSs). One attribute of these systems is that they customize the content, form, and timing of feedback to the learner. Typically, ITSs offer feedback either incrementally, as a learner is completing steps within a task (typically called the inner loop), or upon the learner's completion of a task (typically called the outer loop; VanLehn, 2006). In the context of the inner loop, feedback may be delivered in many forms; examples include a hint, an assessment of a correct or incorrect response (e.g., error flagging; Corbett & Anderson, 1991), an explanation of why a response is considered incorrect, a prompt of the student for an answer, suggestion for self-reflection, or other items. The outer loop approach, in contrast, uses the assessment of the learner's state to recommend future training content. Implementations of inner loop feedback can fail in a number of ways (c.f., Sottilare, Goldberg, Brawner, & Holden, 2012). For example, feedback concerning the accuracy of task performance may be insufficient to enable the learner to recognize the root cause of failure, to learn an alternative response, and to enact it reliably in the future (Higgins, Hartley, & Skelton, 2002). In addition, the ITS may be unreliable in its selection and delivery of feedback to ensure future learning and improved learner performance. It has traditionally been a costly and slow business to develop ITSs that avoid these shortcomings with respect to feedback, as there are long cycle times between development, implementation, and effect measurement.

The Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden, 2012; Sottilare, et al, 2012, p. 3) is intended to make the development and use of ITSs efficient. Thus, GIFT should help deliver the considerable benefits of ITSs (VanLehn, 2011) to a wider audience. The power of intelligent tutoring is, in large measure, due to automated delivery of timely, learner-specific feedback (c.f., Schooler & Anderson, 1990). Such tailoring intends to help optimize learning and further move the Army toward a Soldier-centered approach to learning, mentioned in requirements documentation.

The US Army Research Laboratory (ARL) seeks to improve the capability for customized feedback within GIFT, in support of the ALM vision. In this paper, we report on progress in developing an Adaptive After-Action Review (AAR) Module that is grounded in the well-validated theory of deliberate practice, which suggests that expertise grows best when the learner focuses study and practice on specific, deficient knowledge and skills, and receives feedback concerning the efficacy of their performance (Ericsson, Krampe, & Tesch-Romer, 1993). The AAR model will improve the learner's competence on specific competencies. It is based on a domain-general mathematical approach, and contains adjustable parameters that will be adjusted by looking across instances in which feedback has been applied and assessing the impact of that feedback on learning. This model will address both components of the deliberate practice formula, as it will focus study on specific competencies and optimize feedback that drives learning and improves performance. The Module will plug into the GIFT architecture, making it simpler for ITS developers to dynamically optimize feedback for users of GIFT-based ITSs.

The Adaptive AAR Module is designed to support the delivery of AAR during self-guided learning and in classroom instruction. Initially, we are developing our technology to interact with Newton's Playground (Zhao, W., M. Ventura, et al., 2015) in the physics tutoring domain. The capability within will be generic, and in future work will be expanded to other domains.

Overview of Approach

Our approach is grounded in the theory of deliberate practice (Ericsson, et al, 1993). The result is a Module that dynamically adjusts feedback using a domain-general mathematical approach, and will plug into the GIFT architecture. The feedback is based on several learner-specific factors, including the individual learner's training objectives, performance against those training objectives, competency level on independent or interdependent competencies, and the priority of the competency with respect to the individual's role or mission. We report on a prototype that selects the feedback, given current learner state. The selected feedback can then be provided through the training environment to the learner.

The feedback is generated by policy consisting of a set of rules. The policy can also potentially include rules for selecting future training content, but this work focuses on the feedback. These rules can either be specified by an instructor manually, or else automatically generated by Educational Data Mining (EDM) software that interprets historical data, generates a model of feedback effectiveness, and in turn uses that model to generate the feedback rules. In this paper, we describe this EDM software.

In **Figure 23**, we show a workflow consisting of the following steps. The steps form a cycle, and after the last step, the first step is repeated.

- Step1 (Skill Update): The Adaptive AAR Module interprets the learner's performance history and assesses the learner's progress based on the history;
- Step 2a (Selection): The Policy Component selects AAR materials to show the assessment to both the learner and the instructor
  - Step 2.5a (Optional: Instructor-In-the Loop): The instructor optionally accepts, rejects, or adds to the assessment;
- Step 2b (Training Selection): The policy simultaneously provides the assessment information to an MDP module that selects training content
  - Step 2.5b: Content selection modules use this information to help select content. The instructor optionally accepts, rejects, or adds to the recommended content;
- Step 3 (Learner performance): The learner performs the selected training activity; and

Step 4 (Data Processing and Storage): Data from the learner performance are acquired, and Step 1 is repeated.

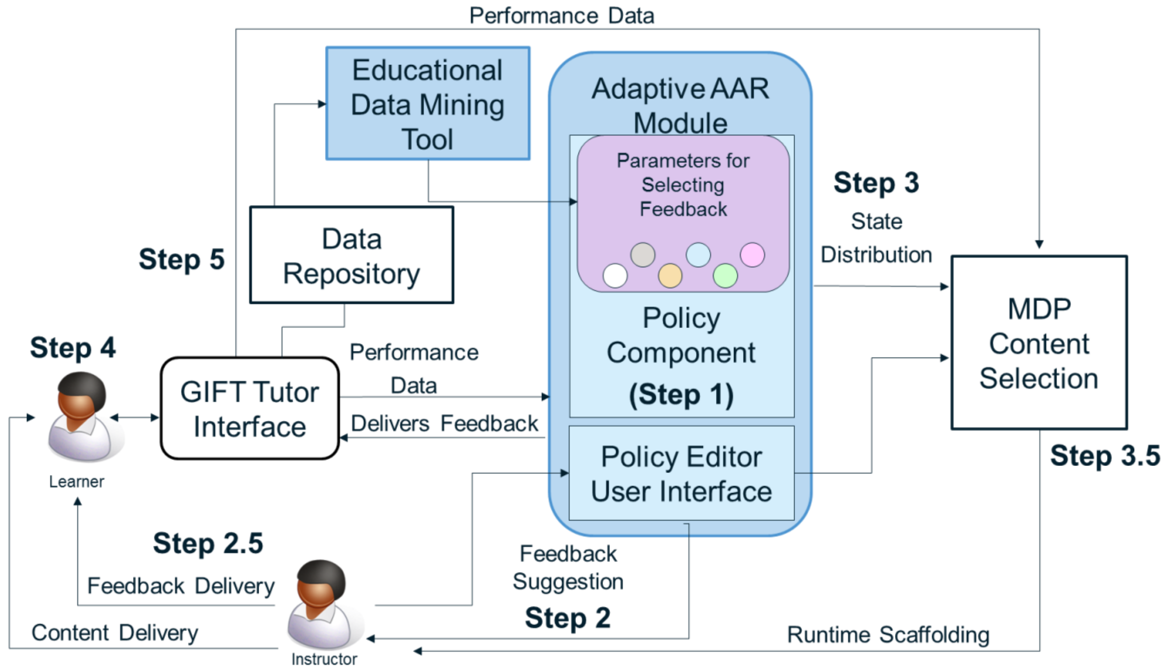


Figure 23: Approach to dynamically parameterizing feedback for AAR.

To accomplish adaptive selection and presentation of individualized AAR feedback, and to support training content selection, the Adaptive AAR Module enables the integration of a mathematical model with the existing GIFT architecture. Figure 1 shows the specific components that we are developing, in blue. An Adaptive AAR Module assesses learner competency throughout runtime. It is used to deliver feedback to the learner, and to modify its assessment and its feedback as the learner performs. The Adaptive AAR Module can be used to provide assessment information to other GIFT algorithms that support training content selection (e.g., Rowe 2013), shown in the box labeled "MDP Content Selection." The Adaptive AAR Module executes a training policy; this policy contains parameters that allow it to be configured. To help set the parameters correctly, a second module analyzes historical performance data to (a) analyze which measures best correlated with learner performance to better inform AAR feedback, and (b) to analyze which feedback and training worked well to enhance AAR feedback selection and provide the MDP Content Selection module with better predictive information that it can use.

As mentioned, Figure 1 has two items in blue, implanted and reported on as part of this work:

- An **EDM tool** that inputs learner performance data and outputs a training model. The training model is used to construct and execute the adaptive policy component. In this way, the resulting AAR is empirically driven.

- An **adaptive policy component** that assesses learner progress and provides AAR. It provides this assessment to the instructor (in the form of a detailed report on learner strengths and weaknesses) and also to the learner (in the form of a similar report, and also advice on how to address the strengths and weaknesses). The policy component is also, optionally, capable of providing the assessment to Markov decision process (MDP)-based instructional scaffolding software. Any

MDP-instructional software could then use the assessment information to plan over the several possible states of the learner, rather than just for one learner state.

In the next two subsections, we describe these two parts of the approach in turn.

Educational Data Mining Tool

The EDM tool inputs learner performance data and outputs a training model. The input required need only conform to a minimal ontology (Table 1) consisting of learner ID, training item, and at least one measure. At least one of these measures must be specified as a "training goal". An example of a "measure" is incorrect GIFT report of At/Below Expectation. An example of a system training objective is to guide learners toward "At Expectation" on all associated concepts.

Table 5. Input to EDM module

Learner ID	Sequence Number	Exercise / CourseID	Measure 1	Measure 2	Learner Competency Level	Scenario Difficulty Level	Competencies addressed in Scenario
123	1	301A	11	.5m	θ_{123}^{1}	$d_{\{301A\}}$	$a_1^{301A}, a_2^{301A} \ldots$
123	2	301A	0	.7m	θ_{123}^{2}	$d_{\{301A\}}$	$a_1^{301A}, a_2^{301A} \ldots$
123	3	301B	5	.5m	θ_{123}^{3}	$d_{\{301B\}}$	$a_1^{301B}, a_2^{301B} \ldots$
124	1	301A	2	.7m	θ_{124}^{1}	$d_{\{301A\}}$	$a_1^{301A}, a_2^{301A} \ldots$
124	2	301A	6	.5m	θ_{124}^{2}	$d_{\{301A\}}$	$a_1^{301A}, a_2^{301A} \ldots$
252	1	456A	2	.7m	θ_{252}^{1}	$d_{\{456A\}}$	$a_1^{456A}, a_2^{456A} \ldots$
252	2	301A	9	.5m	θ_{252}^{2}	$d_{\{301A\}}$	$a_1^{301A}, a_2^{301A} \ldots$

The values of the variables in the last three columns of data are unknown, but can be inferred by computation. The EDM Tool fills in the missing variables (represented in the last three columns) using Gibbs sampling (Geman & Geman, 1984).

The filled in variables are used to construct any model which contains the following components:

- States: The competencies assessed in the scenario correspond to the partially observable Markov decision process (POMDP) state space.
- Actions: The Exercise IDs correspond to POMDP actions.
- Transitions: The goal is to find P(s',s,a), the probability that a learner advances to state s' given that the learner was in state s and trained on exercise a.
    - This is best illustrated by example. Suppose the domain involves a single competency, and the Educational Data Mining Tool assesses Learner 123 as a Novice in Row 1 and Learner 124 as a Novice in Row 4. Suppose Learner 123 advances to Intermediate in Row 2, but remains as Novice in Row 5. Then, the transition effect of Exercise 301A on a learner at Novice level would then be computed as 50%.
- Observations: The goal is to find O(o | s'), the probability that a measure o is received given that the true learner state is s'. Similar to transitions above, once Table 1 is completely filled out, this can be inferred by counting the data.
- Rewards: Rewards are largely up to the user. In general, the system assigns rewards for achieving high levels of competency.

The approach is based on a generative model, given performance data. Each item i is assigned a variable corresponding to its difficulty, written d_i. Each $d_i$is only approximately known at the outset, but note that if we knew which competencies the item tests, and if we also knew the competency level of all learners in the performance data on those competencies, we could make a reasonable estimation as to the difficulty level using an extension to Item Response Theory (IRT) (Lord, 1980), by finding the probability distribution of d_i when fitting the learner data for all the learners to the equation

$$pr(correct) = \frac{1}{1+e^{\sum_i a_i(\theta_i - d_i)}}, \quad (1)$$

(where $\theta_i$is the learner level of competence) and sampling from this probability distribution to come up with a reasonable value for d_i. Each item is also assigned a set of competencies that it tests, and we store this in a vector variable a_i and adjust Eq. 1 so that the exponential term is a dot product of $a_i(\theta_i - d_i)$. a_i is unknown at the outset, but note that if we knew the difficulty of all the items and the competency levels of all the learners in the data, we could then make a good guess as to the competency assignments of each item. Finally, each learner is assigned a competency level on each of the competencies in vector θ_i is unknown at the outset, but note that if all the item difficulties were known, and all the competencies that the items tests were known, then we could use IRT to make a reasonable estimation as to learner competency level. In summary, we have several unknown variables, and we wish to find the most likely value for each of them that explains the data. There are several other variables that are solved for as well, such as the probability that the lesson increases learner competency. The EDM tool uses Gibbs sampling to learn the value of all the unknown variables at once, and this approach could be expanded to Markov Chain Monte Carlo (MCMC) if it becomes necessary.

ADAPTIVE POLICY COMPONENT

An adaptive policy generator inputs the training model described earlier and outputs an adaptive training policy, as shown in Table 6. The structure of an adaptive policy is best illustrated with an example, shown in the table. At any given time, the policy is in a "node". A node is best thought of as a state in a state machine, with a pointer maintained to one of the nodes, which is the current node. Suppose we start with the current node at Node 1. The table says that at Node 1, we select Scenario 991. Suppose measures are all on a 0 to 10 scale. Then we separate into three cases. If the student produces Measure 1 between 0 and 2, we print "advice here" and remain in Node 1, which will dispense Scenario 991 again. If the measure is between 2 and 4, we print "not quite" and proceed to Node 2, which produces Scenario 999. If the score is between 4 and 10, we print "good score!" and proceed to Node 3.

Table 6. Adaptive policy

Node	Scenario ID	M1 lower bound	M1 upper bound	M2 lower bound	M2 upper bound	M3 lower bound	M3 upper bound	Next node	AAR
1	Scenario 991	0	2	0	10	0	10	1	Print 'advice here'
1	Scenario 991	2	4	0	10	0	10	2	print 'not quite!'
1	Scenario 991	4	10	0	10	0	10	3	print 'good score!'
2	Scenario 999	0	10	0	10	0	10	3	

Each node is a state of execution for the program (not to be confused with learner state). At any point in time, the policy is associated with the current node, which can span several rows. When measures are received, the policy looks up which range (i.e., which row) the measures correspond to. It then issues the AAR in that row. It then advances to the node in the "Next Node" column.

For a policy to be legal, it must follow these rules:

- Rule 1: all rows containing the same node (e.g., in this example, the first three rows contain node 1) must contain the same Scenario ID. The policy in Table 1 is legal, because all nodes containing Node 1 identify Scenario 991 as the executed scenario.
- Rule 2: The measures collectively cover all the possible cases *for the scenario identified in the row*. Otherwise, the policy does not know how to proceed for the uncovered case.

INTEGRATION INTO GIFT DOMAIN MODULE

In this section, we describe how a policy gets executed and AAR gets selected in GIFT. To do this, we build on work from the Integrated Performance Assessment (IPA) program (Hruska et al., 2011). Figure 2 shows the context of the work within GIFT. To implement, we implemented software in the domain module which interacts with the LMS and Courses to receive information, and will interface with the Tutor User Interface (TUI) to display AAR.

