Proceedings of the Ninth Annual GIFT Users Symposium

May 2021
Orlando, Florida

Edited by:
Benjamin S. Goldberg

Part of the Adaptive Tutoring Series

Proceedings of the 9th Annual GIFT Users Symposium (GIFTSym9)

Proceedings of the 9th Annual
Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym9)

Edited by:
Benjamin Goldberg

The views expressed herein are those of the authors and do not necessarily reflect the views of the U.S. Army Combat Capabilities Development Command – Soldier Center

Use of trade names or names of commercial sources is for information only and does not imply endorsement by the U.S. Army Combat Capabilities Development Command – Soldier Center.

This publication is intended to provide accurate information regarding the subject matter addressed herein. The information in this publication is subject to change at any time without notice. The U.S. Army Combat Capabilities Development Command – Soldier Center, nor the authors of the publication, makes any guarantees or warranties concerning the information contained herein.

Printed in the United States of America
First Printing, May 2021

U.S. Army Combat Capabilities Development Command
Soldier Center
Orlando, Florida

International Standard Book Number: 978-0-9977257-9-7

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

Contents

From the Editor

GIFT is a free, modular, open-source tutoring architecture that is being developed to capture best tutoring practices and support rapid authoring, reuse and interoperability of Adaptive Instructional Systems (AIS). The authoring tools have been designed to lower costs and entry skills needed to author AISs and our research continues to seek and discover ways to enhance the adaptiveness of AISs to support self-regulated learning and competency development within a learning ecosystem.

This year marks the ninth year of the GIFT Symposia and we accepted 18 papers for publication. None of this could happen without the efforts of a fantastic team. Our program committee this year did an outstanding job organizing and reviewing, and we want to recognize them for their efforts.

- **Elyse Burmester**
- **Keith Brawner**
- **Jeanine DeFalco**
- **Greg Goodwin**
- **Michael Hoffman**
- **Anne Sinatra**
- **Joan Johnston**
- **Rodney Long**

We are proud of what we have been able to accomplish with the help of our user community. This is the eighth 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 has well over 2,100 users in over 103 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 editor and program committee wishes 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 AIS community and across the training and education enterprise.

GIFT and the GIFT Symposium has taken a broader perspective as the Army moves forward on modernizing their technology-driven training strategies through the Synthetic Training Environment and (STE) and Army Learning Ecosystem Concept (ALEC) 2035.

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 feedback on GIFT and influence its future development.

Many thanks to all GIFT users…

Ben

Benjamin Goldberg, Ph.D.
GIFTSym8 Chair and Proceedings Editor

Theme I: New GIFT Features and Guides

The GIFT Architecture and Features Update: 2021 Edition

Michael Hoffman[1], Benjamin Goldberg[2], Keith Brawner[2]
Dignitas Technologies[1], U.S. Army Combat Capability Development Command – Soldier Center[2]

INTRODUCTION

The first version of the Generalized Intelligent Framework for Tutoring (GIFT) was released to the public in May of 2012. 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 current paper continues the con- versation with the GIFT user community in regards to the architectural "behind the scenes" work and how the GIFT project is addressing user requirements suggested in the previous GIFTSym8 proceedings. The development team takes comments within the symposium seriously, and this paper serves to address requirements from prior years.

As a follow up to the previous GIFT Symposium architecture updates (Brawner & Ososky, 2015; Ososky & Brawner, 2016; Brawner, Heylmun, & Hoffman, 2017; Brawner & Hoffman, 2018; Brawner, Hoffman, Nye & Meyer, 2019; Goldberg, Brawner & Hoffman, 2020) this version highlights new tools and feature requests accomplished over the latest development cycle. The feature requests and derived architectural improvements are derived from two primary sources: (1) symposium paper recommendations collected across the GIFT user base, and (2) stakeholder interactions linked to capability and project needs. The features are organized into logical sections within this update and cover modifications across all core modules operating within GIFT.

WELCOME

First, to the new members of the GIFT community and new GIFT users – Welcome! There are a number of recommended resources that will help to orient you to this project and ecosystem. GIFT has come a long way since its original goals were defined in its description paper (Sottilare, Brawner, Goldberg, & Holden, 2012). First, we would encourage you to simply get started, as the tools and example courses have been designed to assist users in exploring GIFT's tools and methods for the purpose of creating Adaptive Instructional Systems

If you struggle with any individual aspect of the system, the team has produced short "how to" videos to help around the sticking points. There are now many videos available on the GIFT YouTube channel, which is the first result if you search "Generalized Intelligent Framework for Tutoring Youtube" on Google. The YouTube videos have not been updated for the new release; however, the vast majority of the GIFT challenges and authoring has remained unchanged.

Outside of the introductory materials and tutorials available in GIFT, there is also developer support through detailed documentation and active help forums. The GIFT user community is also invited to ask questions and share your 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 1500 postings and responses.

Documentation has been made freely available online at https://gifttutoring.org/projects/gift/wiki/Documentation, with interface control documentation https://gifttutoring.org/projects/gift/wiki/Interface_Control_Document_2021-1, and a developer guide https://gifttutoring.org/projects/gift/wiki/Developer_Guide_2021-1. These documents are updated each software release. In this release, we would also like to highlight the available instructions for hosting your own AWS instance (https://gifttutoring.org/projects/gift/wiki/Amazon_Web_Service_Install_Instructions).

GIFT Development and Release Strategy

There are two GIFT instances available to everyday users, GIFT Cloud and GIFT Desktop. GIFT Cloud follows an every-Friday system update schedule when relevant updates are ready from the engineering team. For the desktop version, we have maintained a 12 month release cycle with a recent regression tested released in April 2021. To support experimentation, intermittent extensions of the core GIFT baseline are performed to facilitate data and interaction requirements based on specific research questions of interest. These are performed on a "as needed" basis, and often serve at the feature extensions included in the next public-release. In the upcoming cycle, there will be a focus on providing one or more examples of team training and collaborative learning that can be executed on GIFT Cloud rather than having to acquire applications such as Virtual Battlespace (VBS) through a Distribution Agreement process. Adjustments to the release strategy will be considered as more agile software development approaches are being applied at the organizational and enterprise level. As a member of the community, if you see a feature in the cloud release which you would like to use locally, simply ask.

GIFT Cloud General Reporting

GIFT Cloud (see Figure 1) has been running continuously for the last five years over Amazon Web Ser-vices. The cloud instance is kept online and updated in advance of the downloadable version, meaning that cloud content must be backwards-ported to be compatible with the perpetually out of date offline version. We do our best to keep the downloadable version to regularly scheduled improvements, but, for ordinary users, we would encourage you to use the Cloud version. It supports hundreds of simultaneous users for experiments. We are generally confident in the systems' ability to stay up and cope with demand. The current limitations are that team training in a virtual environment and sensor-based interactions are not supported on the cloud instance, but that requirement will be addressed.

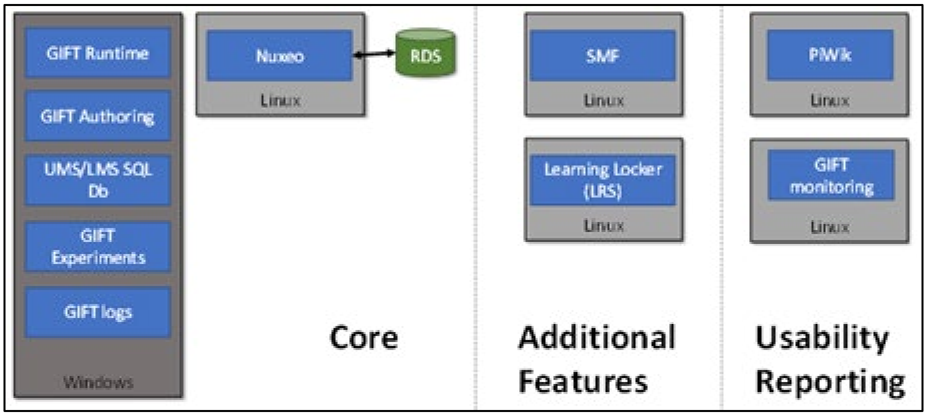


Fig. 1. Simple Diagram Overview of GIFT Cloud Items

Behind the scenes, however, the re-tooling to move to a deployment version of development in a desktop instance to a cloud environment in production has been working well. For the remainder of the paper, we will cover the latest improvements added over the last development cycle.

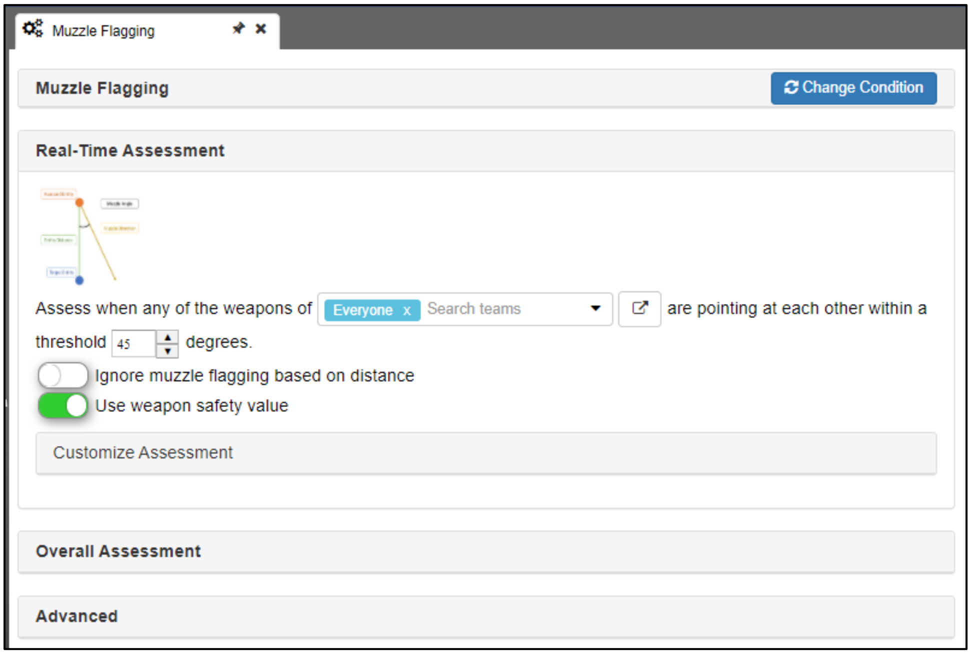
NEW GIFT Features and updates

Since the last feature update from GIFT Sym 8 (Goldberg, B., Brawner, K., & Hoffman, M), there have been multiple additions to the GIFT capability set. Each tool or method described in this section is now available in the latest public-facing open source version of GIFT. Each new feature will be presented with information on the functions it supports and the system and data level dependencies to implement.