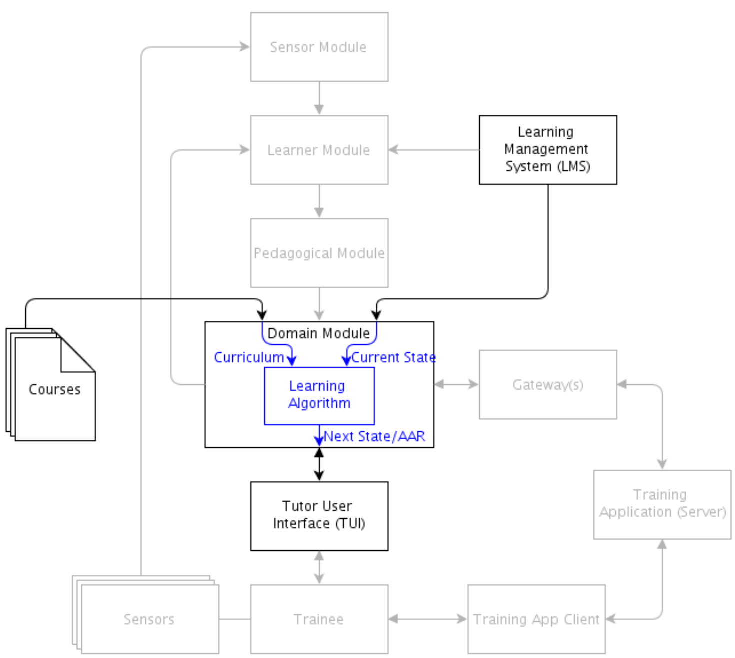


Figure 24: Modified domain module software and its interfaces. We modify the domain module in GIFT to include a learning algorithm, which outputs an AAR (and optionally the current state or node).

Below, we describe the framework further, in context of the workflow during training. With the exception of the initialization step, the steps below correspond to the steps in Figure 1, earlier in this paper.

Domain Module Initialization Step

For the purpose of efficiency, when the Domain Module is started and refreshes its internal course list, we allow it to generate a "curriculum" object, shown in Table 3. This object will contain the list of courses, the skills associated with those courses (likely compiled from the tasks within the scenarios), and the skill levels (likely BelowExpectation, AtExpectation, and AboveExpectation).

Table 7: Curriculum object

Property	Type	Description
Courses	Course[]	The available courses in the curriculum
Skills	Skill[]	A summary of skills used by courses in the curriculum

Domain Module Course Selection Step

There are two cases for course selection. One is when the student history is known, and an entry for the student exists for in an LMS, and the history is populated by one or more courses. The second case, is when no student history exists, and the student is completely unknown. We call this latter case an Initial State.

Initial State: If training records are available, we may wish to recommend the simplest course from the curriculum as determined by the Training Algorithm. (Go to the Skills Update step).

Post-scenario: When the domain options are presented to the user, the Domain Module requests course history and assessments from the LMS Module in order to make simple recommendations. At this point, the recommended course from the training policy component will also be placed as recommended. Any After Action Review information should be displayed as the reason for recommendation.

Skills Update Step

When a scenario is completed, the Domain Module assesses the learner and generates the score tree, which is sent to the LMS Module for storage. This data will also be passed to the training policy component along with the curriculum and learner's previously known state.

Training Selection Step (via Training Policy)

Using the scores, the training policy component determines the learner's next state from the previous state within the curriculum. As discussed above (see "Adaptive Policy"), we generalize the format of a training policy, so that it can support policies generated via a single MDP, multiple MDPs, a POMDP, or other policy generation algorithms. The software interface of the training policy consists of several structures.

Skill: A skill is a something that can be trained by one or more courses.

Table 8: Skill definition

Property	Type	Description	Example
Name	String	The unique name of the skill	*canDrawNeededAgents*
Levels	String[]	The skill levels associated with the skill in order of increasing proficiency	*{BelowExpectation, AtExpectation, AboveExpectation}*

Course: A GIFT course has a name, and an array of associated skills trained.

Table 9: Course definition

Property	Type	Description	Example
Name	String	The unique name of the course	*NewtonianTalk (Playground 2, Puzzle 1)*
Skills	CourseSkill[]	The skills trained within the course	*Impulse Learning Objective*

Course Skill: A course skill is a skill as it exists within a course. The skill has an applicability to the course, which specifies how relevant the skill is to the course. The more applicable the skill, the more the course will measure that skill. The difficulty is the difficulty of the course with respect to the skill. Course difficulty is weighed into assessments, e..g., if a student passes a very difficult course, this impacts assessment more than if the student passes an easy course.

Table 10: Course skill definition

Property	Type	Description	Example
Name	String	The name of the skill	*canDrawNeededAgents*
Applicability	Decimal	The applicability of the skill to the course as a percentage (0.0 to 1.0)	*0.80*
Difficulty	Decimal	The difficulty of the skill within the course as a percentage (0.0 to 1.0)	*0.25*

Curriculum: A curriculum is a collection of all available courses and associated skills.

Table 11: Curriculum definition

Property	Type	Description
Courses	Course[]	The available courses in the curriculum
Skills	Skill[]	A summary of skills used by courses in the curriculum

After Action Review Step

Each Node in Table 1 is associated with an After Action Review (AAR) in the right column. The intuition behind this mapping is that each Node is reached based on the trainee's history of courses taken, and measurements received. Thus, if the policy generation is reasonable, then it should be true that each node maps to a student with a particular set of competences as known by that student's history of courses taken and measurements received. Note that the node current node within the policy depends most on the most recent measurement from the most recent course! Therefore, each node in the policy maps to an After

Action Review. An AAR, within the framework, can take any form that the course designer prefers, including pop-up windows, assessments, video comparisons, etc. Future work will include a study as to the effectiveness of various AAR's on the states of various students, through the results of this data collection study we will assign the AAR that works best for Newton's Playground.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

The recommendations for GIFT and for the future of this work are simple and straightforward to include, but involve significant work. There is an outstanding question of whether all of the above techniques will, in practice, be useful in an educational context. In order to show the use of this technology, the project has chosen to use Newtonian Talk (Zhao, W., M. Ventura, et al., 2015) from the GIFTSym3 conference to serve as a testbed for the application of these technologies. An experiment is planned within the next calendar year to show the effectiveness of the above techniques in practice. If successful learning gains are seen, the implication is that they should be included in the GIFT modules during future releases. The technology discussed should have simple application across all other domains of instruction, with zero (or very little) integration required, representing significant opportunity.

REFERENCES

Carlin, A., Dumond, D., Freeman, J. & Dean, C. (2013). Higher Automated Learning through Principal Component Analysis and Markov Models. In Artificial Intelligence in Education (pp. 661-665). Springer Berlin Heidelberg.

Corbett, A. T. & Anderson, J. R. (1991). Feedback control and learning to program with the CMU LISP tutor. Department of Psychology, 28.

Ericsson, K. A., Krampe, R. T. & Tesch-Römer, C. (1993). The role of deliberate practice in the acquisition of expert performance. Psychological review, 100(3), 363.

Geman, S. & Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. Pattern Analysis and Machine Intelligence, IEEE Transactions on, (6), 721-741.

Higgins, R., Hartley, P. & Skelton, A. (2002). The conscientious consumer: surveillance sidering the role of assessment feedback in student learning. Studies in Higher Education, 27(1), 53-64.

Hruska, M., Dewey, M., Paonessa, G., Paonessa, M., Poeppelman, T., Nucci, C. & Ayers, J. (2013). Software Design Document (SDD) for Soldier Performance Planner Application. Report to US Army RDECOM Army Research Laboratory – STTC. Problem Solutions.

Lord, F.M. (1980). Applications of item response theory to practical testing problems. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Rowe, J. P. (2013). Narrative-centered tutorial planning with concurrent Markov decision processes. North Carolina State University.

Schooler, L. J. & Anderson, J. R. (1990). The disruptive potential of immediate feedback. In Proceedings of the Twelfth Annual Conference of the Cognitive Science Society, 702-708, Cambridge, MA.

Sottilare, R. A., Goldberg, B. S., Brawner, K. W., & Holden, H. K. (2012, January). A modular framework to support the authoring and assessment of adaptive computer-based tutoring systems (CBTS). In The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)(Vol. 2012, No. 1). National Training Systems Association.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

U.S. Department of the Army, Headquarters (2011). The U.S. Army Learning Concept for 2015 (TRADOC PAM 525-8-2). Fort Monroe, VA: Author.

Zhao, W., M. Ventura, et al. (2015). GIFT-Powered NewtonianTalk. Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3), Orlando, FL.

ABOUT THE AUTHORS

Dr. Alan Carlin *is a Senior Research Engineer at Aptima, Inc. His interests focus on problems of artificial intelligence and machine learning. These include problems of decision making under uncertainty, communication between members of a team, and meta-reasoning for decision-makers. His publications include works on Decentralized Partially Observable Markov Decision Processes, game theory, and distributed meta-reasoning in uncertain environments. While at Aptima, he has developed intelligent training systems, NextGen pilot alerting systems, and data mining systems. Dr. Carlin received a Ph.D. in Computer Science from the University of Massachusetts, an M.S. in Computer Science from Tufts University, and a dual B.A. in Computer Science and Psychology from Cornell University. As part of his M.S., he also completed the MIT Lincoln Scholar's Program, sponsored by the Massachusetts Institute of Technology.*

Ms. Diane Kramer *is a Principal Software Engineer and Certified Scrum Master at Aptima, Inc. with over 25 years of experience designing and developing software applications using various programming languages and platforms. Additionally she has academic experience teaching Computer Science at both the college and high school levels. Her current work at Aptima involves leading small teams of engineers, and developing applications for adaptive training, working with scientists who encode algorithms such as Partially Observable Markov Decision Process (POMDP) and Best Fit Optimization (BFO) models. One example is a Small Business Innovative Research (SBIR) Phase II project involving developing a training platform for Full Motion Video Imagery Analysts. Ms. Kramer received a M.S. in Computer Science from Worcester Polytechnic Institute, and a B.A. in Computer Science from the University of Massachusetts, Boston. She is a member of the Association of Computing Machinery, and the national Computer Science Teachers Association.*

Ms. Chris Nucci *is a Senior Software Engineer at Aptima, Inc. with engineering experience in a variety of fields including live training, gunnery, cyber security, secure coding. His current work focuses on the ongoing development of SPOTLITE™, Aptima's tablet-based application for performance measurement, data collection, debrief, and analysis. Other work has included development on Aptima's PM Engine™ and Army's GIFT framework. His previous experience at Lockheed Martin includes development of a Storage Area Network (SAN) configuration service for the National Cyber Range (NCR), development of gunnery and gallery target and mover controller software for the Saudi Arabian National Guard (SANG) Range Modernization program, exercise planning and AAR tools for U.S. Army Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI) Common Training Instrumentation Architecture (CTIA), Live Training Transformation (LT2), and Combat Training Center (CTC), as well as experience in secure coding, databases, and web application development. Mr. Nucci received a M.S. and B.S. in Computer Science from the Florida Institute of Technology.*

Dr. Jared Freeman *is Chief Scientist at Aptima, Inc. As a Cognitive Scientist, Dr. Freeman investigates human problem solving and decision making in real-world settings, and defines methods of monitoring and managing these processes using modeling and training applications.. He has served as P.I. on projects to measure, monitor and manage team knowledge and collaboration; automate the analysis of voice communications; automate the analysis of written usability documents; and model the fit between warfighter capabilities, organizational structure, technologies, and mission processes.*

Dr. Keith Brawner *is a researcher for the U. S. Army Research Laboratory's Human Research & Engineering Directorate (ARL-HRED), and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT). He has 10 years of experience within U.S. Army and Navy acquisition, development, and research agencies. He holds a Masters and PhD degree in Computer Engineering with a focus on Intelligent Systems and Machine Learning from the University of Central Florida. His current research is in ITS architectures and cognitive architectures. He manages research in adaptive training, semi/fully automated user tools for adaptive training content, and architectural programs towards next-generation training.*

THEME III:
PRACTICAL APPLICATIONS OF GIFT

From Concept to Publication - A Successful Application of Using GIFT from the Ground Up

Michael W. Boyce
Army Research Laboratory and Oak Ridge Associated Universities

INTRODUCTION

The idea for this paper comes from the desire to help people who are unfamiliar with, or novices using the Generalized Intelligent Framework for Tutoring (GIFT, Sottilare, Brawner, Goldberg, & Holden, 2012) understand how GIFT can effectively gather and collect data without large amounts of programming or intelligent tutoring experience. By providing guidance for building an assessment for an experiment, a potential author can give GIFT a try without intimidation. This paper serves as a companion to previous papers on experimentation and GIFT (Goldberg & Cannon-Bowers, 2015; Sinatra, 2014) as well as papers discussing the authoring process using GIFT (Brawner & Ososky, 2015). Sinatra (2014) discusses GIFT from the perspective of a research psychologist. She explains the use of the Course Authoring Tool (CAT), the creation of surveys using the Survey Authoring System (SAS), and the exporting of data using the Event Reporting Tool (ERT) to collect data. Brawner & Ososky (2015) discuss the ongoing development with GIFT as a result of user requirements or feature requests. The paper discusses GIFT as a content manager, where it assists in the interaction with the Augmented REality Sandtable (ARES, Amburn, Vey, Boyce & Mize, 2015).

The structure of this paper is as follows:

- Experiment Background – this section discusses the rationale of the experiment as well as the different components that make the system work. This is relevant because the discussion of GIFT tools that follow are intended to help facilitate communication between system components.

- A Perspective of Authoring in GIFT – this section discusses the various tools available in GIFT and how those tools can provide the communication to exchange information that can be output for data analysis. Emphasis is on understanding the connection between the CAT and the SAS, and Authoring in the Domain Knowledge File (DKF). This is relevant to the user as it will help in understanding the type of information the ERT is receiving.

- Retrieving Data from GIFT – this section discusses features of the ERT that assist in getting the data into a format that experimenters can use to import into a data analysis package. Rather than analyzing the ERT in its entirety, this section focuses on helpful aspects of the ERT that might not be readily apparent to the experimenter. This section is relevant to the user because it can provide them with a few useful operations when working with data.

- Recommendations for GIFT Features – This section talks about useful features that can assist in the development of future GIFT research projects. This is relevant to users because it provides an understanding of current limitations from an experimenter perspective, but also ideas for future GIFT development.

The paper ends with a short conclusion that provides helpful tips for users attempting to learn GIFT.

EXPERIMENT BACKGROUND

This research project focuses on content to support military tactics instruction through the use of displays that GIFT manages. Through asking a series of questions, GIFT can gather data on learner performance. Whether or not a specific tactical answer is correct or incorrect is usually a product of the subjective judgment of the instructor. However, by reducing the potential responses using specific tactics scenarios, it is possible to do a check on learning for cadets.

Research Objective

Existing research has demonstrated 3D perspective display advantages in terms of integrated complex understanding (Haskell & Wickens, 1993). The research objective is to examine the effects of different types of interface displays involving integrated information on the understanding assessment strategies to military tactics. This will be assessed through participant answers to questions on tactics in the context of a scenario. Performance measures (e.g., accuracy and time on task), an individual difference measure (i.e., mental rotation), and a physiological measure (i.e., electrodermal activity), as well as video and audio recording will be used to provide pilot data for a larger scale study involving military tactics.

Course Components

The course on military tactics consists of the following components: 1. ARES which displays map content onto the sand. 2. GIFT which controls presenting content in a correct sequence for the learner. 3. The Affectiva Q-Sensor (Poh, Swenson & Picard, 2010) which measures electrodermal activity via a Bluetooth connection with GIFT (Figure 1).

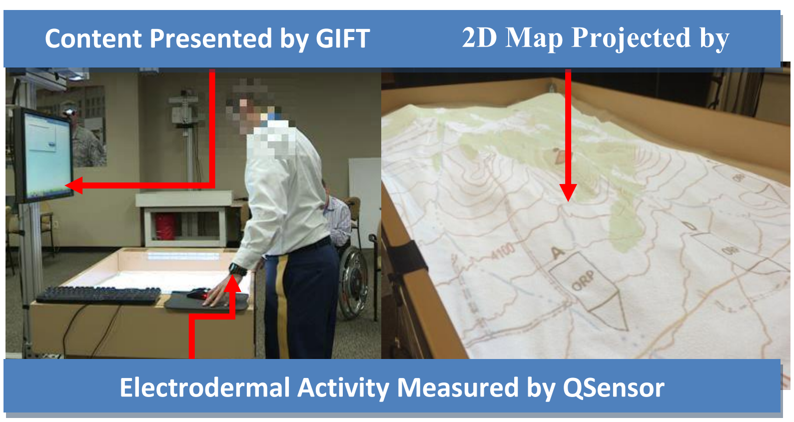


Figure 25: Participant Interacting with System / ARES Display

The Augmented REality Sand Table (ARES)

ARES is a research system test bed that uses a commercial projector, monitor, laptop, and Microsoft Kinect. For this experiment, ARES serves as a device to project a two dimensional map onto either a flat or contoured sand surface to see if the display medium alters performance outcomes.