New Conditions for Automated Assessments and Data Capture

One of the most powerful features of GIFT is the ability to automate real time assessments across an array of supported training and simulation-based environments. Currently the majority of these assessments take place in condition classes written in Java. In an effort to provide more automated assessments, five new conditions classes were added in the latest version of GIFT.

Muzzle flagging



Muzzle Flagging

Muzzle Flagging

Change Condition

Real-Time Assessment

Assess when any of the weapons of Everyone x Search teams ▾ are pointing at each other within a threshold 45 degrees.

○ Ignore muzzle flagging based on distance

● Use weapon safety value

Customize Assessment

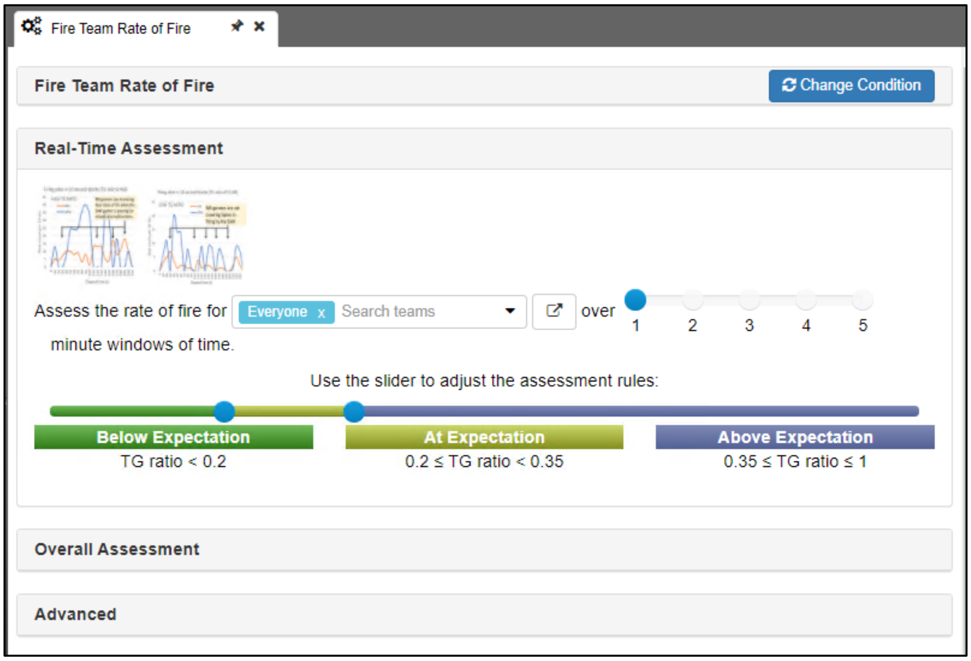
Overall Assessment

Advanced

Fig. 2. Screenshot of the Muzzle Flagging condition authoring user interface in the GIFT course creator.

The Muzzle flagging condition (see Figure 2) is used to assess whether learners are avoiding potential fratricide by not aiming their weapons at other identified friendly forces. This condition requires at least two team members defined as the actors to be assessed for violations. An assessment such as this was designed for a use case of small teams, but can be extended into mounted operations. The author has the ability to customize the angle that determines whether the learner is flagging their buddy or not. A larger value can result in more Below Expectation assessments. You can also apply additional criteria to ignore flagging based on distance or if the weapon safety is enabled. The weapon safety information is retrieved by querying the training application (e.g. VBS) for that information during scenario execution.

Fire team rate of fire



Fire Team Rate of Fire

Assess the rate of fire for Everyone x Search teams over 1 2 3 4 5 minute windows of time.

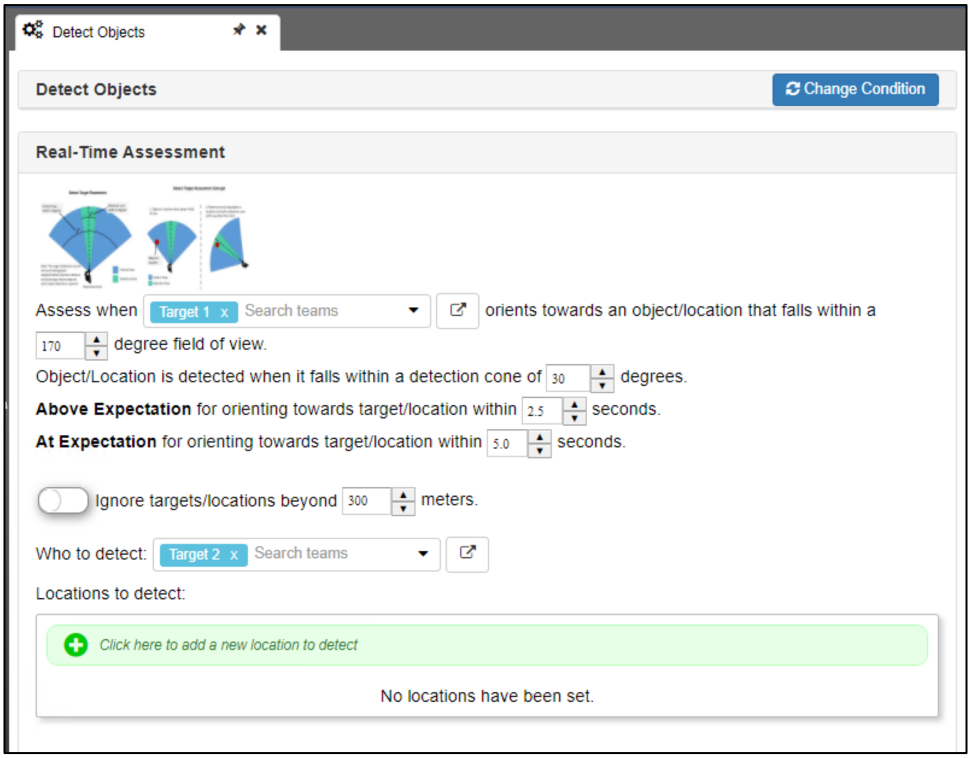
Use the slider to adjust the assessment rules:

Below Expectation	At Expectation	Above Expectation
TG ratio < 0.2	0.2 ≤ TG ratio < 0.35	0.35 ≤ TG ratio ≤ 1

Fig. 3. Screenshot of the Fire Team Rate of Fire condition authoring user interface in the GIFT course creator.

The Fire Team Rate of Fire condition (see Figure 3) is used to assess whether a fire team echelon sized unit is maintaining an appropriate level of suppressive fire over a specified period of time. This condition can be used in situations where a distributed ratio of weapon fire should happen to prevent the enemy from maneuvering. While this is happening, there might be instances where a weapon system might be inoperable due to a jam or reloading. In order for the appropriate rate of fire to be maintained, the other team members must increase their rate of fire. This is normally a coordinated effort involving leadership and communication skills. The condition provides a mechanism to assess the underlying behaviors and diagnostics of cooperation and performance across the team.

Detect objects



Detect Objects

Real-Time Assessment

Assess when Target 1 x Search teams orients towards an object/location that falls within a 170 degree field of view.

Object/Location is detected when it falls within a detection cone of 30 degrees.

Above Expectation for orienting towards target/location within 2.5 seconds.

At Expectation for orienting towards target/location within 5.0 seconds.

Ignore targets/locations beyond 300 meters.

Who to detect: Target 2 x Search teams

Locations to detect:

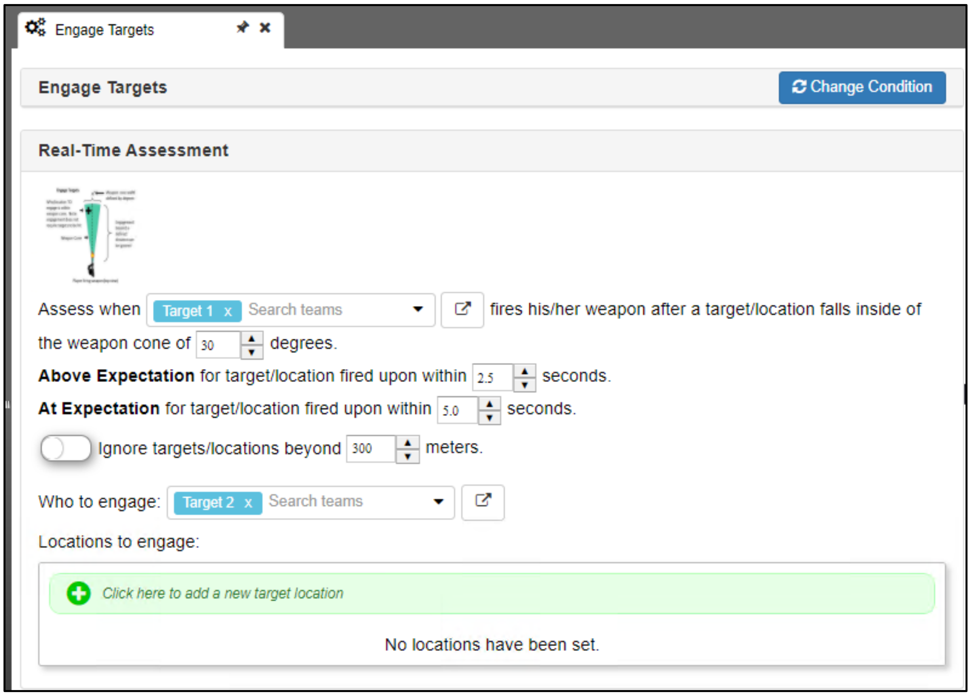
Click here to add a new location to detect

No locations have been set.

Fig. 4. Screenshot of the Detect Objects condition authoring user interface in the GIFT course creator.

The Detect objects condition (see Figure 4) is used to assess whether an object such as another entity or static location was detected when within an individual's perceived field of view. To be considered detected, the object must first enter the field of view of a team member and then that team member must face that object. The author can define the field of view and detection cone, as well as the assessment levels to apply based on the time it takes to orient toward the object once the object is in the field of view. This condition is useful when scanning an area to maintain situational awareness for example. Ideally, this condition class will be extended to include eye tracking oriented data to get a better estimate of what an individual is truly perceiving within the interacting environment.

Engage targets



Engage Targets

Engage Targets

Change Condition

Real-Time Assessment

Assess when Target 1 x Search teams fires his/her weapon after a target/location falls inside of the weapon cone of 30 degrees.

Above Expectation for target/location fired upon within 2.5 seconds.

At Expectation for target/location fired upon within 5.0 seconds.

Ignore targets/locations beyond 300 meters.

Who to engage: Target 2 x Search teams

Locations to engage:

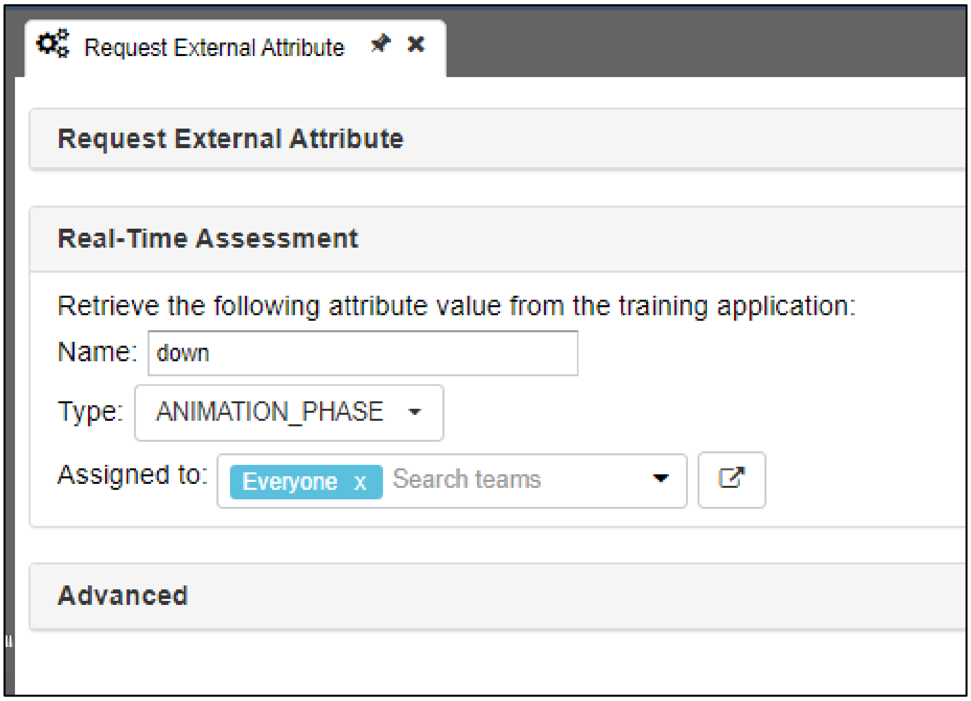
Click here to add a new target location

No locations have been set.

Fig. 5. Screenshot of the Engage Targets condition authoring user interface in the GIFT course creator.

The Engage Targets condition (see Figure 5) is used to assess whether an object such as another entity or static location is fired upon once in view – answering two questions, “are they firing at this thing?” and “is it well-aimed fire?”. The author can define the weapon cone as well as the assessment levels to apply based on the time it takes to fire the learner’s weapon once the object is in that weapon cone. Prior to this condition, in the VBS environment GIFT depended on DIS damage and health information to see if a shot hit anything. This extended logic now allows GIFT users to determine if a shot is close-enough to an entity to be considered well-aimed. This condition is useful when engaging a target requires quick and decisive action once that object is oriented towards, and can be expanded into any “main player sees something and should act upon it” contextual situation.

Request external attribute



Request External Attribute

Request External Attribute

Real-Time Assessment

Retrieve the following attribute value from the training application:

Name: down

Type: ANIMATION_PHASE

Assigned to: Everyone x Search teams

Advanced

Fig. 6. Screenshot of the Request External Attribute condition authoring user interface in the GIFT course creator.

The Request External Attribute condition (see Figure 6) is used to request additional information from the training application that is running the scenario while GIFT is performing real time assessment. The information requested is not provided in the default data stream from that application. One use case for this condition is when GIFT is being used to collect data prior to fleshing out a complete DKF. In this instance additional information may be needed to conduct post analysis modeling and to develop additional assessment logic later on or for playback purposes. For example, you might not know the exact tasks, concepts, conditions you want to use but you know you need to collect DIS traffic and whether a billboard style target is currently exposed or hidden. In other words, this condition can be used to request other information from the simulation that GIFT knows is accessible at runtime.

Adaptive Learning Service API

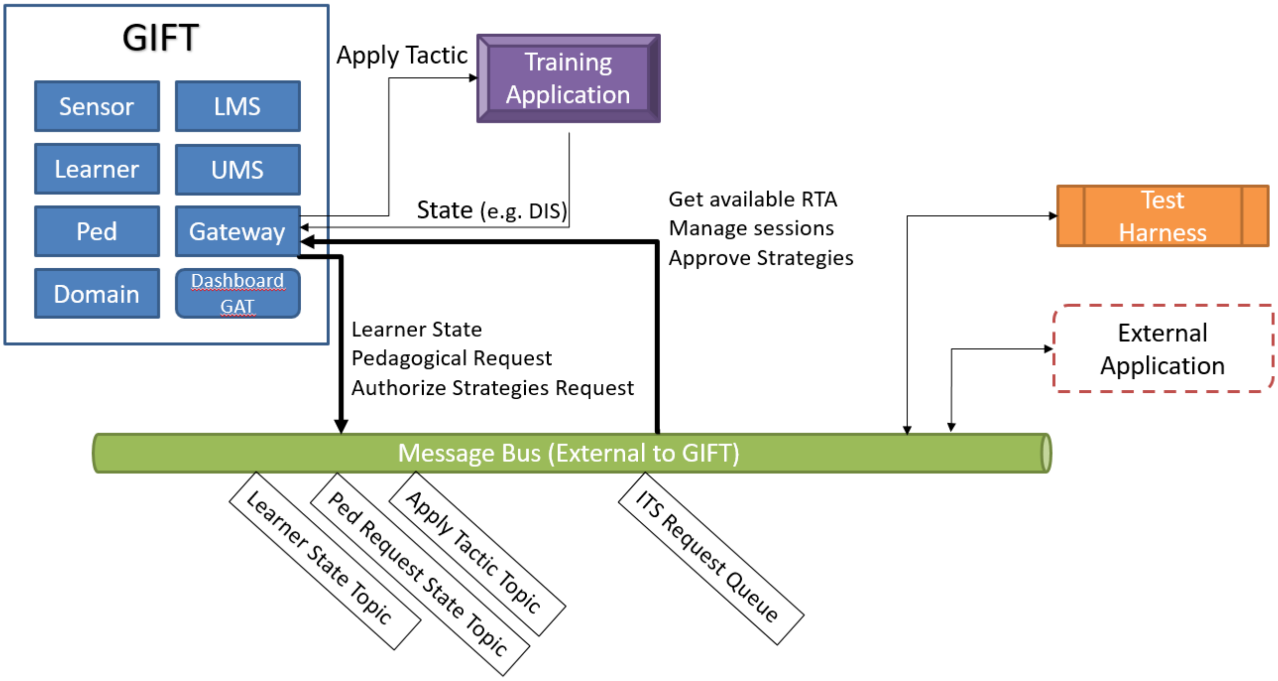


Fig. 7. Depicts the various components and messages used in the new Adaptive Learning Service configuration.