The Affectiva Q-Sensor

The Affectiva Q-Sensor is a wearable skin conductance sensor that identifies when there are changes in the electrical potential of the skin. Electrodermal activity can increase / decrease with emotional states (Poh, Swenson & Picard, 2010). The sensor connects with GIFT via the sensor configuration file which needed to be added as part of the GIFT course. It is also synced to GIFT domain session information to provide data relative to where a participant is within the course. The Q-Sensor collects data at a rate of 4 Hz. The Q-Sensor uses Bluetooth to connect with GIFT.

The Generalized Intelligent Framework for Tutoring (GIFT)

GIFT serves as the content manager for the course, synchronizing question content with ARES displaying maps. Unlike other GIFT courses, this course is not custom for the individual learner but rather serves as an assessment of participant knowledge. Vogel-Walcutt, Koss, Phillips & Stensrud (2016) note that data from these experiments can help determine where participants are in terms of course knowledge.

Procedure

All participants sign an informed consent approved by the Institutional Review Board for the Army Research Laboratory (ARL IRB). Participants then place the Q-Sensor on their wrist. Participants also fill out a series of surveys: a demographic survey followed by the Vandenberg and Kuse (1978) mental rotation test. Participants also take the Self-Assessment Manikin (SAM, Bradley & Lang, 1994). The SAM is a picture-oriented scale to assess dimensions of valence, arousal, and dominance. Then participants answer a pretest on their knowledge of concepts related to platoon and squad level tactics. The experimental task consists of eight military tactics questions, validated from course instructors from three separate institutions. The post experiment activities includes an administration of the NASA Task Load Index (NASA-TLX, Hart & Staveland, 1988), a second administration of the SAM, a post-test on tactics, and a usability survey.

A PERSPECTIVE ON AUTHORING IN GIFT

This section will walk through processes of authoring content for the experiment in GIFT. The procedures are specific for the experiment and do not cover all possible authoring pathways. When launching GIFT, the author seeing three icons across the top of the screen: my courses, my stats, and my tools. When there are no courses built, both my courses and my stats are empty. That leaves clicking on my tools. Clicking on my tools brings up four icons: Course Authoring, Survey Authoring, AutoTutor Script Authoring, and TRADEM. This discussion will focus on course authoring and survey authoring since those are the primary elements for this experiment. This is followed by a discussion of authoring from within the DKF authoring tool.

Working with Course Authoring Tool (CAT) and Survey Authoring System (SAS)

When authoring a new course file, one of the first options available to the author is name and description. Although there are other fields available, a novice user may not understand enough to fill them in. Survey context is important, but that may not be apparent until the author needs to add surveys. It is relatively simple to add guidance messages, which can assist in laying out a structure for the course. When a user tries to add a survey, they will quickly realize that a list of surveys is not present. The error message explains that since there is no survey context, the surveys will not show up. During the development of the experiment, the survey context is just a single number so it was necessary to go into the survey authoring system to find out more information about the context. Switching to the SAS, there is a survey context tab available to the user. This allows the user to create their own context which will also allow them to populate it with surveys.

When creating a survey in GIFT, it is often a good practice for novice users to begin by copying and modifying an existing survey. This allows for an exploration into the different surveys options, and provides insight into the relationship between survey questions and surveys. When creating a question, there are some straightforward options such as the question itself and the type of question (i.e. Multiple Choice vs. True / False vs. Slider Bars, etc.). One area that may cause confusion is shared answer sets. The way shared answer sets work is that the user can either define the answers that they want for the question, or they can use a predefined answer set. Also important to note, next to the shared answer set is the manage categories link. The manage categories link is a way to tag questions so that when searching for a specific question, it can filter around a specific name or experiment. Once surveys are created, it is possible to go back to the course authoring tool to import it into the course, via a survey course element.

Another area which may cause confusion is when trying to develop surveys with images and with specific layouts for answer choices. For the purposes of the experiment, the survey was the Vandenberg and Kuse (1978) Mental Rotation Test. The test presents a three-dimensional figure and asks for two correct locations of that figure out of a set of four, where the other two options are created specifically as distractors with minor differences from the original figure. Further, there is an instruction page at the front of the survey that doesn't have any questions associated with it. When developing the survey for the experiment, GIFT does not have a way to present an image by itself only in conjunction with the question. However there is a workaround: each time there is an image that needs to be displayed, create a validation question which requires participants to select that they had read the text above. This is often seen in the context of user agreements for software programs or add-on features, and may help in achieving the goal of reading the information. To resolve being able to position the answer choices correctly (in this case below each figure) a left margin feature adds the capability to shift question responses off of the left margin. There is also the ability to define the spacing in between the options for more precise positioning. A multi-select feature is also available to allow for the selection of two options (see Figure 2).

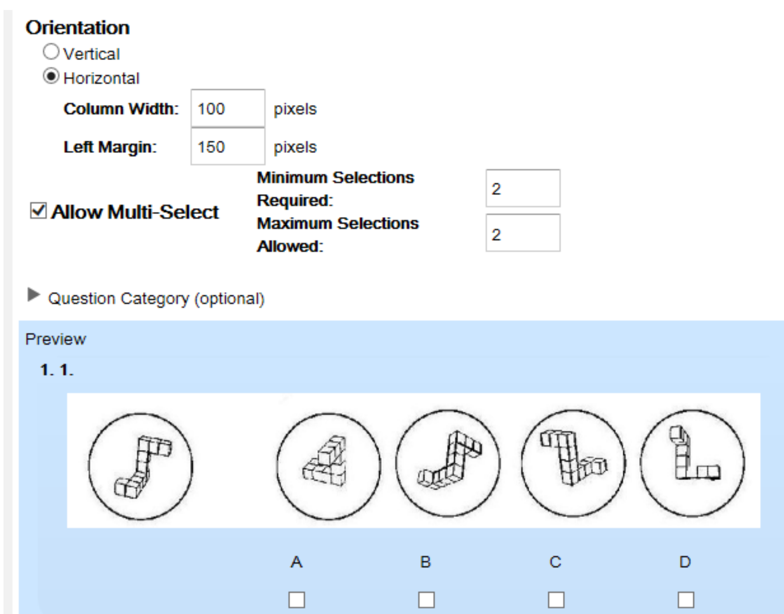


Figure 2. Multiselect and Spacing Adjustments on SAS

Once images are associate with the appropriate survey, it is possible that these images may need to be moved to another GIFT installation. In order to ensure that the images moved with the export, the user can manually move the folder that holds the images. In the experiment this folder was located within the GIFT directory and contains all images that were uploaded by the user, the path is: GIFT\data\surveyWebResources\uploadedImages.

Working with the Domain Knowledge File Authoring Tool (DAT)

A part of the course includes eight tactics questions which correspond to maps displayed on the sand. Each of the tactics question needs to display with the appropriate map. To do this, the author needs to interact with the DKF, which the authoring XML tool helps with. To access the XML tool, the GIFT control panel has to be launched (GIFT\scripts\launchControlPanel.bat). From within the control panel, access to the XML editing tools is through using the authoring tools tab. Once in the authoring tools tab, there is a sub tab for the desktop tools. The DKF Authoring Tool (DAT) is the second option on the list (see Figure 3).

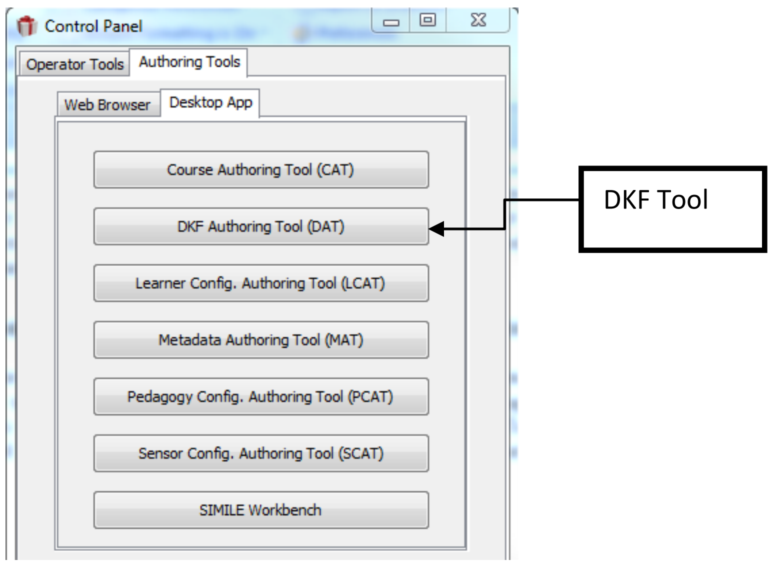


Figure 3. Display of the DKF Authoring Tool within GIFT Control Panel

Once the authoring tool is running, the first step is to locate the appropriate DKF files. In the case of this experiment, the development team use a scaffolding DKF so that the edits are specific to the answer choices. At the top, the first few fields are self explanatory: name and description. For the purposes of this experiment, editing the learner ID is not necessary, but editing DKF resources is necessary due to its reference to the survey context. Like the course authoring tool described earlier, the DKF authoring tool uses the survey context to populate the survey and survey question / response fields.

Following this, the tasks menu needs to expand for several levels in order to reach the specific question and answer choices. A complete discussion of the breakdown is beyond the scope of this paper; however relevant information includes that each question had a timer and a single concept which consisted of the one item survey (i.e. Q1 was a single concept, Q2 was a single concept etc). Each of these tie to the appropriate survey key, as well as the appropriate answer key. Figure 4 provides more detail on the parts of the DKF that correspond to survey elements.

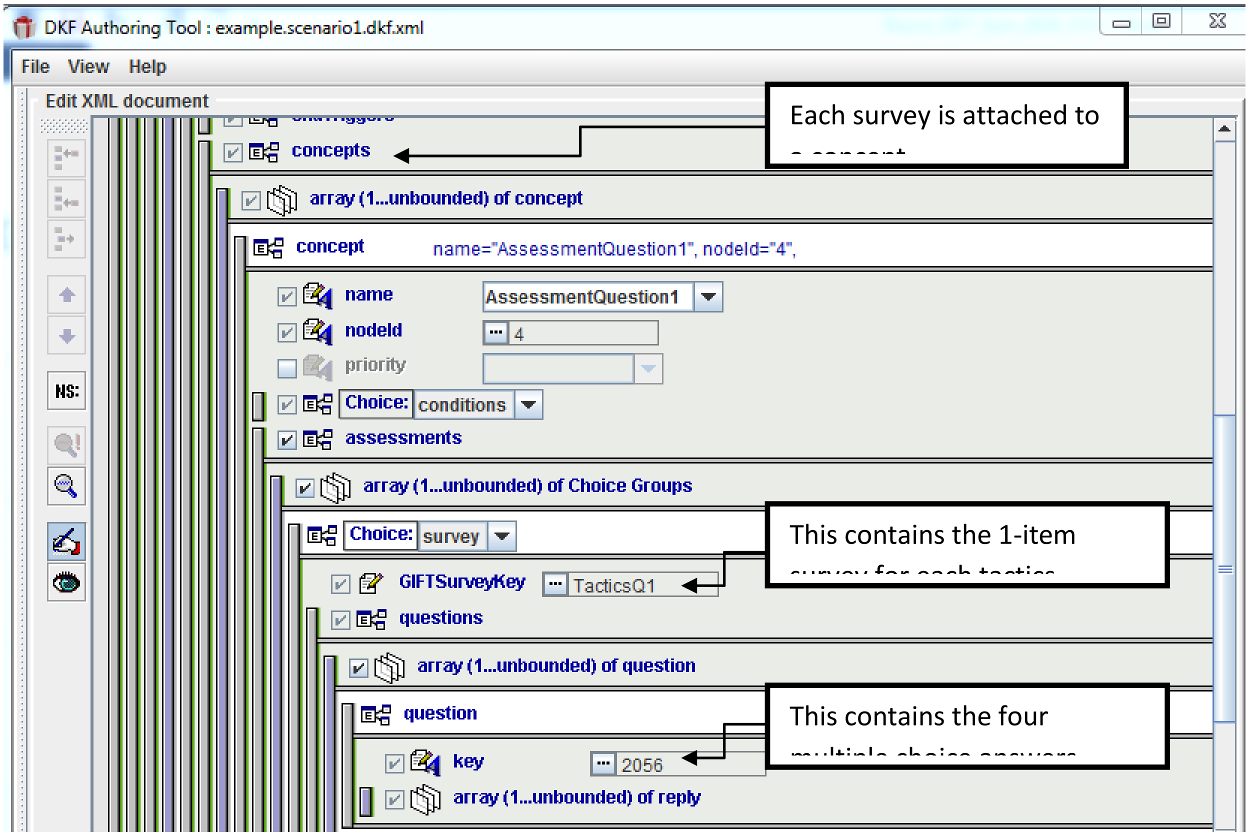


Figure 4. DKF Authoring Screenshot

It was through using the DKF that the course was successful at exchanging messages between GIFT and ARES.

RETRIEVING DATA FROM GIFT

Once data collection is complete, the next step for the researcher is to extract the data from GIFT for analysis. The ERT supports this function in GIFT.

Since the data from the experiment contains several different types of information, various aspects of the ERT are used. The most common feature, as Sinatra (2014) mentions, is the submit survey results option. This feature provides the output of any surveys that accompany a GIFT course. The naming convention which submits survey results uses consists of the name of the survey, the instance number of the survey, and the individual number of the question itself. The reason there is an instance number and an individual number is because there are situations where the same question could be used in multiple places throughout the course.

Since the experiment research questions are also interested in electrodermal activity, sensor data is also captured via the sensor writer data option (note: in order to get the sensor data associated with a domain session, the researcher needs to select both the domain session and the unfiltered sensor information file (GIFT/output/sensor). This is one of three sensor options within the ERT. However, this particular option because it contains the raw sensor data as GIFT received it. Also included was temperature information because of the impact that it could have on electrodermal responses.

Another feature of the ERT that is relevant for the experiment is the course state option. The course state is a pointer for when GIFT moves between transitions. The course state also maintains a timestamp associated with each transition. This is a valuable piece of information as it allows you to synchronize activities that are happening with the participant and the corresponding activities occurring in GIFT. For example, if you were trying to locate the electrodermal response within 10 seconds of exposure to a stimulus you could use the course state feature to determine when the interface changed and then monitor responses from there.

In establishing the default columns for the ERT, domain session time is the primary time field. To compare across participants, the user ID can be a good delineator, even though participant data files themselves are handled one at a time. Each participant is handled individually because of the mixture of between subjects and within subjects data. There is not currently a way to dynamically map sensor configuration files and domain session files to make sure that the data presented represented the same participant in GIFT. The content column option is also necessary because it is through the content column that GIFT representations information for the course transitions in GIFT.

RECOMMENDATIONS FOR GIFT FEATURES

For future development supporting GIFT users, there is a need to be able to provide the user with an easy-to-read printout of how they did, what they got right and what they got wrong. An alternative solution could be to provide them a summary in GIFT of what they get right and wrong, and then provide some recommendations on what they could use more practice in. This way the researcher is not providing the participant with the actual answers that they could copy and share, but they are still using GIFT to provide them with some level of performance feedback.

An additional feature that would be helpful is to take the existing ERT and provide a series of videos on how to export data from the ERT to getting the data into an SPSS ready format. A lot of the postprocessing time for the study was taken up trying to manipulate the data that was already present in the ERT into an SPSS ready format. What is missing is a way to guide the user in how to do things like converting the data from a between subjects row-based design to a within subjects column based design.

Another recommendation would be customizable views for GIFT. Depending on the style of the author, they may prefer different arrangements or amounts of information. As an example, a researcher may want to have a minimalist interface showing only what they desire to see. This is a philosophy of Edward Tufte in which he called for the elimination of anything that wasn't essential information to display (he used the term "chart junk", but it could be applied to interfaces as well). Information that an author doesn't need can serve as a distractor and slow their workflow. It is possible that this could be a feature available for advanced users of GIFT, so that beginning users do not miss important features that could be valuable to them.

CONCLUSION

The purpose of this paper was to discuss the use of GIFT to develop an experiment on military tactics. For the interested reader the results of the experiment are currently in press (Boyce, Goldberg, & Moss, 2016, In Press; Boyce, Reyes, et al., 2016, In Press). Overall, even with the associated challenges, the experiment was a success. For individuals new to GIFT, below are a couple of helpful tips to make GIFT easier to learn.

- Start with developing a complete research protocol. Many times during the authoring process when the research gets confused, it is helpful to fall back to the protocol as a guide. This helps to keep focus on the fundamental research questions and not get sidetracked by features or capabilities.

- Choose surveys that will be easy to integrate in a text type format. A lot of time was used during this study to build an electronic version of the paper based mental rotation test. Working within the current functionality can decrease development time and increase stability of the course.

- Remember that surveys need to be linked to the database. Currently that functionality is through the survey context, but regardless of how it is implemented in the future, understanding architecture can go a long way to the successful creation of courses.

ACKNOWLEDGMENTS

The author would like to thank our subject matter experts from the SFC Paul Ray Smith Simulation and Training Technology Center, United States Military Academy at West Point, the Florida National Guard at Camp Blanding and the faculty of the University of Central Florida ROTC program.

Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-12-2-0019. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

REFERENCES

Amburn, C. R., Vey, N. L., Boyce, M. W., & Mize, J. R. (2015). The Augmented REality Sandtable (ARES) (ARL-SR-0340). Orlando, FL: Army Research Laboratory.

Boyce, M. W., Goldberg, B., & Moss, J. D. (2016, In Press). *Electrodermal Activity Analysis for Training of Military Tactics* Paper presented at the Human Factors and Ergonomics Society Annual Meeting, Washington, D.C.

Boyce, M. W., Reyes, R. J., Cruz, D., Amburn, C. R., Goldberg, B., Moss, J. D., & Sottilare, R. A. (2016, In Press). Effect of Topography on Learning Military Tactics - Integration of the Generalized Intelligent Framework for Tutoring (GIFT) & the Augmented REality Sandtable (ARES).

Bradley, M. M., & Lang, P. J. (1994). Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, *25*(1), 49-59. doi: 10.1016/0005-7916(94)90063-9

Brawner, K., & Ososky, S. (2015). *The GIFT 2015 Report Card and the State of the Project.* Paper presented at the Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym3).

Goldberg, B., & Cannon-Bowers, J. (2015). Feedback source modality effects on training outcomes in a serious game: Pedagogical agents make a difference. *Computers in human behavior, 52*, 1-11.

Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Advances in psychology, 52*, 139-183. doi: 10.1016/S0166-4115(08)62386-9

Haskell, I. D., & Wickens, C. D. (1993). Two- and Three-Dimensional Displays for Aviation: A Theoretical and Empirical Comparison. *The International Journal of Aviation Psychology, 3*(2), 87-109. doi: 10.1207/s15327108ijap0302_1

Poh, M.-Z., Swenson, N. C., & Picard, R. W. (2010). A wearable sensor for unobtrusive, long-term assessment of electrodermal activity. *Biomedical Engineering, IEEE Transactions on, 57*(5), 1243-1252.

Sinatra, A. M. (2014). *The research psychologist's guide to GIFT.* Paper presented at the Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym2).

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

Vandenberg, S. G., & Kuse, A. R. (1978). Mental rotations, a group test of three-dimensional spatial visualization. *Perceptual and motor skills, 47*(2), 599-604.

Vogel-Walcutt, J. J., Ross, K. G., Phillips, J. K., & Stensrud, B. (2016). Improving the efficiency and effectiveness of adaptive training: using developmental models as a framework and foundation for human-centred instructional design. *Theoretical Issues in Ergonomics Science, 17*(2), 127-148. doi: 10.1080/1463922X.2015.1111460

ABOUT THE AUTHOR

Dr. Michael W. Boyce *is a third-year postdoctoral research associate at the Army Research Laboratory, Human Research and Engineering Directorate (ARL-HRED) under the mentorship of Dr. Robert Sottilare. His current focus is on investigating the use of adaptive training systems to support the instruction of military tactics and understanding how to design for ill-defined domains.*

The Updated Research Psychologist's Guide to GIFT

Anne M. Sinatra
U.S. Army Research Laboratory

INTRODUCTION

DON'T PANIC. Those two words were made famous by the book *The Hitchhiker's Guide to the Galaxy* (Adams, 1979), as their being on the cover was one of reasons that the fictitious guide had sold more copies than the *Encyclopedia Galactica* (it was also slightly cheaper). Those classic words were also included at the beginning of my original *Research Psychologist's Guide to GIFT* paper (Sinatra, 2014) that was presented at GIFTSym2. Having "Don't Panic" at the top of the page is meant to serve as a reminder that even though things can seem overwhelming at times when using software such as the Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden, 2012), there are documents like this one that include explanations on how to easily and efficiently use GIFT. In a world filled with movie sequels and prequels this guide serves as a sequel of sorts to the original *Research Psychologist's Guide to GIFT*. However, it does not necessarily replace it. The information provided in it is still helpful, however, many changes have occurred in the past 2 years and the current work provides information about how to use updated features in GIFT, and considerations that should be made as a result of the changes. This guide itself will be tightly coupled with the releases surrounding it, GIFT 2015-2X, GIFT Cloud/GIFT Virtual Open Campus, and the soon to be released GIFT 2016-1. Perhaps the current guide is a middle piece of a trilogy (or a series of trilogies), and hopefully future editions will cover the new advancements that are made with GIFT over time.

THE GENERALIZED INTELLIGENT FRAMEWORK FOR TUTORING AND THE RESEARCHER

GIFT has been designed to allow for low-cost, flexible adaptive tutoring to be authored by individuals of varying levels of expertise and backgrounds. Individuals who wish to create intelligent tutoring systems (ITSs) can bring their materials to GIFT, and using the included authoring tools they can create tutors that will adapt to the individual student's performance based on rules that they have authored. While the primary function of GIFT is to support creating adaptive tutoring systems, an additional goal of GIFT is to provide a means of conducting research. The flexibility that is available in GIFT provides many opportunities that do not exist elsewhere to test/compare components of ITSs, which could support research on what belongs in an ITS learner or pedagogical model. However, the GIFT architecture and authoring tools can also be leveraged to conduct traditional research that research psychologists conduct.

A research psychologist may be interested in testing out varying strategies and adaptive remediation paths that a learner may take in an ITS. GIFT provides a means of doing this, as well as examining the types of feedback and modalities of the feedback that are presented to a learner. However, one extremely useful function of GIFT is being able to construct a linear course that includes segments of text, surveys, opening and closing external applications, and blocks of text.

Oftentimes, conducting psychology experiments can be time consuming and requires experimental monitors and research assistants who have responsibilities such as opening and closing programs or presenting surveys to the participant. GIFT allows for this process to be streamlined and only requires an assistant to set it up, at which point the participant can proceed on his or her own through the experiment. This also greatly increases the number of participants that can be run at the same time. Studies that have used GIFT have been able to run multiple participants simultaneously at different computers with limited experimenter support (Goldberg & Cannon-Bowers, 2015; Sinatra, Sims, & Sottilare, 2014; Sinatra, Sottilare, & Sims, in press).

Conducting an Experiment with GIFT

Authoring a GIFT "course" (experimental condition)

Once an experiment is planned it is necessary to create a GIFT course file which the participants will experience. For the purposes of traditional psychology experiments, a linear course can be created in GIFT. In the current design, if there are multiple experimental conditions each one would be a separate course, which would then be selectively opened for the participant based on the condition he or she was assigned. Figure 1 provides a screenshot of the GIFT Authoring Tool in GIFT 2015-2X. This tool is used to create the flow of a course or experiment. "Transitions" are course elements that are added by the author in a linear fashion, which is demonstrated by the list on the left side of the screen under "Course Properties". For instance, in the course displayed in Figure 1, initial guidance (text that is presented) is provided to the participant, followed by a survey, guidance, a PowerPoint based interactive tutorial (Training Application), a survey, etc. This is the experimenter defined course flow including the materials and the order in which they will be presented. Among the available transitions that an experimenter can use are Guidance, Instructions, Training Applications (e.g., PowerPoint), Surveys, and AAR (After Action Review). Existing surveys can be used, or new ones can be generated using the Survey Authoring System. Among the question types that can be generated are multiple choice, short answer, fill-in-the-blank, and questions with a slider-bar for input. Individual questions can be tagged with titles by the experimenter so that the survey data can be easily examined after it is exported. Training Applications are external applications that GIFT can interface with and open/close. At current time these include PowerPoint, Virtual Battlespace 2, and Tactical Combat Casualty Care. PowerPoint is a very useful tool as it can be made interactive using Visual Basic for Applications (VBA). A developer's guide which is available for GIFT provides additional information about how to integrate GIFT with new external training applications. If adaptive feedback is desired in a Training Application a Domain Knowledge File (DKF) with specific feedback can be authored.

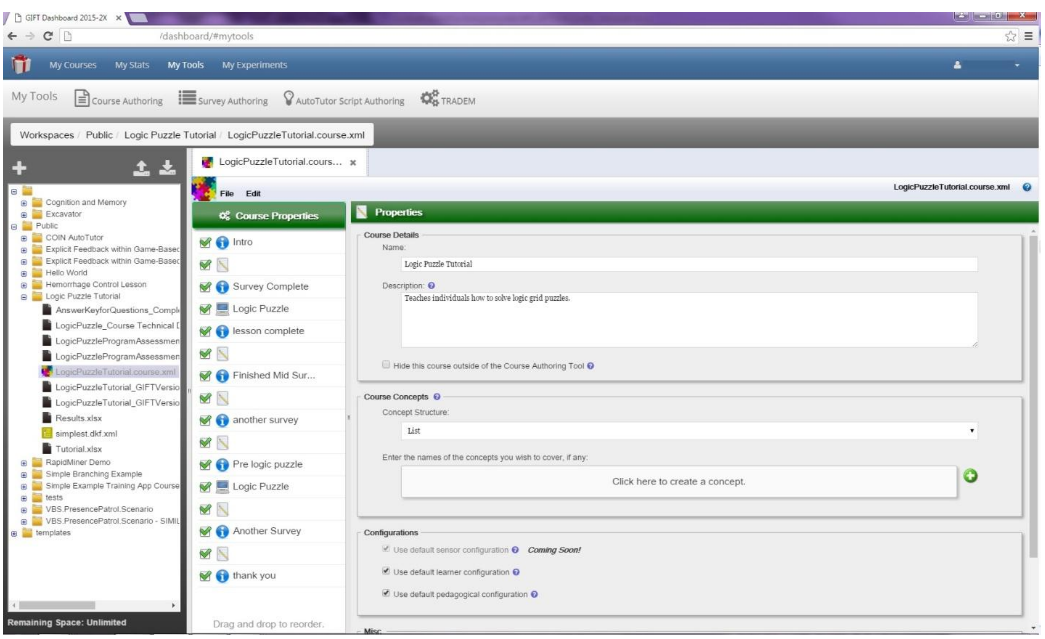


Figure 1. Screenshot of a course in the GIFT Authoring Tool in GIFT 2015-2X.

For the purpose of the current paper, the focus on using the GIFT Authoring Tool and Survey Authoring System is reduced as they are covered in depth in the original paper (Sinatra, 2014), and GIFT 2016-1 will have an authoring tool with a new interface in order to simplify the course authoring process. The GIFT Authoring Tool is nearly identical in both the GIFT Local desktop and the GIFT Cloud/GIFT Virtual Open Campus versions.

Updates to Authoring in GIFT that impact the researcher

A number of features have been added to GIFT in the past two years. In addition to feature additions, there have been behind the scenes changes in GIFT that may impact an individual who is creating and running an experiment with GIFT. Among these changes are updates to the authoring tools, updates to the way that external applications are launched, updates to the way that individuals login to GIFT, and cloud Functionality.

Domain Folders

In previous iterations of GIFT, an individual created a folder that was specific to his or her course by going into the file browser and creating a new folder in the **GIFT\Domain** directory. The individual would store all of the content that was specific to the course in the new folder for organization. While this is still true, the course folder is more visible in the authoring tool versions of GIFT 2015-2X, as all content in the folder can be viewed while constructing the course. Figure 1 provides a screenshot of the course authoring interface in GIFT 2015-2X. The left side of the screen shows the different course folders, and when they are expanded it lists all of the documents inside the folder. Additionally, of note, only one course.xml file can be included in each folder. This is a change from previous versions, and may impact an individual if he or she is running an experiment that has multiple conditions or versions of the course file.

Behind the scenes changes in GIFT have also impacted the way that files are accessed from these folders. While in the past, content such as PowerPoint show presentations were opened directly from the folder that they were initially saved in, it is now slightly different. When a PowerPoint show presentation is opened, a temporary copy of it is made and that is what is run as opposed to the original. While this should not cause any issues for normal PowerPoint shows, it does limit the ability for PowerPoint with VBA to save to an Excel file, since this generally requires that a file exists in the same directory as the PowerPoint file itself. Since a copy is being made, the directory is now a temporary one and it can lead to errors when trying to save files in this way. Other approaches that do not require storing information in the same directory as the running file are necessary if data is intended to be collected from PowerPoint.

Authoring: Public vs. Personal Courses

The course authoring tool has two different distinctions for courses: Public and Personal. Public courses are the general courses that are included in all instances of GIFT. These are also available in GIFT Cloud/GIFT Virtual Open Campus. These courses are great examples that can be used to see how to create new courses. These courses are not directly editable, however, they can be copied and the copy can be edited. When an individual creates a new course, or imports an existing course, it will reside in his or her own folder, which for the purposes of this paper is being called a personal course. As of 2015-2X, everyone who has logged in on the same computer will have access to everyone else’s courses on that computer. However, this is not the case if a course is created in GIFT Cloud/GIFT Virtual Open Campus. Courses that are created locally on a computer can be exported and then imported into GIFT Cloud/GIFT Virtual Open Campus, or into another computer’s local instance of GIFT.

Running Participants: Dashboard Mode vs. Experimenter Mode

In previous versions of GIFT, the interface that the participant or student logged in with was different than the experimenter interface. In the current versions when an individual logs into GIFT using his or her GIFTtutoring.org account, he or she is directed to the dashboard which includes all of the available courses. See Figure 2 for a screenshot of the login screen, and Figure 3 for a screenshot of the new default desktop screen that GIFT users see after logging in.

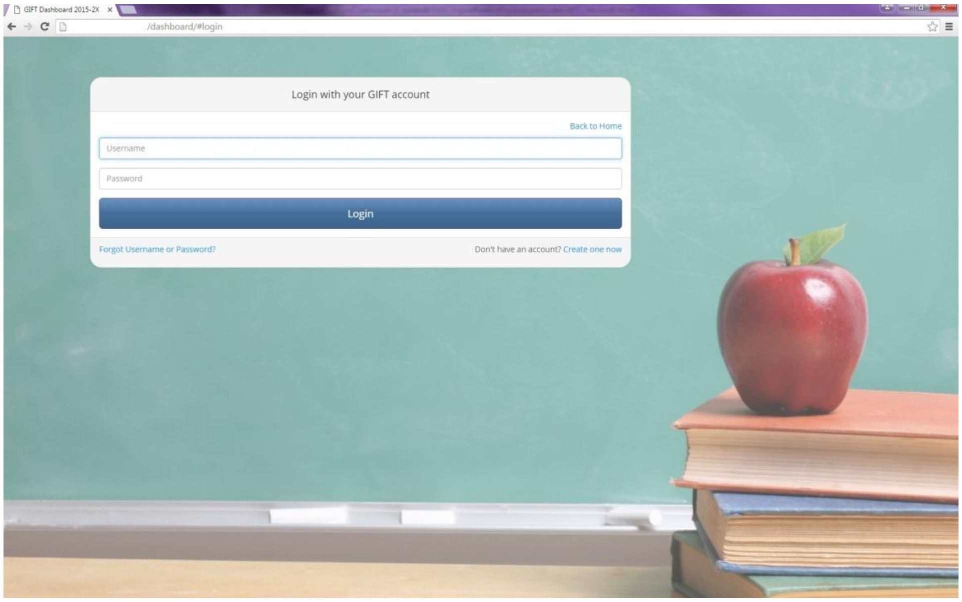


Figure 2. GIFT login screen; This screen is identical for the desktop and cloud versions, and for all user roles.