When taking a course or using GIFT's real time assessment capabilities, the learner would be required to use the Tutor User Interface (TUI) webpage to start and then follow along until completion. This requirement limited the ability for GIFT to fully expose the service oriented architecture to externals systems that want to completely manage the user experience. In this release, a new API was included called the Adaptive Learning Service API. A high-level depiction of the GIFT components being used, the subset of GIFT messages exposed and how an external message bus is utilizes is depicted in Figure 7. This new reconfiguration of a GIFT instance delivers tutoring capabilities to an external system as a service. The external system has the ability to manage sessions, receive learning effect chain events (Sottilare, Ragusa, Hoffman & Goldberg, 2013) such as learner states and approve scenario adaptation requests. The API provides the following functionality:

- query for existing real-time assessments in the GIFT instance to start (i.e. one course object courses)
- start a real-time assessment (i.e. one course object course)
- receive lesson started messages
- receive learner state messages as changes happen
- receive lesson ended messages
- receive authorize strategy request messages
- send apply strategy messages

We created a test harness Java application that can be used to test the API during a real time assessment being executed in GIFT. This open-source application can provide developers a jump start when extending their applications to be able to manage GIFT's real-time assessment services. This allows the total tutoring system to perform as a package and not only have interchanged parts, but be wholly interchanged as a service. This lends itself well to cloud deployment. For more information refer to the GIFT wiki page of https://gifttutoring.org/projects/gift/wiki/Adaptive_Learning_Service_API_2021-1.

Unity Desktop Integration

In one of the previous releases of GIFT we introduced the capability of bidirectional communication to the Unity WebGL player running in GIFT's learner browser (TUI). This included a GIFT Unity SDK that can be downloaded on gifttutoring.org and placed in your Unity project. That logic has now been extended to work with a Desktop Unity build. This includes a new Unity course object in the course creator and new Gateway module interop plugin (UnityInterface.java) used to communicate over a java.net.Socket. By using this single GIFT Unity SDK, you can choose which Unity build, WebGL or Desktop, to use and then add the corresponding Unity course object in the GIFT course creator without having to reconfigure GIFT.

New Demo Course

We strive with every GIFT release to deliver at least one new Public course. GIFT is designed to be domain-general and can be applied across all forms of training. In this version, a new course called "Cyber Security – Phishing and HTTPS" is included. This course was provided by our friends at the Institute for Creative Technologies (ICT); we thank them for their inclusion in the GIFT release. It covers two concepts through a series of lesson material, quizzes and conversation trees.

Extended Remediation Support

Following the introduction of the ICAP pedagogical model (Spain, Rowe, Goldberg, Pokorny & Lester, 2019) in the last release of GIFT, only the interactive remediation type was remaining to be implemented. GIFT now supports running 'Interactive' activities for remediation. An interactive activity is a training application (e.g. VBS, PowerPoint, Unity) that uses a real time assessment. However, unlike a traditional real time assessment, the overall/summative assessment scoring is not taken into account for updating learner state, influence future course adaptations/flow or LMS/LRS records. This new interactive remediation is available as an option in both training application and adaptive course flow course objects. It enables the configuration of custom scenarios and problem sets in an external training environment for the purpose of remediating specific concepts and skills that are being assessed at the lesson level. GIFT now has all of the ICAP activity types supported for remediation - interactive, constructive, active and passive. Another extension was made to the types of activities available for remediation when conversation trees were added. Conversation trees can be used as another form of active remediation

Furthermore, additional logic was added to GIFT's remediation service. The ICAP model was once reliant on randomized activity prioritization as the default policy. When passive was the applied type, the context selection algorithm would follow the AI rules defined at https://gifttutoring.org/projects/gift/wiki/Engine_For_Management_of_Adaptive_Pedagogy_(eMAP)_2021-1#MetadataContent-Selection-Algorithm. Now GIFT utilizes a Machine Learning policy thanks to contributions from North Carolina State University. The policy file is located at GIFT\config\ped\configurations\Default.icap.policy.xml. It uses the pretest score and remediation count as features to determine the best ICAP remediation activity to recommend next (Spain et al., 2019). In essence if a learner receives a journeyman or expert assessment on the pre-test, active remediation type is given the highest precedence. When the result of the pre-test is novice or no grade, passive remediation type is prioritized first until the learner receives two remediation attempts at which point constructive is the prescribed activity. If the ideal activity type was not authored, a random selection is applied. Normally this would result in passive content being delivered if available from the Rule/Example phases of the ICAP course object.

New Scenario Adaptations

In addition to extending automated real time assessments, we added two new scenario adaptations (seen in Figure 8). The first is the ability to add breadcrumbs in the virtual environment. Breadcrumbs can be used to guide learners to a specific destination. In VBS this appears as an X on the screen with a distance in meters to that location. The author also has the ability to remove the breadcrumbs as a strategy. Another position based indicator was also added called highlight object. An object can either be another entity or fixed location in the virtual environment. In VBS this appears a colored 3D marker. If attached to an entity, the marker will move with the entity. Highlighting an object is similar to a breadcrumb as it can be used to direct or track a specific location but the difference being whether the object is moving. As with breadcrumbs the author can also author strategies to remove the highlight.

While we have only implemented these using VBS, the underlying core architecture changes for authoring and requesting these adaptations should be applicable to other training environments. Naturally, "add this point to the game or simulation" is readily available to all simulations with points of interest. This is applicable to military training, but also items such as racing games(with simulated waypoints) or physics tutoring – such as the hints provided within the Physics Playground environment for items to click on.

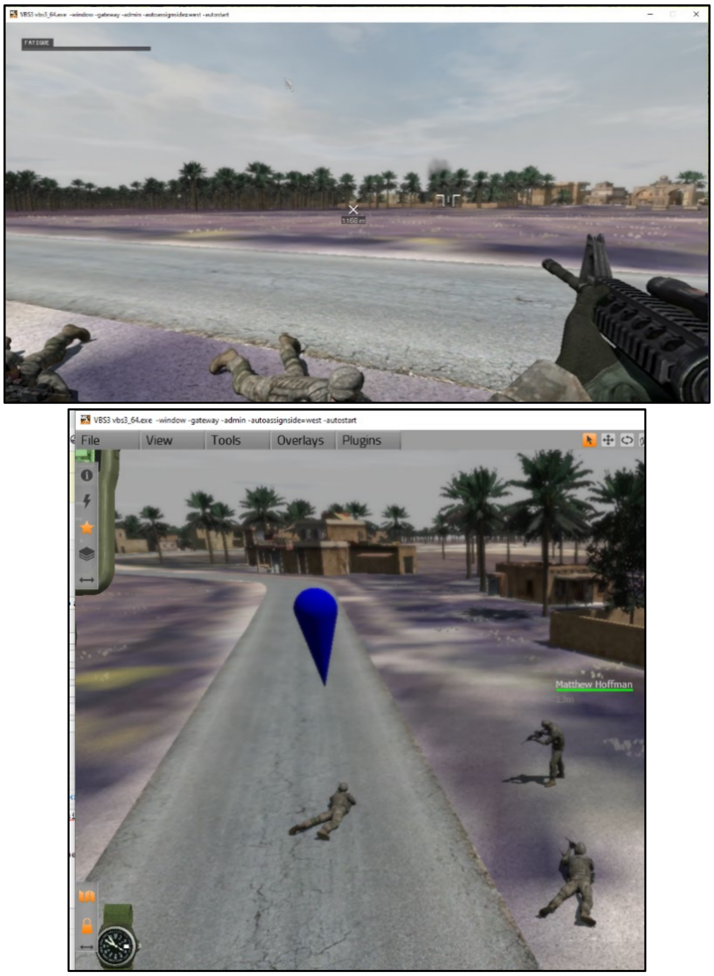

FATIGUE
VBS3 vbs3_64.exe -window -gateway -admin -autoassignside=west -autostart
File
View
Tools
Overlays
Plugins
Matthew Hoffman

Fig. 8. Screenshots of VBS showing the new breadcrumb (left) and highlight (right) scenario adaptations

Game Master UI Refactor

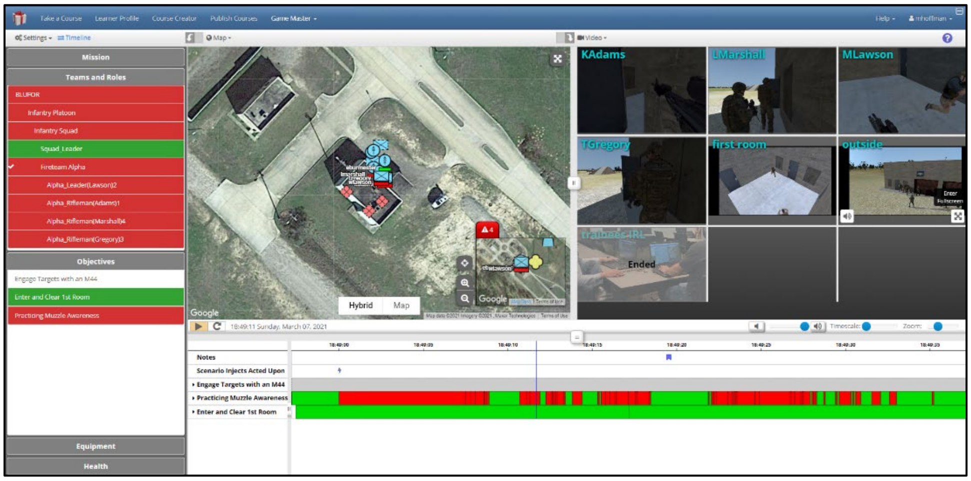


Fig. 9. Screenshot of the newly refactored Game Master user interface in Past Session playback mode

During the summer months of 2019 the GIFT Game Master was created. The first public version made its way into the GIFT 2020-1 official release in April of 2020. Since that time the Game Master has been used across several projects and use cases; more recently in the STEEL-R (STE Experiential Learning for Readiness) project. In conjunction with Applied Research Laboratories' colleagues at the University of Texas at Austin, we designed and implemented several improvements to the user interface with a focus on an improved customizable layout with mission oriented filtering (see Figure 9). The new design permits users in both active and past session experiences to choose displayed views in the two middle panels at any point in time from among map, video, scenario injects, and assessment. The display itself is customizable to the end user's preferences.