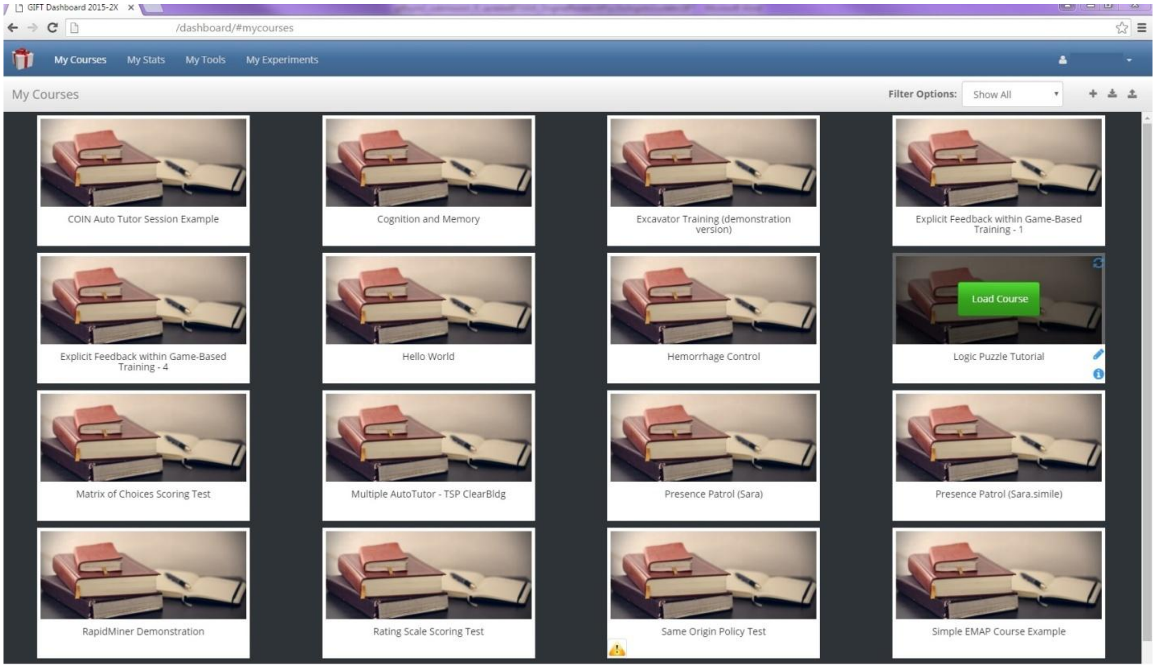
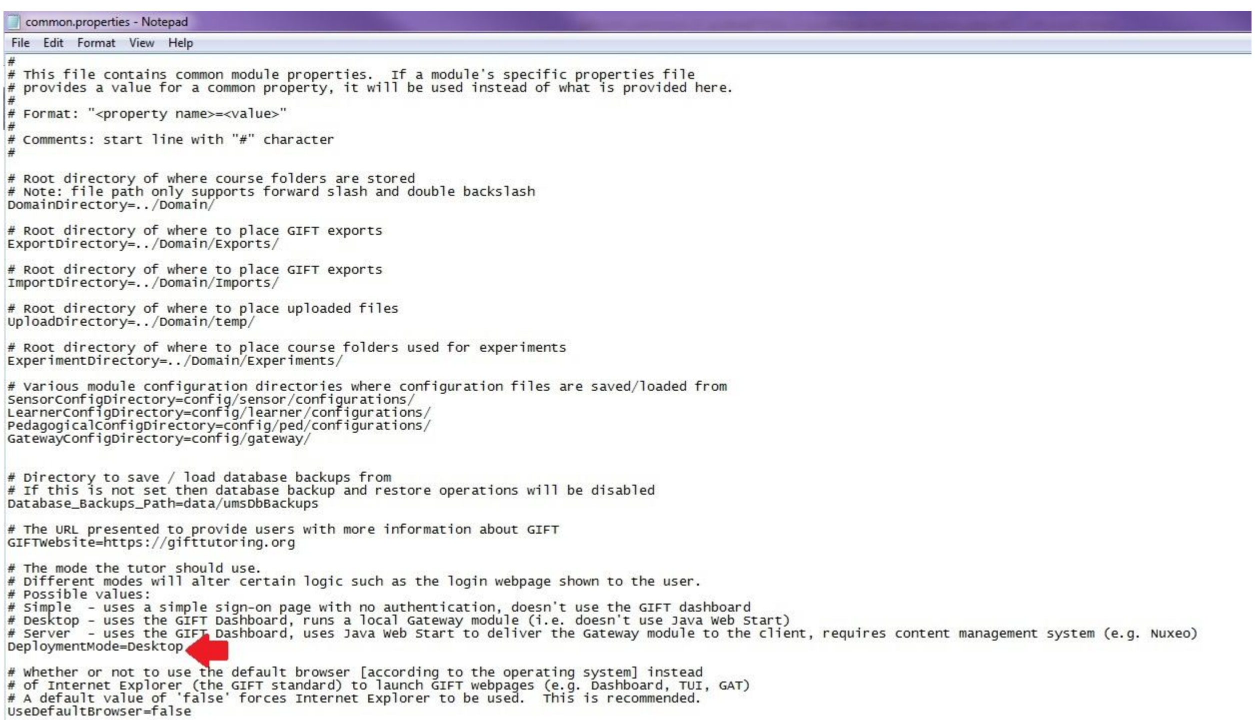


Figure 3. Screenshot of new GIFT Dashboard at login; Participants click on the course tile to run the course; Authors can edit the courses using “My Tools”.

As seen in Figure 3, the Dashboard includes all available courses. For the purposes of an experiment, all non-necessary folders can be removed from the **GIFT\Domain** directory such that only the relevant ones are available for participants to click on. However, in order to use this dashboard interface a login must occur with the GIFT account. While this may be acceptable for certain experiments, most experiments wish to use a participant number instead of the individual's actual information. One way to work around this problem would be to have participants create new accounts for their participation in the study that is not linked to their names (however, their email addresses would likely still need to be used for the account). The downside of using this approach is that it would artificially increase the GIFT user population, and it would likely be the only time that the particular user interacted with the system. A secondary approach for experimenters is to have participants login using the simple sign in (Experiment mode), which is not accessible from the GIFT dashboard. In order to access the simple login, users have to make an edit to a configuration file so that GIFT launches in Experiment mode instead of Desktop mode. A third option for experimentation and participant management is described in detail in the next section in regard to GIFT Cloud/GIFT Virtual Open Campus.

To access Experiment mode (the simple login screen) go to the **GIFT\config\common.properties** and double click to open the common.properties document (notepad is a good application to use when it asks what you would like to open it in). Find "DeploymentMode" in the document and type in "=Simple" (without the quotations) instead of "=Desktop". Then save the file. See Figure 4 for a screenshot of the configuration file that should be changed, with a red arrow pointing to the line to edit. If you already have GIFT running you will need to stop it, and then click "Launch GIFT". GIFT will now launch in the Experiment mode (simple login) instead of the default mode. This switch should be made when you want to examine what it will look like, and when you are ready to run your experiment. In order to use the normal login interface to make edits to your course you will need to edit the file again to switch it back to "DeploymentMode=Desktop". There are other methods to access the authoring tool, but it would be simpler to make the switch in the configuration file.



```
common.properties - Notepad
File  Edit  Format  View  Help
#
# This file contains common module properties.  If a module's specific properties file
# provides a value for a common property, it will be used instead of what is provided here.
#
# Format: "<property name>=<value>"
#
# Comments: start line with "#" character
#

# Root directory of where course folders are stored
# Note: file path only supports forward slash and double backslash
DomainDirectory=../Domain/

# Root directory of where to place GIFT exports
ExportDirectory=../Domain/Exports/

# Root directory of where to place GIFT exports
ImportDirectory=../Domain/Imports/

# Root directory of where to place uploaded files
UploadDirectory=../Domain/temp/

# Root directory of where to place course folders used for experiments
ExperimentDirectory=../Domain/Experiments/

# Various module configuration directories where configuration files are saved/loaded from
SensorConfigDirectory=config/sensor/configurations/
LearnerConfigDirectory=config/learner/configurations/
PedagogicalConfigDirectory=config/ped/configurations/
GatewayConfigDirectory=config/gateway/


# Directory to save / load database backups from
# If this is not set then database backup and restore operations will be disabled
Database_Backups_Path=data/umsDbBackups

# The URL presented to provide users with more information about GIFT
GIFTWebsite=https://gifttutoring.org

# The mode the tutor should use.
# Different modes will alter certain logic such as the login webpage shown to the user.
# Possible values:
# Simple  - uses a simple sign-on page with no authentication, doesn't use the GIFT dashboard
# Desktop - uses the GIFT Dashboard, runs a local Gateway module (i.e. doesn't use Java Web Start)
# Server  - uses the GIFT Dashboard, uses Java Web Start to deliver the Gateway module to the client, requires content management system (e.g. Nuxeo)
DeploymentMode=Desktop

# Whether or not to use the default browser [according to the operating system] instead
# of Internet Explorer (the GIFT standard) to launch GIFT webpages (e.g. Dashboard, TUI, GAT)
# A default value of 'false' forces Internet Explorer to be used.  This is recommended.
UseDefaultBrowser=false
```

Figure 4. Screenshot of common.properties document; change DeploymentMode, which his next to the red arrow, to =Simple to launch GIFT by default in Experiment Mode.

When GIFT is set for Experiment mode, clicking “Launch GIFT” will launch the web-browser with the sign on page that allows for users that are not logged into a GIFTtutoring.org account to run courses. Figure 5 shows what experiment mode looks like.

Figure 5. Experiment Mode (Simple Login)

Participant Management in Experiment Mode

There are a few different methods for participant management to ensure that your data can be lined up with the participant number that the individual is assigned. One approach is for the experimenter to create participant accounts ahead of time, and to tell the participant the number to enter on the login screen. To create a new participant account, the experimenter will click “New User”, then enter in a username (in the example this is “Participant2”). This username will be available in the GIFT logs, and data can be merged by row using the username. However, in order to login, the number assigned by the system will need to be used. See Figures 6 for an example of the user management in the current Experiment mode. As noted in Figure 6, the GIFT system assigned User ID will need to be logged in with regardless of the selected username. One approach to simplify the management of participant numbers is to assign participant numbers that are numeric only and are consistent with the User ID number that the GIFT system provides so that it leads to less confusion when examining data. It is also important to use unique pre-made User ID number accounts if participation is happening on multiple computers. Regardless of the approach that is used to logging in and assigning participant numbers, a survey question can be included within the course that requests that the individual enter the assigned participant number. The data can then be merged by username or User ID, and then sorted by the survey column in Excel in order to organize the participant data.

Figure 6. While the new user has been created in the image on the left with the name “Participant2”, the system has assigned it the User number of “3” (right). In order to login to this participant account the number “3” will need to be put in. The exported log can be merged by username or user number.

Extracting Data with the Local Desktop Version of GIFT

In the traditional approach to conducting research with GIFT on a desktop computer, data can be extracted from the participant logs using the Event Reporting Tool (ERT). All of the information about what the participant did in the session, including their survey responses are stored in their specific GIFT log. The selected information can be merged as desired and extracted into a .csv file, which can then be imported into Excel or SPSS for analysis. The ERT can be reached by going to **GIFT\scripts** and clicking **"**launchControlPanel" and then clicking on the "Event Report Tool (ERT)" button. The control panel interface is shown in Figure 7 (left).

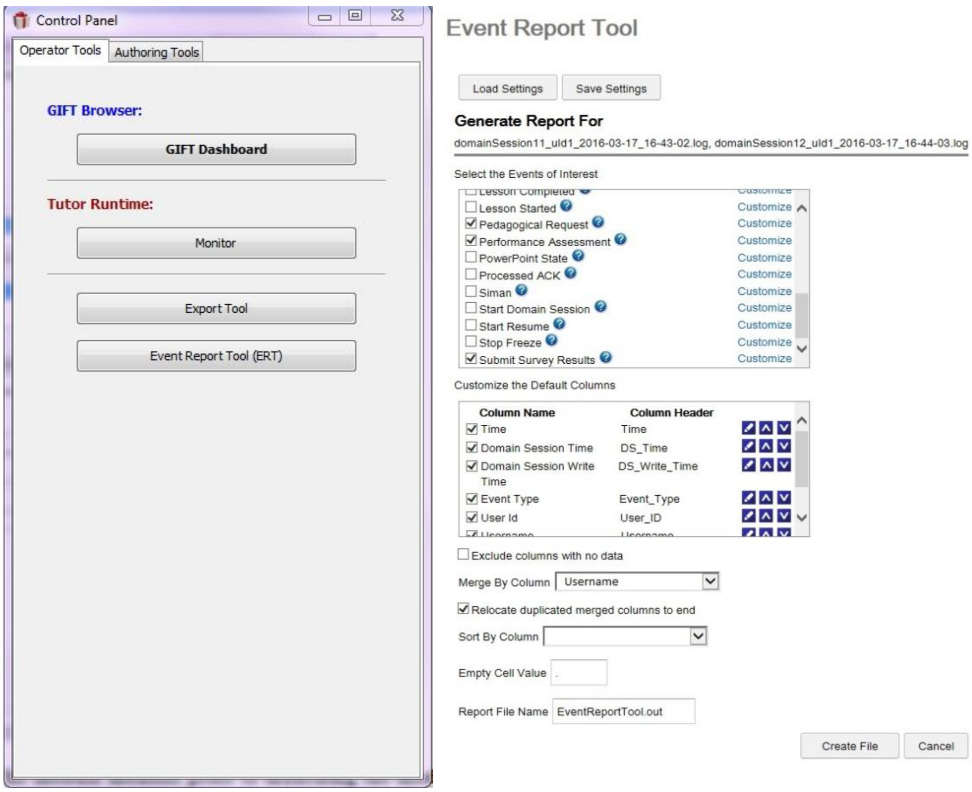


Figure 7. The ERT can be launched by the control panel interface (left). The ERT interface includes a number of different selection options (right).

Participant logs from sessions that were run on your computer will be in the **GIFT\output\logger\message** folder, and will be visible in the ERT interface (if you are running participants on multiple computers you may want to extract the data on each individual computer to keep everything separate and avoid multiple users with the same User ID; if you choose to extract them all on the same computer you will paste your log files in this directory). Multiple participant logs can be selected at the same time by holding down "shift" and making the selection. See Figure 7 (right) for a screenshot of the ERT interface. Different events of interest can be selected, and information is provided about them by clicking on the question mark button. One of the events that will be commonly used by research psychologists will be "Submit Survey Results" which provides all of the data entered in the surveys. To merge the files into separate rows for participants, select "Username" or "User ID" as appropriate in the Merge by Column. User ID is the system assigned number (i.e. "3" in Figure 6), Username would be the username entered by the participant or researcher (i.e. "Participant2" in Figure 6).

Once the file is created it will be stored in **GIFT\scripts\output** with the file name that you selected. The file can then be opened using Excel to save it as an .xls file if desired.