In the new game master interface the map contents is similar to previous releases, but the other panels have been improved. For example the assessment panel can now show nested concepts from the DKF. The leaf concepts under a task are shown by default with observed assessments being listed first. The scenario injects panel is no longer nested within the assessment panel for easier access. Finally a new user interface was added to upload, assign metadata, synchronize and playback video files captured during a live domain session. The video metadata is stored in a Learning Object Metadata (LOM) XML file that contains several GIFT defined extensions for title, start time, start offset time within the video, and video source (e.g. VBS, USB Camera). Generally, the user has significantly more options in regards to storage, playback, assessment, editing of assessments, and other items – the authors encourage the reader to play with it.

Another new feature is the ability to record global bookmarks/notes during an active session. In the past a bookmark had to be authored within the edit task or concept panel, meaning that the bookmark was automatically associated with that task or concept. Now there is a button on the Game Master header to create a bookmark. The user has a choice of creating a bookmark that contains text or an audio recording. For touch screen devices such as tablets we have added a gesture mode where an overlay is displayed over the entire screen. The idea behind this mode is to allow observers to maintain focus on the training happening around them rather than looking down at a screen. In gesture mode a double tap of the screen will create an empty text bookmark and a triple tap will start recording audio. These bookmarks and displayed in the Game Master session playback mode, where the user can revisit each observation and adjust the assessment model as required. This can be completed before the After Action Review is initiated. The next iteration of the Game Master will remove additional components from view during an active session in order to reduce the workload required to utilize this user interface and present what we believe are the notifications that are important to observers.

xAPI and CaSS integration

There has been a focus in recent years to establish a persistent modeling capability in GIFT to support competency-based training methods and data interoperability requirements for utilization within a learning ecosystem. We have an on-going effort to map assessments and outcomes captured in GIFT with competency frameworks that track evidence and performance over long periods of time. This involves integrating and extending core components and data specifications associated with the Advanced Distributed Learning (ADL) Initiative's Total Learning Architecture (TLA; Walcutt & Schatz, 2019) to enable tracking of experiential learning events that are delivered across simulation and synthetic resources. This extends the current utility of the TLA beyond a traditional distributed learning model with an emphasis on tracking human performance related experiences across multiple engagements. This work will inform tools and methods for tracking individual and team level competencies across a suite of training resources, with a focus on training management and accelerating unit proficiencies.

To enable this vision, we are currently integrating the TLA's Competency and Skill System (CASS) with GIFT through a custom xAPI profile. The xAPI community is embracing the technology through the implementation of xAPI Profiles (Bowe & Silvers, 2018) within the IEEE Learning Technology Standards Committee (LTSC) (Robson & Barr, 2018) . This specific profile is designed to create xAPI statements based on GIFT messages and events configured within a DKF. See Table 1 for a list of the GIFT event and associated verb used to structure an experiential statement. This work is still being completed. Stay tuned for additional updates as they become available.

Table 1. Shows the Verbs used in the xAPI statements that capture the various GIFT Events. Refer to https://gifttutoring.org/projects/gift/wiki/XAPI_Statements_2021-1 for more details.

GIFT Event	Did (Verb)
Learner State - Cognitive (Knowledge/Skill)	predicted
Learner State - Performance Assessment	demonstrated
Summative Assessment (Publish Lesson Score, Overall Assessment, DKF scoring)	assessed
Lesson/DKF start	started
Survey Response (Submitted survey results)	responded to
Finished course	completed

In brief, the xAPI from GIFT courses informs competency assessments informs readiness assessments informs course recommendation which generates xAPI data in a virtuous cycle. We welcome participation, and more information on the exact developments can be found at:

- CASS - https://www.cassproject.org/

- xAPI Profiles - http://sites.ieee.org/sagroups-9274-1-1/
- LTSC - http://sites.ieee.org/sagroups-ltsc/home/

REQUESTED FEATURES FROM GIFTSYM8

GIFT is community-driven and we take pride in our user base. Especially as it relates to functions and processes requested to support their research and content delivery needs. From last year's symposium, there were relatively few papers which actively requested or demanded features for development. This is good and shows a robust platform – the majority of papers presented describe an activity which is ongoing with GIFT, rather than addressing some weakness or shortfall. That said, a few papers requested features, with an emphasis on team tutoring assessments – to which the majority of new features presented within this paper address. Additionally, one of the papers requested the linking individual contributions to team contributions – to which we agree and is forthcoming, but requires a metric of both individual and team assessment first; we are getting there. Another paper requested a robust NLP pipeline for use in team assessments; this is currently under development and will be presented within this symposium.

GIFT AND IEEE STANDARDS ON ADAPTIVE INSTRUCTIONAL SYSTEMS

The discussion continues on adaptive instructional systems through the IEEE Learning Technologies StandardsCommittee (LTSC). LTSC coordinates with other organizations that produce specifications and standards for learning technologies. The GIFT community invites the reader to join the conversation on what data exchange standards for learning technologies might look like in the future. GIFT is scheduled to be included in the Adaptive Instructional Systems Consortium Resource Repository later this year as a tutoring architecture. Interested readers are encouraged to go to the IEEE LTSC meetings to become involved.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

The GIFT program has seen significant advancement since its conception in 2011. Each year, the community continues to build out new features and use cases that extend the boundaries of adaptive instructional systems. With a near-term focus on utilizing GIFT to address team tutoring challenges, we are excited to continue evolving the tools and methods to address critical capability gaps to drive future training require-ments and system development. While the focus is on teams, it is well understood that the individual cannot be ignored. Stay tuned for continued improvements that address all facets of intelligent tutoring in today's education and training climate. Check back next year to see what kind of progress we're able to make!

REFERENCES

Brawner, K., Heylmun, Z., & Hoffman, M. (2017). *The GIFT 2017 Architecture Report.* Paper presented at the Proceedings of the 5th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym5).

Brawner, K., & Hoffman, M. (2018). *Architecture and Ontology in the Generalized Intelligent Framework for Tutoring: 2018 Update.* Paper presented at the Proceedings of the 6th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym6).

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), Orlando, FL.

Goldberg, B., Brawner, K., & Hoffman, M. (2020). *The GIFT Architecture and Features Update: 2020 Edition.* Paper presented at the Proceedings of the 8th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym8).

Ososky, S., & Brawner, K. (2016). *The GIFT 2016 community report.* Paper presented at the Proceedings of the 4th Annual GIFT Users Symposium.

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 2013 Interservice/Industry Training Simulation and Education Conference (I/ITSEC)*. Orlando, FL.

Sottilare, R., Brawner, K. W., Goldberg, B. S., & Holden, H. A. (2012). The Generalized Intelligent Framework for Tutoring (GIFT).

Spain, R., Rowe, J., Goldberg, B., Pokorny, B. & Lester, J. (2019). Enhancing Learning Outcomes Through Adaptive Remediation with GIFT. *In Proceedings of the 2019 Interservice/Industry Training Simulation and Education Conference (I/ITSEC)*. Orlando, FL.

Walcutt, J.J., Schatz, S. (2019). Modernizing Learning: Building the Future Learning Ecosystem. Washington, DC: Government Publishing Office.

ABOUT THE AUTHORS

Michael Hoffman is a senior software engineer at Dignitas Technologies and the technical lead for the GIFT project. He has been responsible for ensuring that the development of GIFT, meeting community requirements, and supporting production ITS systems, ITS research, and 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 and their contractors. In addition he utilizes hisexpertise in integrating third party capabilities such as software and hardware systems to enable other organizations to integrate GIFT into their training solutions.

Benjamin Goldberg, PhD is a senior research scientist at the U.S. Army Combat Capability Development Command – Soldier Center, and is co-creator of the Generalized Intelligent Framework for Tutoring (GIFT). Dr. Goldberg is the team lead for a research program focused on the development and evaluation of Training Management Tools for future Army training systems. His research is focused on the application of intelligent tutoring and artificial intelligence techniques to build adaptive training programs that improve performance and accelerate mastery and readiness. Dr. Goldberg has researched adaptive instructional systems for the last 12 years and has been published across several high-impact proceedings. He holds a Ph.D. in Modeling & Simulation from the University of Central Florida.

Keith Brawner, PhD is a senior researcher for the U. S. Army Combat Capability Development Command Soldier Center at the Simulation and Training Technology Center (CCDC-SC-STTC), and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT). He has 13 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 efforts are on artificial intelligence for the Synthetic Training Environment Simulation and Network Compression. He manages research in adaptive training, semi/fully automated user tools for adaptive training content, and architectural programs towards next-generation training.

The 2021 Authoring Guide for GIFT

Anne M. Sinatra
U.S. Army Combat Capabilities Development Command (DEVCOM) – Soldier Center –
Simulation and Training Technology Center (STTC)

Introduction

Welcome to the 2021 Authoring Guide for creating courses in the Generalized Intelligent Framework for Tutoring (GIFT). I have written a series of two guides through the years that have had two different goals: the Research Psychologist's Guide to GIFT (Sinatra, 2014; Sinatra, 2016; Sinatra, 2018; Sinatra, 2020), and the Instructor's Guide to GIFT (Sinatra, 2015; Sinatra, 2019). The Research Psychologist's Guides focus on the current version of GIFT at the time of writing, and lessons learned when utilizing GIFT for research. The Instructor's Guide to GIFT focuses on how to create instruction using GIFT, but also what features would be most important from the perspective of an instructor. Recently, there has been interest in an Authoring Guide for GIFT which provides an example of how to create an actual course in GIFT including an adaptive courseflow object, and has step by step instructions, as well as suggestions/lessons learned. While there is an overlap between this area and the previous guides, the current one is unique in that this can be used along with the 2021 version of GIFT and GIFT Cloud to create courses.

WHAT IS GIFT AND HOW DO I USE THIS GUIDE?

GIFT is a domain-independent intelligent tutoring system framework that is continually being improved (Sottilare, Brawner, Sinatra, & Johnston, 2017). As a result of continuing improvements, the general terminology, concepts and approaches used to create courses in GIFT stay relatively stable, however, there may be some updates that occur that may change how things work. The current guide aligns most closely with the 2020 and 2021 versions of desktop GIFT, and GIFT Cloud in 2021. If you are reading this paper a few years after it was written, it still is likely to have great relevance, there may just be some minor changes that have occurred to GIFT in the meantime. GIFT includes a suite of authoring tools that allow an author to create GIFT courses on any topic that they choose. The tools in GIFT remain the same, but the material that is being tutored is up to the individual author of a GIFT course.

The current guide can be read in order, or you can jump to the portions of it that seem like they will be most helpful to you. The guide starts with GIFT Terminology, then provides a Step by Step worked example of a course that you can create in GIFT using an adaptive courseflow object, and finally ends with examples of important decisions to make when constructing your GIFT course.

Terminology

The following are some helpful terms that are frequently used in GIFT:

GIFT Cloud and GIFT Desktop: There are two versions of GIFT. The Cloud version of GIFT is accessed online and allows the user to interact with the course without needing to download any software. GIFT Desktop is the downloadable version of GIFT which runs a separate instance of GIFT on your computer. This version is often used by developers. Most general users will use the Cloud version of GIFT.

GIFT Course: In GIFT, the material that a learner is expected to interact with in one session is defined as a GIFT Course. The author has the choice of how much material is going to be covered in each generated course, however, something like a traditional online course would be made up of many GIFT courses if it was implemented in GIFT. You can think of a GIFT course as similar to a module in a more traditional online class.

GIFT Concepts: After you create a course in GIFT, you will need to identify the concepts that you are covering in the course. This is done by using the menu on the left side of the screen in the GIFT Authoring tool. This guide will include details about how to do this.

GIFT Tile Screen and Course Tiles: When you first login to GIFT you will see a screen with a number of different Course Tiles. Each of these course tiles represent a GIFT course that you are able to access. If you hover your mouse over the tile a "Start" button will appear. If you want to take the course as a learner, or see the learner view this is the approach you will use. In the bottom right hand of each tile there are a number of symbols. Among them are a pencil. If you click on the pencil it will open that specific GIFT course in the GIFT Authoring Tool.

GIFT Authoring Tool: The GIFT Authoring Tool is where you can create your GIFT course. It can be accessed by clicking "Course Creator" from the top of the GIFT menu, or by clicking the pencil icon on a specific course to open that course.

*Sharing a Course:*You can directly share a course with another GIFT user by clicking the share symbol on the tile for the course on the GIFT Tile Screen. You will need to type in the user's GIFT ID, and you will need to select what type of privileges they should have for the course (e.g., 1. Can only take the course, 2. can take the course or view it in the authoring tool, 3. have full edit and functionality privileges for the course).

Published Course: If you want to quickly share a course with a learner who does not have or does not need a GIFT account you can use the publish course feature. This feature is ideal for an experiment or a lesson if you do not intend to use GIFT consistently in your class. If you would like to link the data back to your course make sure that you include in a demographics question that asks for a student number of student name. When publishing a course it will take the version of the course that currently exists and provide a link to it that can be shared with learners. When the learner clicks the link they will be brought to a page with a "Start" button. By clicking the start button they will engage with the GIFT course, and then the data will be saved and you can later extract it. However, they will not login to GIFT.

Survey Authoring System: The Survey Authoring System is a component of the GIFT Authoring Tool that allows you to create questions, questions banks, and full surveys in your GIFT course. Question Banks are tied to the course itself, whereas surveys can be imported from one course to another.

Adaptive Courseflow: The Adaptive Courseflow object in GIFT is where adaptation occurs. The adaptive courseflow is broken down into four different categories: Rules, Examples, Recall, and Practice. Practice is less frequently used thank the other groups. You will need to associate course concepts with each adaptive courseflow object.

Question Bank: A question bank is associated with each GIFT course that you create. When using the question bank object, or the question bank as part of an adaptive courseflow object it allows you to enter in new questions that can be used in your course. The question bank function displays a random question that is associated with parameters that the author sets. In order for it to function properly the author will need to create concepts for the course, and then specifically add those concepts and a difficulty level (easy, medium, or hard) to the properties of each question. The author then configures the question bank to randomly select a certain number of questions from each desired concept with an associated difficulty level (e.g., 3 easy Short Term Memory questions, and 2 hard Short Term Memory Questions).

*Survey:*A survey is an ordered set of questions that are authored and displayed to the learner. Unlike the question bank, these are not associated with concepts, and are always presented in the same order. They can either be demographics based, or authored in such a way that they are graded and performance can influence the GIFT course. A survey can be imported into a question bank, but a question bank cannot be imported into a survey. You would use a survey if you were creating an experiment, or if you were doing an assessment like a pre or post test where you want to ensure that the same questions are always received in the same order. If you want the ability to import existing written questions into a question bank you can create them in a linear survey and then later import that survey and items into a different course's question bank.

*PowerPoint:*A PowerPoint course object can be selected when you are authoring a course, and it displays a PowerPoint show file (.pps). If you use the PowerPoint course object it directly connects with the learner's instance of PowerPoint that is on their computer; therefore a gateway module needs to be temporarily downloaded to their computer. An author would use this object if they wanted to monitor the amount of time the learner spends in the PowerPoint, or if there are interactive elements to the PowerPoint. Additionally, if you have videos or animations in your presentation you would need to use a PowerPoint. If your PowerPoint is only image and text based, you can use a SlideShow course object instead so that it does not require a download.

SlideShow: The SlideShow course object can be selected when you are authoring a course, and it converts an existing PowerPoint show file (.pps) to images that are presented in order. If you are using only text and images in a PowerPoint this is the ideal course object to use, as it does not require any downloads to run.

GIFT Gateway Module: When using GIFT Cloud if as part of the course you are utilizing to a desktop application such as PowerPoint or Virtual Battlespace 3 it requires GIFT to make a connection with the learner's computer. This is done by temporarily downloading a gateway module that needs to be run on the learner's computer. In some cases this is absolutely necessary. However, in others cases, there are ways to remove the need for a download (i.e. using the SlideShow object instead of the PowerPoint object in your course).

GIFT log files: GIFT stores the actions that a learner takes when they interact with the system in a log file. In current versions of GIFT in order to extract the data and assessments that occurred when a learner was in the system you will need to download the log file and use the Event Report Tool to extract it so that you can view it in a program such as Excel.

Event Report Tool: The event report tool is available on the desktop version of GIFT, or a limited version of it is associated with Published Courses in GIFT. In the near-term there is expected to be a solution that allows data from traditional GIFT courses to be extracted using GIFT Cloud.

How to Use This Paper

Now that you are familiar with the terminology associated with GIFT, the next step is to jump into the worked example. The goal of this paper is to provide background on GIFT, as well as a worked example of creating a course that includes an adaptive courseflow object, which is an adaptive component of a GIFT course. The worked example covered in this guide will be relatively straightforward, and it will not discuss advanced items such as the Domain Knowledge File. This guide should provide an opportunity for new course authors, or those refamiliarizing them with the system to successfully build a course.

WORKED EXAMPLE

When you first login to GIFT you will see a screen with tiles on it. Each of these tiles represents a different existing GIFT course. Some of the courses are Public (available to everyone who is a user of GIFT), whereas, others are ones that you have created and are unique to your GIFT login screen, and there can also be courses that have been shared with you by other GIFT users (though your editing permissions for these may be limited). You can edit one of these courses by hovering your cursor over the tile and clicking the pencil icon. Doing this will open that course in the GIFT authoring tools. In the case of the public courses you have read access, but cannot actually edit them. You will need to copy the course first before you can edit it. An example of the course tiles page can be seen in Figure 1.