CLOUDY WITH A CHANCE OF RESEARCH: GIFT CLOUD/GIFT VIRTUAL OPEN CAMPUS

In November 2015 a cloud version of GIFT, known as GIFT Cloud/GIFT Virtual Open Campus became available online. While a local desktop version of GIFT is still available and supported, the cloud version includes a number of features that are present in the desktop version but does not require a download in order to run it. As GIFT Cloud/GIFT Virtual Open Campus is currently an alpha version, there are still many improvements and features that are expected to be added to it in the near future. A Quick Start Guide is available on GIFTtutoring.org for potential authors and participants who wish to use this version of GIFT (Ososky, 2016).

Using GIFT Cloud/GIFT Virtual Open Campus to Create an Experiment in the Cloud

GIFT Cloud/GIFT Virtual Open Campus, provides an excellent opportunity to run online experiments using GIFT. If an internet connection is present, participants can be run in person on computers that are connected to the internet. Or completely online participation can occur with participants signing up and completing their participation on their own. The authoring tools in GIFT Cloud/GIFT Virtual Open Campus are nearly identical to those in the desktop version of GIFT. An experimenter can create their course by logging into the GIFT Virtual Open Campus website (https://cloud.gifttutoring.org), and using the GIFT Authoring Tool. If the experimenter prefers to work on his or her local computer, the finished course can be exported and imported into his or her cloud account. After an author is happy with a course that he or she has authored he or she can click on "My Experiments" (See Figure 8). The experimenter can then select a course and make it an active experiment. This creates a copy of the course and provides a web link that can be given to participants. The participants use the link and then run through the course without needing to login to GIFT. This is helpful as it removes the problems with participant number assignment, however, it requires that a survey item be included that asks for the individual's participant number so that the researcher can then match up the information later (the ID numbers on the logs will not necessarily have meaning to the experimenter).

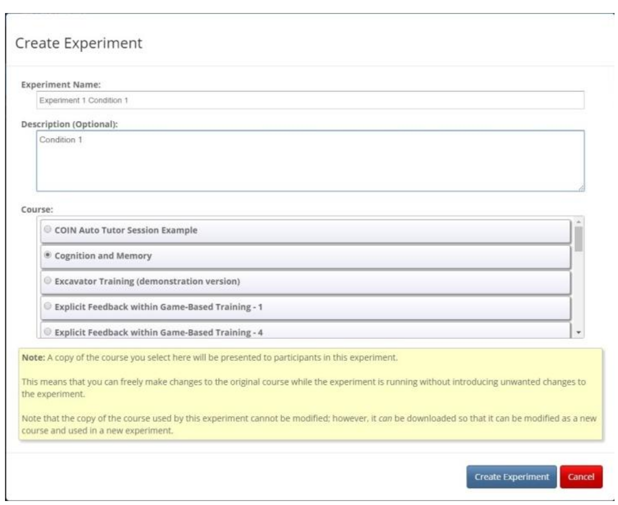


Figure 8. Creating an Experiment from an existing course in the "My Experiments" tab.

Extracting data with GIFT Cloud/GIFT Virtual Open Campus

Experimenters can check on the status of their experiments and view the number of participants that have provided data. If an experimenter wants to view the data it can be exported by clicking "Pause and Report". This will bring up the "Build Experiment Report" interface (See Figure 9). This interface allows the individual to extract the desired data and save it in a .csv file that can be opened with Excel or SPSS. The online extraction tool has a subset of features that are available in the local desktop version of the ERT. Data files (and survey output) can be merged onto a single line for each participant; however, there are not additional data merge options.

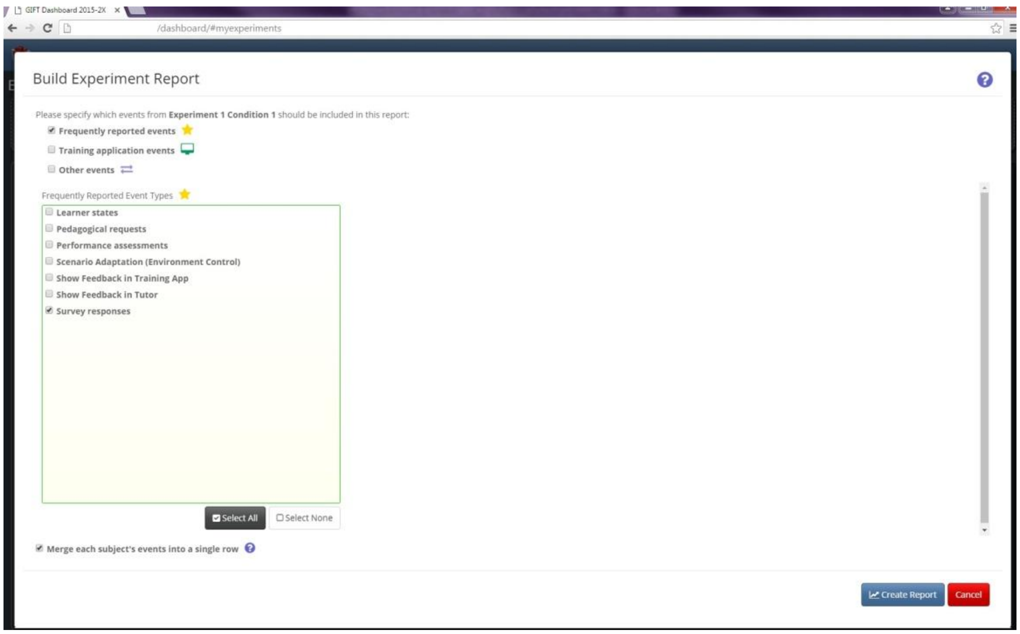


Figure 9. Building an experiment report in GIFT Cloud/GIFT Virtual Open Campus

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

GIFT provides many opportunities for research psychologists to harness it's functionality for their research. At current time experiments can be run with GIFT Local on an individual computer, or online with GIFT Cloud/GIFT Virtual Open Campus. Further, there are a number of different approaches in GIFT to running studies and participant management. For future versions of GIFT, it would be helpful for participant interfaces to be standardized and for clear options to be given to the researcher in regard to establishing participant numbers and user identification. The current versions allow for anonymous data collection, with the option of including a participant number request to be entered in a survey. This is an efficient workaround, but it would be helpful to have the participant number represented in the log file title in order to reduce confusion when trying to match up participant data. Entirely survey based studies can currently be conducted online using GIFT Cloud/GIFT Virtual Open Campus, and if is desired, full courses using connections to Training Applications on the participant's computer (e.g., PowerPoint) can be run on the web.

For in-person studies, using GIFT can greatly reduce the required number of research assistants and provide a means for moving participants through study participation efficiently. A number of the benefits of using GIFT for psychology experiments have been documented in the original Research Psychologist's version of GIFT (Sinatra, 2014). The current paper serves as an update highlighting the new features, changes to features that impact the researcher, and future directions. As GIFT continues to expand it will likely provide more opportunities to researchers who want to use it to conduct their own experiments.

REFERENCES

Adams, D. (1979). The hitchhiker's guide to the galaxy. New York: Pocket Books.

Goldberg, B. & Cannon-Bowers, J. (2015). Feedback source modality effects on training outcomes in a serious game: Pedagogical agents make a difference. Computers in Human Behavior, 52, 1-11.

Ososky, S. (2016). Generalized Intelligent Framework for Tutoring (GIFT) Cloud/Virtual Open Campus Quick-Start Guide (No. ARL-CR-0796). ARMY RESEARCH LAB ABERDEEN PROVING GROUND MD.

Sinatra, A. M. (2014). The research psychologist's guide to GIFT. In Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym2) (p. 85).

Sinatra, A. M., Sims, V. K. & Sottilare, R. A. (2014). The Impact of Need for Cognition and Self-Reference on Tutoring a Deductive Reasoning Skill (No. ARL-TR-6961). ARMY RESEARCH LAB ABERDEEN PROVING GROUND MD.

Sinatra, A.M., Sottilare, R.A. & Sims, V.K. (2016, *in press*). The effects of self-reference and context personalization on task performance during adaptive instruction. Proceedings of the 2016 Human Factors and Ergonomics Society International Meeting.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

ABOUT THE AUTHOR

Dr. Anne M. Sinatra *is an Adaptive Tutoring Scientist at the Army Research Laboratory's SFC Smith Center. Her background is in Cognitive and Human Factors Psychology. She has taught Psychology college courses at both introduction and upper-division levels. She conducts adaptive training research as a member of the Learning in Intelligent Tutoring Environments (LITE) Lab and works on the Generalized Intelligent Framework for Tutoring (GIFT) project.*

Designing the User Experience of the GIFT Cloud authoring tools

Scott Ososky
U.S. Army Research Laboratory (ARL) and Oak Ridge Associated Universities (ORAU)

INTRODUCTION

This paper presents a new user experience for the authoring tools associated with the web-based version of the Generalized Intelligent Framework for Tutoring (GIFT; Sottilare, Brawner, Goldberg & Holden, 2012), known as GIFT Cloud. The new user experience endeavors to provide user interfaces (UI) and interaction paradigms that are more congruent with human mental models of adaptive tutor authoring, compared to the system representation model upon which prior versions of the authoring tools were more closely associated.

Based on a flowchart-interaction metaphor, the new authoring tools will allow authors to take an object oriented approach to sequencing course content. The new design is intended to be learnable and efficient for authors without computer programming or instructional design experience. This current paper will describe the evolution of the GIFT authoring tools, articulate the vision for the new authoring experience, and discuss the principles upon which the new design is based. This work is intended to benefit novice and intermediate users of the GIFT authoring tools, and will be of value to researchers and practitioners interested in developing or managing adaptive training content using GIFT.

GIFT and the GIFT Authoring Tools

GIFT was described as "an empirically-based, service-oriented framework of tools, methods, and standards to make it easier to author computer-based tutoring systems (CBTS), manage instruction and assess the effect of CBTS, components and methodologies" (Sottilare, Graesser, Hu, & Holden, 2013). For more information about GIFT, in general, the reader is encouraged to visit the GIFT project website (http://gifttutoring.org), where papers, proceedings, and eBooks are available in the Documents section. Simultaneously, GIFT is an open-source research project and public-facing application. GIFT is currently under development and includes a number of technologies, features, tools, and methods intended to support a variety of users including instructional designers, authors, instructors, researchers, and learners. It is important that user experiences for GIFT are developed with an awareness of the different skills and backgrounds of its current and future users.

The GIFT Authoring Tool (GAT) is the software interface that enables a GIFT user to build and configure an adaptive tutor. Those tools have undergone significant change since the inception of GIFT, driven by research and usability, respectively. Fundamentally, the GAT is intended to support the following overall goals for GIFT (Sottilare et al. 2013): decrease the effort and skill threshold required for authoring ITSs; support users in organizing their domain content and knowledge; support effective instructional strategies; allow for rapid prototyping of ITSs; leverage standards for integration of external content (media and software applications); and promote content reuse and interoperability through standards. The following sections discuss the evolution of GIFT's authoring interfaces, as well as the background work that served as the catalyst for a new user-centered design experience for the GAT.

BACKGROUND

Evolution of the GIFT Authoring Tools

As GIFT has evolved, so have the authoring tools. Originally, GIFT courses were created exclusively by writing and/or editing schema within a file formatted as eXtensible Markup Language (XML). This required the author to know how to write properly formatted XML code and possess specific knowledge of how the XML code powered a course within the GIFT platform, in addition to general knowledge requirements associated with ITS and adaptive tutoring, in general.

Direct Manipulation

An XML editor was eventually integrated with GIFT to facilitate the creation of the XML files. This editor represented the first steps in assisting users in creating GIFT courses. For instance, the XML editor visually presented the XML code in a manner that was easier for the author to read. Fields within the XML code that the author could edit were highlighted; further, fields that had a known list of options were constrained to a drop-down selection menu. In reducing the opportunity for error, either in coding or in selecting appropriate options, the XML editor also reduced the computer programming burden on the author, with respect to creating tutors for GIFT. However, creating a tutor with that tool still required general knowledge of intelligent tutoring as well as specific knowledge about the GIFT architecture.

Content Organization

An operator control panel was also introduced with the XML editor that allowed an author to quickly access each individual component of the framework. For instance, clicking on the Course Authoring Tool (CAT) button opened the corresponding course XML file within the editor. This UI made each component of GIFT visible and accessible to the user, but did not necessarily suggest the relationships between those components. Some buttons intuitively referenced editors for intelligent tutoring system (ITS) concepts, such as the learner model and pedagogical model. However, other buttons referenced functions that are more specific to GIFT, such as the SIMILE Workbench and the Metadata Authoring Tools. That UI reduced the burden on the author to recall the name and location of each GIFT component, but still required knowledge about the relationship and function of components within the GIFT architecture.

Specialized Interfaces

A standalone Survey Authoring System (SAS) was then developed to help users in the creation, management, and reuse of survey content. The SAS was accessible from the operator control panel, and was the first GIFT authoring UI to appear within a web-browser. The SAS eliminated the need to write XML to generate survey content. Further, by leveraging web-based form elements such as lists and tabs, authors were also able to sort and filter survey elements, organize survey content into containers, and manage question banks for dynamic assessments. This also marked the first time that any course element could be previewed (e.g., questions, surveys) within the same UI that was used for authoring that content. In addition to further reducing the opportunity for authoring error, web-based forms also enabled the ability for error-checking on the fly. For example, the SAS would flag a question for further attention if the author omitted the question text and only wrote out the responses for that item.

Comprehensive Experiences

The current version of the GAT moves away from the control panel and XML editor in favor of a fully browser-based interface. It was designed to be accessible from within GIFT Cloud (see Ososky, Brawner, Goldberg, & Sottilare, 2016, submitted); and, it is easy to switch between the UIs for the GAT, taking courses, and viewing course history, respectively. All elements of GIFT are secured and managed by individual GIFT Accounts, which are available at no-cost. Just as the SAS supported authors in survey creation, the GAT enables the ability to author most course objects in web-based forms. An integrated file browser allows authors to manage system files and organize course content. The GAT also includes some help at the point of need, in the form of tool tips and mouse-over text. Notably, the GAT inherently imposes some structure to the authoring workflow. For example, a survey context (i.e., collection of surveys) must exist before a survey course object can be configured; and, course concepts must be specified within the course's properties before an adaptive course flow course object can be configured.

Opportunities to Improve the GAT

The authoring experience in GIFT has changed significantly since GIFT's initial version. Each version of the authoring experience has served, in part, to reduce the skill barrier required to author a tutor, as well as reduce the time required to create tutors. Naturally, authoring tools that are more accessible allow more users to try GIFT, which, in turn, creates more opportunities to gather feedback and learn how to further improve the authoring experience, particularly for novice and intermediate users. Additionally, GIFT development is largely research driven, suggesting that functionality and architecture will continue to evolve, requiring new user experiences that keep pace with software development.

The control panel was one of the first custom UIs designed specifically for authoring in GIFT. It was during this time that some of the first usability feedback was gathered on the authoring experience. From the small response sample, respondents indicated that they understood the potential of GIFT but found it somewhat difficult to use. Further, some respondents commented that a more efficient method than XML editing for authoring was needed, such as flowchart-based editing (Holden & Alexander, 2015). Following that report, a heuristic evaluation of the control panel and SAS authoring experience was conducted (Ososky & Sottilare, 2016) based on Nielsen's 10 principles for UI design (1995). The recommendations from that evaluation revealed three main areas for improving the authoring experience for novice and intermediate users: interface consistency, user-centered design, and help/support materials.

Consistent and Understandable

The heuristic evaluation revealed an opportunity to improve upon the consistency of elements within the GAT. Consistency, for the purpose of this discussion, applies to labels and names of elements within the platform, as well as the visual appearance of interfaces across the various screens within the GAT. Occasionally there are slight variations in the way in which an element within GIFT is named. For instance, the control panel referred to the Survey Authoring System; the actual SAS interface displayed the title, GIFT Survey System [*sic*]. Those two labels referred to the same UI. Additionally, the visual appearance of the SAS was different than that of the newer web-based GAT.

In addition to being consistent, it is also important that the labels chosen are not rooted in system-level language. Terminology and labels should be human-readable, with intuitive meaning within the context of adaptive tutor authoring. The readability of GIFT information extends to help and error messages as well. Thus, there is an opportunity to review and standardize the language and messages that are used throughout the GIFT user experience.

User-Centered Design

The second recommendation theme from the evaluation was to continue to refine and structure interfaces around authors' needs and goals (the *what*); and to design interactions that align with authors' mental models of the tutor creation process (the *how*). The purpose of those are to increase the learnability and usability of the system, and deliver tools that help authors to meet their overall goals for learning, training, and/or research. That is inherently difficult, because the concept of tutor authoring is a relatively new content creation paradigm and is still evolving within ITS platforms, in general.

User goals are composed of tasks that may be central or auxiliary to GIFT, such as the ability to generate printable versions of surveys for researchers that need to submit materials to an institutional review board (IRB). The consideration of authoring goals should also identify tasks that authors should not need to do. For example, a system file browser in the GAT allows a course author to manage course XML files; however, file management is not necessarily a course authoring goal, and such a function should be relegated to some automatic process within GIFT, by default. Instead of files, authors may benefit from tools that help them manage and organize collections of surveys, courses, or media objects.

Once authoring goals are clearly identified, they can be implemented into the UI in a way that aligns with human mental models. Mental models influence users' expectations regarding a system's functionality and guide user interaction behavior (Ososky, 2013). An individual's mental model regarding a particular system is influenced by past experiences and perceived similarity of other systems to the target system. It is expected that subject matter experts will be able to build tutors using GIFT, without prior knowledge of instructional design, programming or ITS. In the absence of a mental model for ITS authoring in GIFT, users will attempt to leverage a known interaction model of another system, and then test assumptions about the authoring tools based on that model. GIFT, therefore, can become more learnable with interface designs that suggest more accurate interaction metaphors to authors. For example, the linear representation of the current course object list is subjectively similar to the default view in PowerPoint. This analogy is useful in understanding that course objects can be added and removed from the course sequence, and reordered with a drag-and-drop operation. However, GIFT also supports adaptive branching, which is not congruent with that analogy, and not visualized within the current authoring UI. Therefore, it may be more accurate to provide a course authoring UI that evokes the interaction mental model of "flowchart" construction instead of the current "slide deck" metaphor.

Finally, the authoring tools should be flexible to accommodate different workflows. The current GAT is somewhat rigid in the steps required to create a tutor. For instance, a collection of surveys (i.e., survey context) must be designated before configuring the Survey course object. Therefore, it is currently more efficient to create survey content in GIFT before adding survey course objects to the timeline. The opposite is true for media content; it is necessary to add a Media course object to the course flow before media content can be uploaded to GIFT Cloud. Accommodating workflow preferences are another way in which mental model based expectations are met; and that also provides the foundation for future GAT enhancements, such as collaborative tutor design by teams of authoring specialists.

Support and Documentation

User support was the third recommendation theme from the heuristic evaluation of the GIFT authoring tools. One point of feedback, particularly from new GIFT users, is that the authoring tools do not necessarily make clear to the author the order in which tasks need to be completed to build an adaptive tutor. It is like trying to assemble furniture without an instruction manual. The builder understands the goal, but is not sure how to go about accomplishing it by only looking at the parts and tools provided. Documentation is one way to address the support theme, but system developers should not count on users to read all or any of the documentation before exploring the software. As in interim step, a GIFT Quick

Start guide was recently developed, which is intended to quickly provide an overview of GIFT's current features and terminology, including those of the authoring tools (Ososky, 2016).

In practice, the notion of support extends far beyond documentation. For instance, the authoring UI can be made more intuitive by adding elements that call the author's attention to what needs to be done first, or next. Configurable elements can be enhanced with in-line tips, plus automatic error checking of the author's inputs. Once the core UI is established, it could be further enhanced for novice users with the addition of dialogs (or wizards) that ask a series of questions to help authors to select the correct configuration options. However, wizards are not a panacea for user experience design; the author should still be able to review the result of a wizard in the primary UI to ensure the output is exactly what was desired and/or make further adjustments to the configuration of a course object.

Many of the topics identified in the heuristic evaluation were addressed with the introduction of GIFT Cloud (Ososky et al., 2016, submitted) and the web-based GAT. The efforts described in the following section, represents a continuation of that work toward an authoring user experience that is both efficient and usable, while retaining the functionality of GIFT that distinguishes it from other computer-based training platforms and intelligent tutoring systems, respectively.

DESIGNING A NEW USER EXPERIENCE FOR AUTHORING

The roadmap for upcoming user experience improvements to the GIFT Authoring Tools (GAT) is informed by the opportunities described in the previous section: consistency, user-centered design, and support. The new UI is also inspired by UX design principles found in Nielsen (1995) and Cooper et al. (2007), development-tool interface design considerations in Lightbown (2015), and UX research guidance provided in Buley (2013), as well as numerous UX blogs and related communities of practice. Specifically, the near-term updates will address three main improvements to the GIFT authoring user experience. Please note that the information in this section represents work-in-progress. The final design of various GAT UI and elements may change as they are being built and tested. The underlying design motivations should remain consistent in moving from conceptual prototype to live implementation.

Standardized Terminology

In an effort to increase the learnability of GIFT, the development team examined the terminology within the GAT. Terms that were identified to have been rooted in the system architecture of GIFT were revised for clarity. Further, the names of some elements were updated or created in order to keep pace with the long-term development trajectory of GIFT. The following GAT terminology is proposed (**Table 12**).

Table 12. Proposed updated and new terminology related to GIFT course authoring

Term	Description (and/or former terminology)
GIFT Cloud	The web-based version of GIFT located at cloud.gifttutoring.org
GIFT Local	The downloadable of version of GIFT for offline use
Course Objects	Elements that make up the sequencing or configuration of GIFT courses; previously *transitions;* individual [1]**Course Objects** listed below
Information[1]	Simple course object that displays text to the learner; previously *guidance*
Media[1]	Used for display of media, documents and/or web based resources; previously *lesson material*
Adaptive courseflow[1]	The outer-loop tutoring / pedagogical engine in a GIFT course; previously referred to as *MBP or Merrill's Branch Point*

Term	Description (and/or former terminology)
Branch[1]	[new] A manual course branch based on the Adaptive Courseflow object; intended for direct control of course logic
Real-time assessment[1]	The inner-loop tutoring engine, typically used with an External Application; previously referred to as *DKF* or *Domain Knowledge File*
External application[1]	Connects GIFT with outside applications for training or presentation of media, including VBS and PowerPoint; previously *training application*
Structured review[1]	This object provides for review of results from Survey and Real-Time Assessment Course Objects; previously *AAR (After Action Review)*
Survey[1]	Interactive course objects that either assess knowledge or collect information about the learner; [2]**special survey types** listed below
Dynamic assessment[2]	Type of GIFT Survey generated from question banks by learning concept
Conversation tree[2]	[new] Type of GIFT Survey that allows learner to engage in a chat with GIFT, governed by a pre-configured branching dialogue structure
Auto-Tutor conversation[2]	Type of GIFT Survey providing dynamic conversation with an agent created with the Auto-Tutor Script Authoring Tool (a plug-in/extension of the GAT)

Workspace Composition

A unified workspace is at the foundation of the new GAT, from which all elements and functions of the GAT are accessible (**Figure 26**). The goal is to allow users to quickly move between elements without needing to manage multiple windows or switch between visually distinct UIs within the browser-based interface. Complementary to this, menu bar clutter and white-space padding will also be reduced, providing the author with more on-screen space (e.g., Course Flow) in which to work and visualize the sequence of course elements.

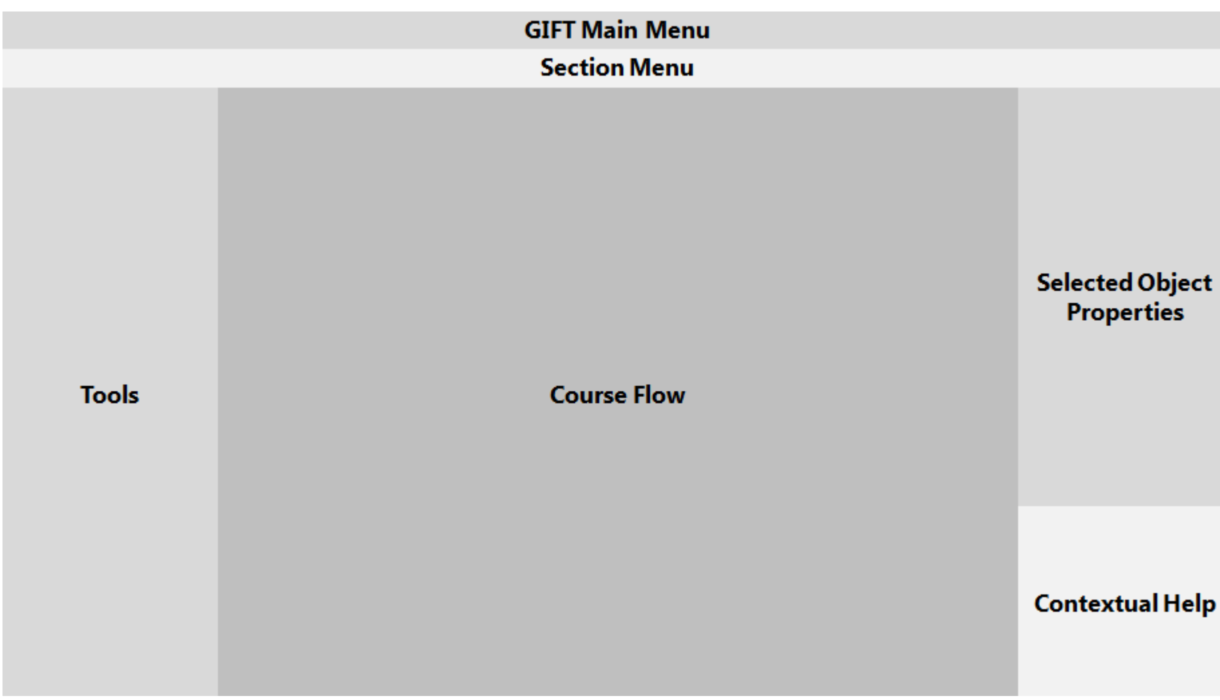


Figure 26. Wireframe depicting relative position and size of primary authoring elements

Visual Course Authoring

A near-term update to the GIFT authoring tools will introduce a more intuitive way to sequence courses in GIFT using a redesigned visual interface (**Figure 27**). The interaction is based on a familiar flowchart metaphor, with the ability to drag and drop objects from the course object panel onto the course structure. This provides a number of benefits over the previous design (i.e., text-based list). With a flowchart style representation of the course, it will be possible to quickly recognize all of the possible paths a learner might take in the course, with respect to course branching based on the learner model. The course object elements in a given course-flow will be designed with dynamically updating information about their contents at-a-glance, allowing for faster and more efficient editing and sequencing of course objects.

The course-flow view is intended to be self organizing such that the spacing between course objects will dynamically adjust when objects are added, removed, or re-ordered. The timeline itself will also dynamically adjust its size relative to the number of course objects that are present. The flowchart view of the course is intended to accommodate future enhancements to the GAT. These might include nodes on each of the course object boxes that represent the number of available connection points leading in to and out of each course object. That would allow an author to quickly identify the number of conditional branches available for a course object as well as the unconnected nodes (i.e., dead ends) requiring further attention. Some elements, like the Real-Time Assessment course object, are intended to operate in conjunction with other elements, such as the External Application course object. The course flow interface can aid the author in identifying when those advanced features are enabled and/or required within their courses.

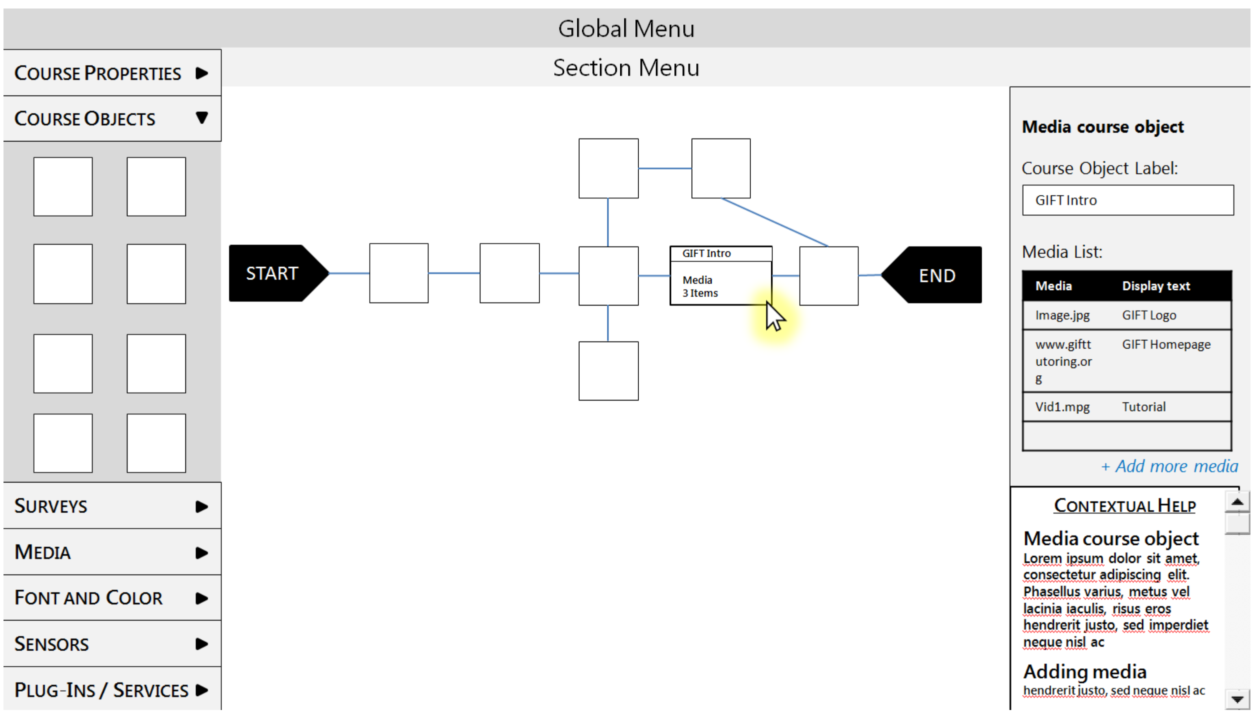


Figure 27. Conceptual interface prototype depicting drag-and-drop of course objects from the tool bar (left) to the course timeline (center), plus configuration and help for activated course object (right). Note: course object icon design not shown in this prototype

Organization of Content and Authoring Workflow

The prototype (Figure 2) illustrates the effort to design new UI around user goals and accommodate different workflows for authoring a tutor. Some of the most common and important tasks in tutor authoring are sequencing course objects, adding media to a course, and adding survey content to a course. Therefore, those are some of the elements that are most prominently featured in the prototype UI. By comparison, the file explorer in the current web-based GAT is absent from this design. Authors may want to organize media and survey content, respectively, but the goal of organization may be better served within each of those sub-menus, using labels and folders instead of a file system explorer UI.

The prototype UI is also intended to provide greater flexibility in course authoring with respect to workflow. In the new GAT UI, for instance, authors would be able to choose whether to first upload/create course content, or sequence the course with course objects on the timeline (i.e., outline the course first). Thus, the new GAT UI is being developed with workflow flexibility in mind, which can also include multiple ways to build courses and interact with course content.

Help at the Point of Need

The new UI is intended to aid authors in information recall through a new contextual help panel. Its content (Figure 2) is expected to update depending upon the most recent user interaction. For example, clicking on a course object will display information about that object and how to configure it. Links within the contextual help panel will lead to expanded information on the current and related topics within a separate tab or window, should the author choose to seek more information. Additional help and support resources (e.g., videos, tutorials) are in the planning stages for future GIFT Cloud updates.