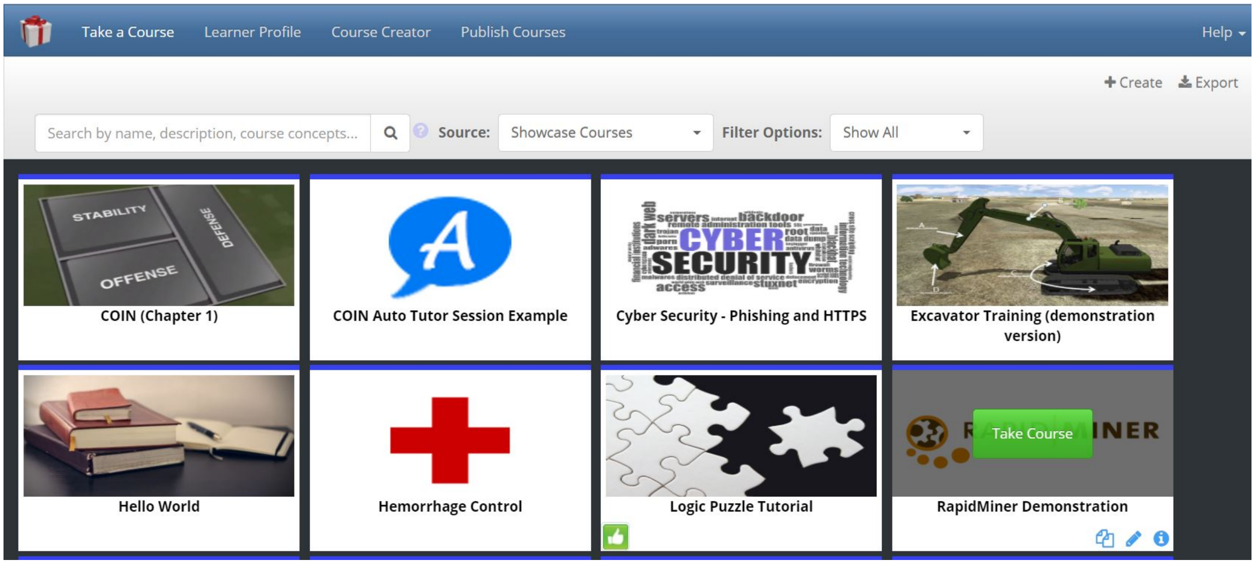
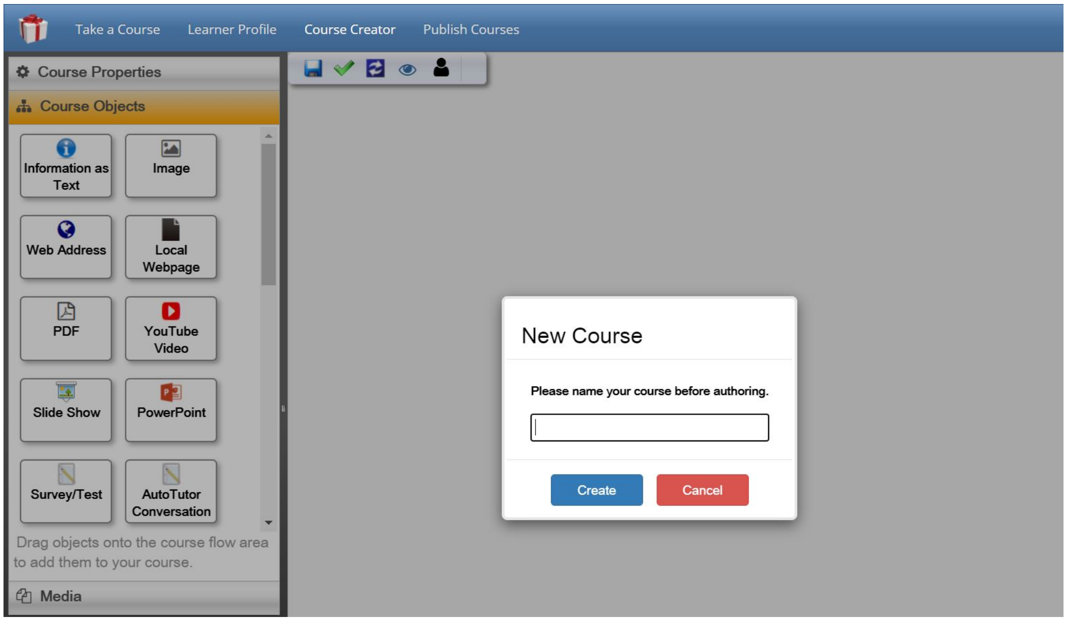


Figure 1. GIFT Course Tiles login interface.

For the purposes of this paper we will be creating an entirely new course.

1) On the menu on the top of the page and click "Course Creator". This will create a new course.

2) Next, you will see a pop up that asks you to name your course. Type in "TestCourse1" or your chosen course name as the name. (See Figure 2 for an image).



Drag objects onto the course flow area to add them to your course.

Please name your course before authoring.

Figure 2. Interface for adding the name of your course.

3) After you have entered the name of your course, hit the floppy disk icon (the icon to the left of the green check) within the interface. This will SAVE your course. You should do this frequently after taking actions. NOTE: There is no Undo button in GIFT, but there is a "REVERT TO LAST SAVE"

button, which will revert your course to how it was before you last pushed the save button (it is the blue button between the checkmark and eye).

If you want to continue working on your course later, make sure that you hit the save button before exiting. When you log back in find the course tile with the name of your course, hover your mouse over it and hit the pencil icon on the bottom left of it (see Figure 1).

4) Now click on "Course Properties" in the top of the left hand menu. When you click on it the full list of course properties will become available (the color of the words will also become orange to indicate that it is highlighted), and the course objects menu will collapse (See Figure 3).

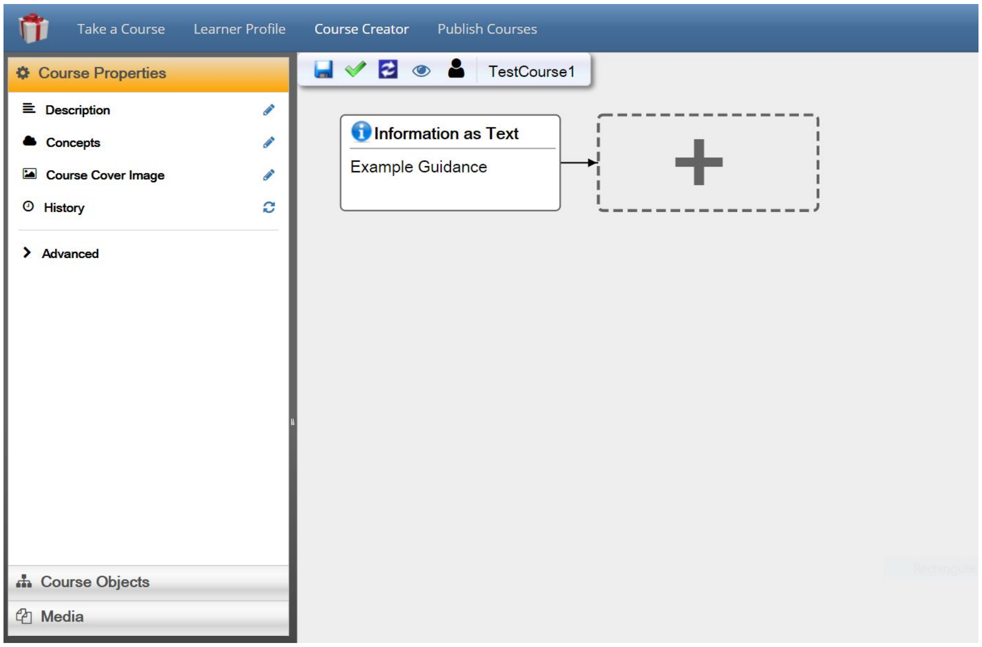
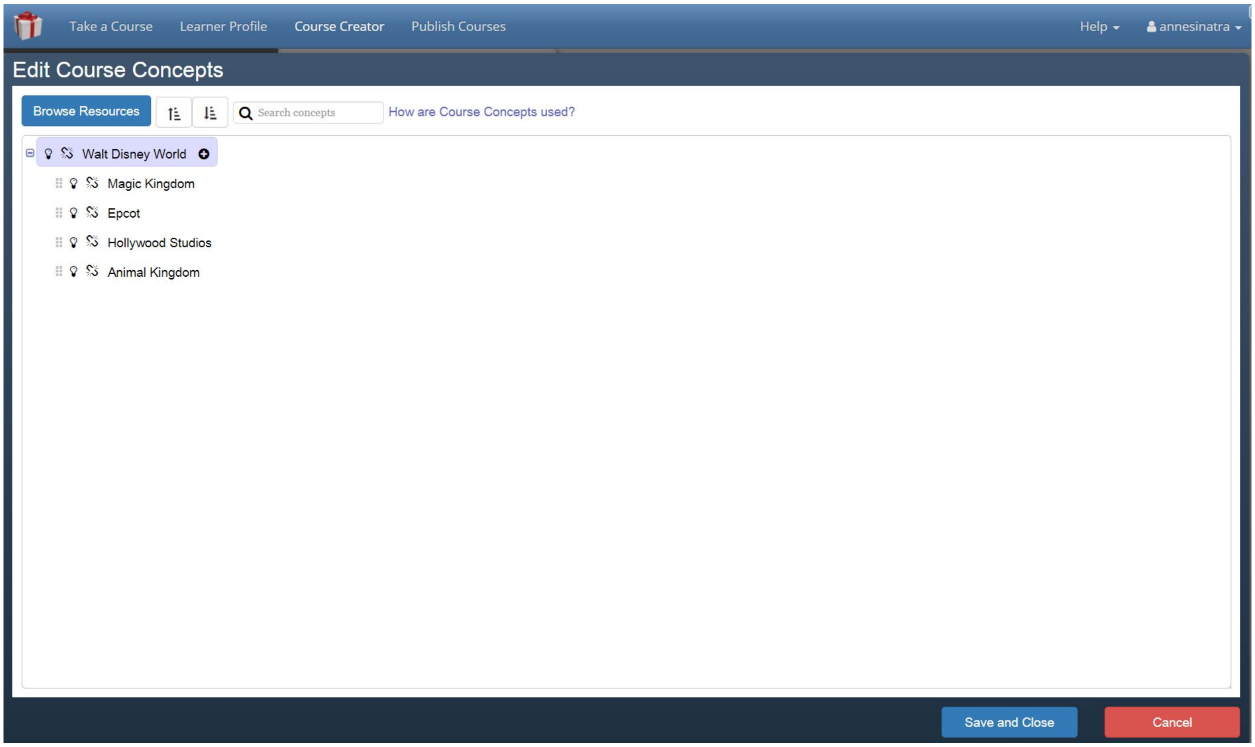


Figure 3. GIFT Course Authoring interface with Course Properties expanded.