Interacting with Survey Content

Surveys are a special type of course content in GIFT. Surveys can be used as a static data collection mechanism, with the results of surveys simply being written to an output file. However, surveys can also be used in order to assess the learner’s knowledge, states, or traits; and then GIFT can then leverage that information for tutor adaption and dynamic presentation of course content. Surveys are thus made up of what the learner sees (i.e., questions and responses) as well as the hidden logic that dictates how a survey is coded and/or scored, then written to the learner record. The original SAS supports authors in managing survey content, but the UI has become inconsistent and less efficient in comparison to more recent UI development in the GAT and GIFT Cloud overall. The prototype survey authoring UI proposes to fully integrate survey management into the new GAT (Figure 3).

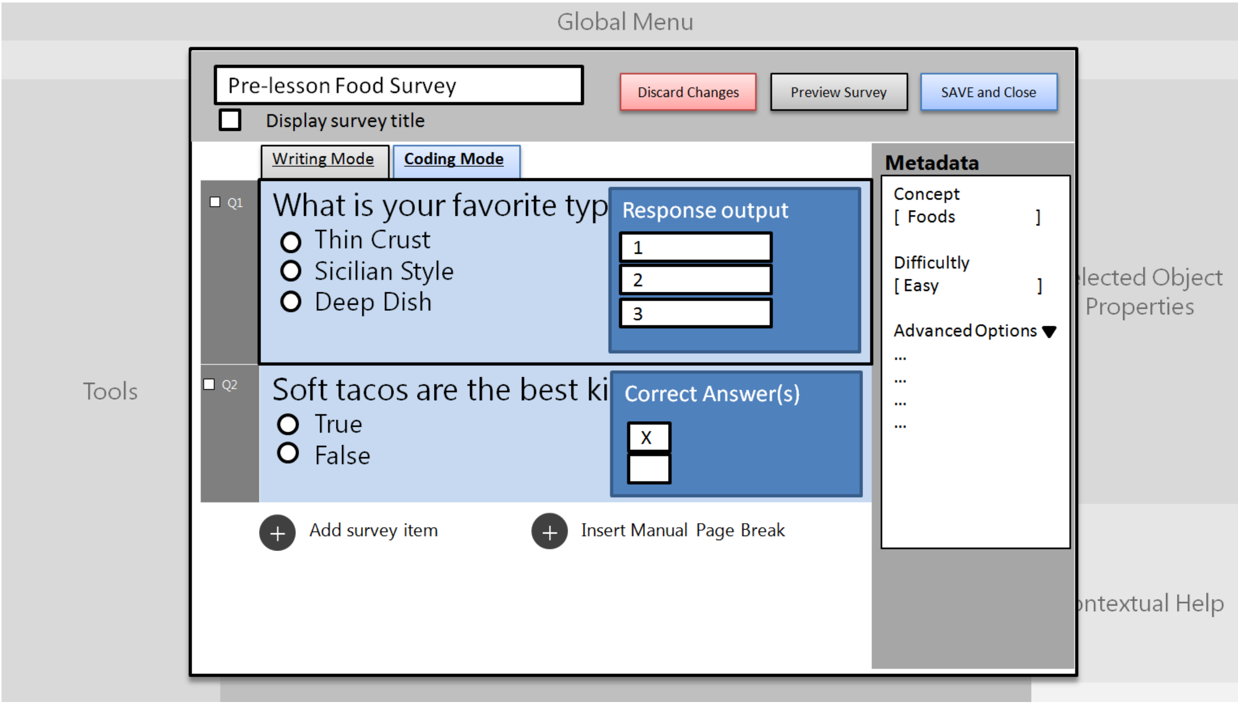


Figure 28. Survey authoring within the GAT UI, with overlays for scoring / coding displayed

The new survey UI allows authors to efficiently manage survey content directly within the new GAT. It is designed to be accessed by clicking on a Survey course object, or via the Survey menu on the left-hand Tools panel within the primary GAT interface. This survey UI is intended to make survey creation more efficient by utilizing a WYSIWYG (what-you-see-is-what-you-get) design, allowing for faster previewing of surveys and survey items. The design prototype also proposes a survey "coding mode". Activating this mode would present an overlay on the survey for easy coding / scoring of survey responses. Survey options in the side menu would also change to reflect the current mode, reducing cognitive load in both the writing and coding modes. Overall, the new survey UI design is intended to reduce the amount of physical effort (i.e., excise) and time required to create and manage surveys.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

A Long-Term Vision for the GIFT Authoring Experience

The planned GAT UI improvements represent the near-term steps toward making adaptive tutor authoring accessible to users without computer programming or instructional design knowledge. The current work also provides the foundation for additional enhancements (**Figure 29**), offering greater flexibility and efficiency to authors, and reducing the initial skill barrier to tutor authoring for beginner users.

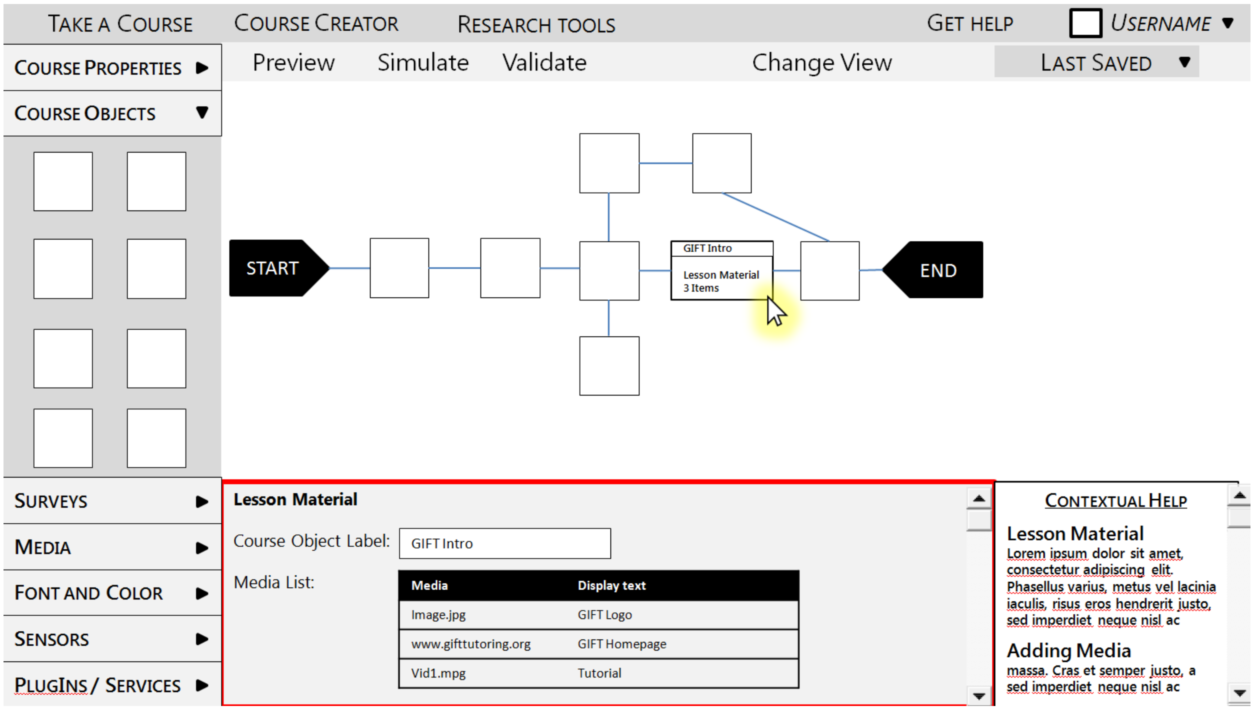


Figure 29. Expanded prototype for new GAT UI including menu bar options and configurable workspaces

The UI could provide additional flexibility to accommodate authors' preference or various screen sizes. In the figure above, the object properties frame appears at the bottom of the screen, rather than the right side, to provide more horizontal space for course flow management. Eventually, it may also be possible to toggle hide/show for the tool pane (left), help (bottom-right) or object properties (bottom-left), respectively, depending on which tasks need more attention. Those panes/frames might also be resizable. The ability to pop-out object properties in a full-screen view for editing or inspection is also being considered.

Future versions of the GAT can expand on the design of the visual course representation. The new course view could represent one level of detail in an overall project hierarchy, from which an author can *zoom in* and *out* of views in the center panel to work on their project at their desired level of detail. For instance, courses with many objects can be organized into subroutines, giving an author the ability to zoom-in to a subroutine or individual course object. Zooming-out, courses might be organized as a series of *lessons*, and then groups of courses would comprise a *project*. That hierarchy would also have implications for sharing and collaborative authoring. This hierarchy interaction concept is currently being explored.

GIFT already supports course validation and preview. Though, it would also be useful to provide faster access to course preview from the within the GAT. Course preview functionality could include the ability to take the course, or to start somewhere in the middle of the course based on a simulated learner model. Taking this idea a step further, the ability to quickly simulate many runs of a course might also be useful for testing and quality assurance purposes. This would aid instructors and researchers to ensure their courses are generating the data outputs needed and course adaptations are functioning as intended.

Finally, a fully realized authoring UI provides the foundation upon which enhanced support materials can be created. This might take the form of tutorial videos that demonstrate how to accomplish tasks. Robust wizards could be offered to aid novice authors or for rapid generation of content; and course *templates* based on instructional theory and best practices can be developed that would serve a similar purpose. Such features ultimately serve the larger goal of creating positive user experiences for authors in GIFT.

Future Research Activities

GIFT is and continues to be informed by feedback from the user community. We will continue to conduct research activities to generate data to further inform GAT design and UX improvements. We also intend to generate benchmark data on authoring efficiency for evaluation against new versions of the GAT. We are also investigating processes and technology that will allow us to more quickly identify issues, design improvement, and implement new features that are informed by authors' feedback.

Complementary to user research, technological research should endeavor to provide quality of life improvements to the authoring user experience. Specifically, efforts should continue to pursue the automation of potentially complex and/or labor intensive processes within GIFT. Some of those processes are centered on the management of learning content, specifically: deconstructing media content for optimal delivery, tagging content with the appropriate metadata, and creating iterations of content objects appropriate for adaptive presentation.

We are committed to maintaining the quality and integrity of the GAT user experience as new features and functionality are integrated into the base system. We look forward to sharing these new experiences with you, the GIFT community, in the near-future, and look forward to hearing your feedback.

REFERENCES

Buley, L. (2013). *The user experience team of one*: Rosenfeld Media.

Cooper, A., Reimann, R. & Cronin, D. (2007). *About Face 3: The Essentials of Interaction Design*. Indianapolis, IN: John Wiley & Sons.

Holden, H. & Alexander, M. (2015). An analysis of GIFT user experiences and design recommendations for enhancing GIFT's usability. In R. A. Sottilare (Ed.), *Proceedings of the Second Annual GIFT Users Symposium* (pp. 83-89). Orlando, FL: U.S. Army Research Laboratory.

Lightbown, D. (2015). *Designing the User Experience of Game Development Tools*. Boca Raton, FL: CRC Press.

Nielsen, J. (1995). 10 Usability Heuristics for User Interface Design Retrieved 27-Feb-2015, from http://www.nngroup.com/articles/ten-usability-heuristics/

Ososky, S. (2013). *Influence of task-role mental models on human interpretation of robot motion behavior.* (Doctoral Dissertation). University of Central Florida, Orlando, FL.

Ososky, S. (2016). *Generalized Intelligent Framework for Tutoring (GIFT) Cloud / Virtual Open Campus quick start guide*. (ARL-CR-0796). Orlando, FL: US Army Research Laboratory.

Ososky, S., Brawner, K., Goldberg, B. & Sottilare, R. A. (2016, submitted). GIFT Cloud: Improving usability of adaptive tutor authoring tools within a web-based application. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*.

Ososky, S. & Sottilare, R. A. (2016). *Heuristic evaluation of GIFT authoring tools*. (ARL-SR-0353). Orlando, FL: US Army Research Laboratory.

Sottilare, R.A., Brawner, K.W., Goldberg, B.S. & Holden, H.K. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). Concept paper released as part of GIFT software documentation. Orlando, FL: U.S. Army Research Laboratory – Human Research & Engineering Directorate (ARL-HRED). Retrieved from: https://gifttutoring.org/attachments/152/GIFTDescription_0.pdf

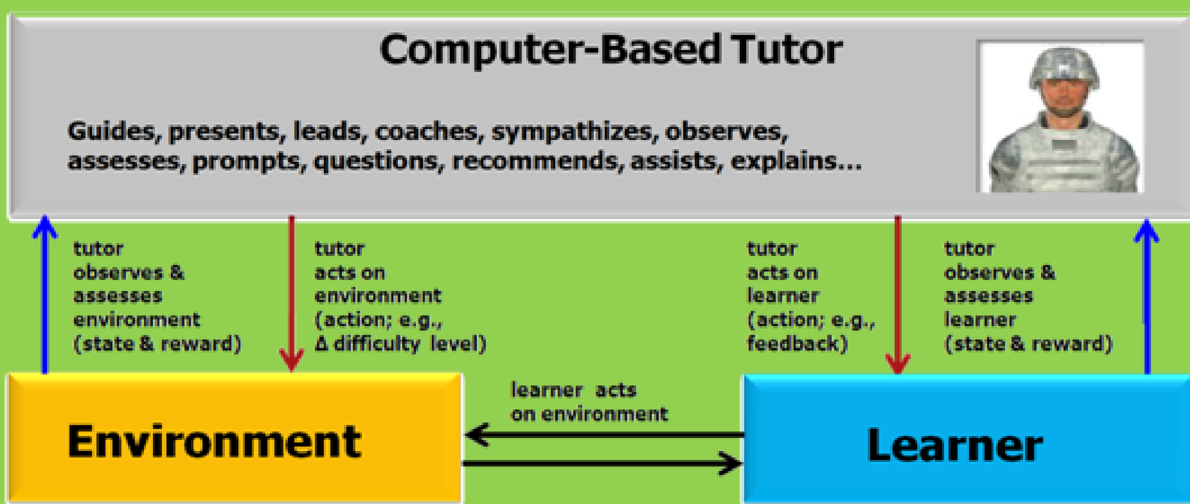
Sottilare, R. A., Graesser, A. C., Hu, X. & Holden, H. (2013). *Design Recommendations for Intelligent Tutoring Systems: Volume 1 Learner Modeling*. Orlando: U.S. Army Research Laboratory.

ABOUT THE AUTHOR

Dr. Scott Ososky *is a Postdoctoral Research Scientist within the US Army Research Laboratory's Human Research and Engineering Directorate (ARL-HRED). His current research examines mental models of adaptive tutor authoring, including user experience issues related to development tools and interfaces within the adaptive tutor authoring workflow. He has also published numerous works regarding human interaction with near-futuristic, intelligent robotic teammates. Dr. Ososky received his Ph.D. and M.S. in Modeling & Simulation, as well as a B.S. in Management Information Systems, from the University of Central Florida.*

Proceedings of the Fourth Annual GIFT Users Symposium

GIFT, the Generalized Intelligent Framework for Tutoring, is a modular, service-oriented architecture developed to lower the skills and time needed to author effective adaptive instruction. Design goals for GIFT also include capturing best instructional practices, promoting standardization and reuse for adaptive instructional content and methods, and methods for evaluating the effectiveness of tutoring technologies. Truly adaptive systems make intelligent (optimal) decisions about tailoring instruction in real-time and make these decisions based on information about the learner and conditions in the instructional environment.



The GIFT Users Symposia were started in 2013 to capture successful implementations of GIFT from the user community and to share recommendations leading to more useful capabilities for GIFT authors, researchers, and learners.

About the Editors:

Dr. Robert Sottilare *leads adaptive training research within US Army Research Laboratory's Learning in Intelligent Tutoring Environments (LITE) Lab in Orlando Florida. He is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT).*

Dr. Scott Ososky *is a Postdoctoral Research Scientist within the US Army Research Laboratory's Human Research and Engineering Directorate (ARL-HRED). His current research examines mental models of adaptive tutor authoring, including user experience issues related to development tools and interfaces for tutor authoring.*

Part of the Adaptive Tutoring Series

978-0-9977257-0-4