5) Now click on the pencil icon to the right of "Concepts". See Figure 4 for the Edit Course Concepts interface.



Edit Course Concepts

- Walt Disney World
  - Magic Kingdom
  - Epcot
  - Hollywood Studios
  - Animal Kingdom

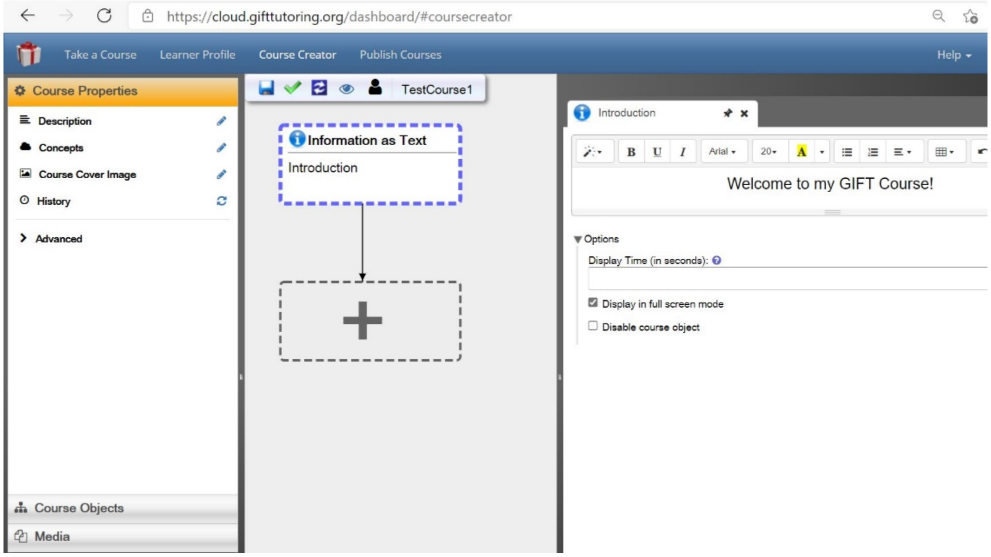
Figure 4. Course Concepts authoring interface.

6) Create a series of course concepts. These are the concepts that your GIFT course is going to cover, and that you will assign to questions. For my example I am creating a course about Walt Disney World, so I have made this my main concept. Click on the words to edit what it says. To add new sub-concepts click on the plus button. For instance, I entered "Walt Disney Word" then clicked the plus button to the right of it multiple times to add the four different Walt Disney World parks below it. If I wanted to add the rides to each park I would hover over the sub-concept (e.g., Magic Kingdom) and click the plus button next to that. Then each time I did that I could rides such as "Pirates of the Caribbean", "Big Thunder Mountain Railroad", etc. After you are done adding your concepts click "Save and Close". Now hit the Save symbol on your entire course.

It is important to define early on what your course concepts will be. You are going to associate these concepts with questions in the question bank, materials that you provide, remediation, and adaptive courseflow objects. As the course author it is up to you how detailed you are when defining your concepts, and it often requires a lot of thought. The more you break down the course items the more items that will require to be authored for you course. Also, just because you create a concept it does not mean that you have to use it or build it out. For instance, I have added Walt Disney World as my main concept, and put my sub-concepts as the 4 Disney parks. I can move forward with this for my course, or I can go down to another level of sub-concepts. Now it is up to me if I want to define this next level as lands in the parks (e.g., Fantasyland, Adventureland, Frontierland, Tomorrowland) or specific rides in the park (e.g., Space Mountain, Dumbo). If I wanted to I could even break it down by the categories of ride type (e.g., rollercoaster). It is up to the author on how they want to define this concept structure. Additionally, even though it is represented as a hierarchy that is enabled conceptually to assist the author, as hierarchical scoring is not currently enabled in GIFT. Each of the hierarchical concepts/sub-concepts will be scored similarly and are linked to how the author puts them together as opposed to a calculation from the GIFT system.

7) Now that you are back on the main authoring screen click on the "Example Guidance" that was automatically added to your course (See Figure 5). Modify the text in the box to provide your introduction message to the learners. Then click next to the "i" to modify what the title of the course

object is so that you will be able to identify it from the main authoring screen. I have changed the text in the object that was there, and updated the title of the object to "Introduction".


https://cloud.gifttutoring.org/dashboard/#coursecreator

Welcome to my GIFT Course!

Display Time (in seconds):

☑ Display in full screen mode

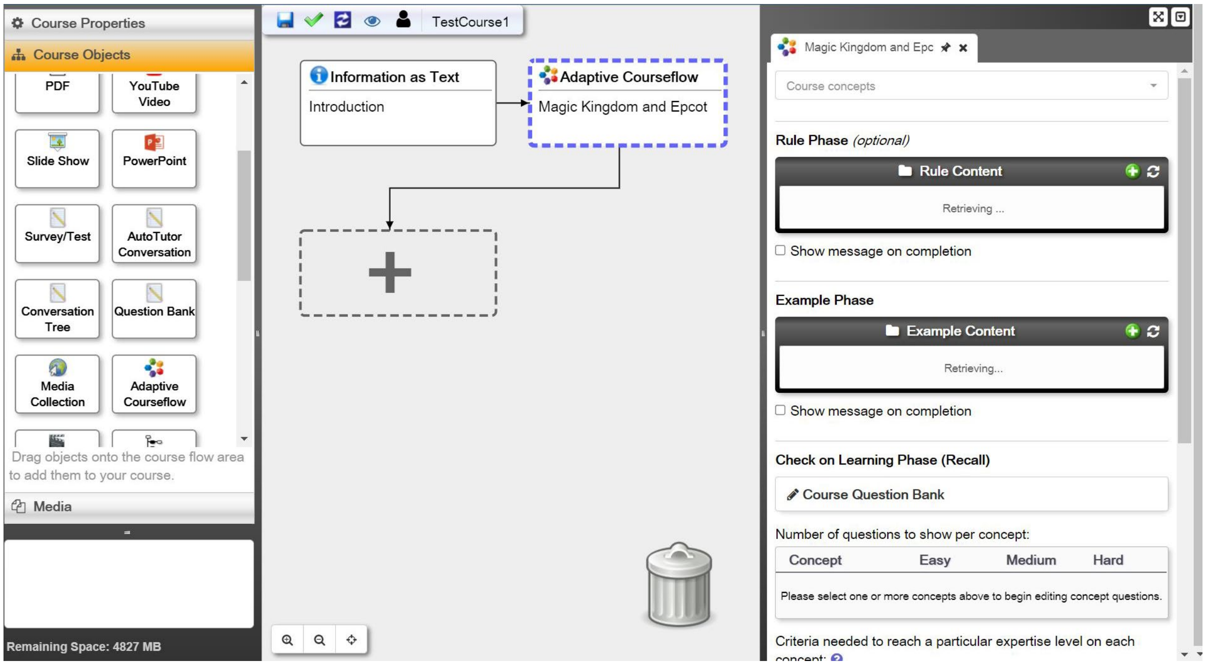
☐ Disable course object

Figure 5. Click on the "Information as Text" in order to modify the content of the message and the title of the course object.

The information as text course object can be used to provide information/messages to the learners without them needing to take any action. Now we are going to add some more items to our course.

8) Click on "Course Objects" on the left side of the screen. That menu will now open again and you will see all the objects that you can use in your course.

9) Pull the Adaptive Courseflow object over and place it on the plus symbol in the courseflow in the center of your screen. It will ask you to name the object. In my case I named it "Magic Kingdom and Epcot", because this is what I plan to cover in this adaptive courseflow. See Figure 6 for a screenshot of the Adaptive Courseflow interface.



Drag objects onto the course flow area to add them to your course.

Remaining Space: 4827 MB

Rule Phase *(optional)*

Retrieving ...

☐ Show message on completion

Example Phase

Retrieving...

☐ Show message on completion

Check on Learning Phase (Recall)

Number of questions to show per concept:

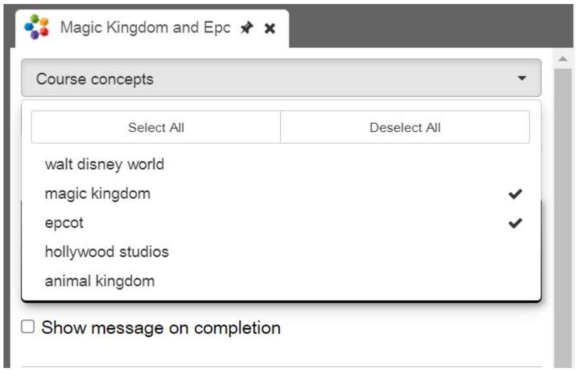
Please select one or more concepts above to begin editing concept questions.

Criteria needed to reach a particular expertise level on each concept:

Figure 6. Adaptive Courseflow interface.

Now that I have added an adaptive courseflow object, the left side of my screen still has the other available course objects, the center of my screen has my overall GIFT courseflow, and the right side of the screen has the properties that I author for the selected course object. If I want to make it larger I can grab the line to the left of the configuration screen and pull it to the left.

10) Select the concepts that you plan to cover in this specific adaptive courseflow using the “Course concepts” pull down menu at the top of the adaptive courseflow configuration panel. For my example I am going to select “Magic Kingdom” and “Epcot”. I can create a second adaptive courseflow to cover “Hollywood Studios” and “Animal Kingdom”. If I wanted to I could cover all 4 concepts in this 1 adaptive courseflow, or I could cover 1 each. It is up to the author. See Figure 7 to see what the course concept selection interface looks like in the adaptive courseflow.



walt disney world
magic kingdom ✓
epcot ✓
hollywood studios
animal kingdom

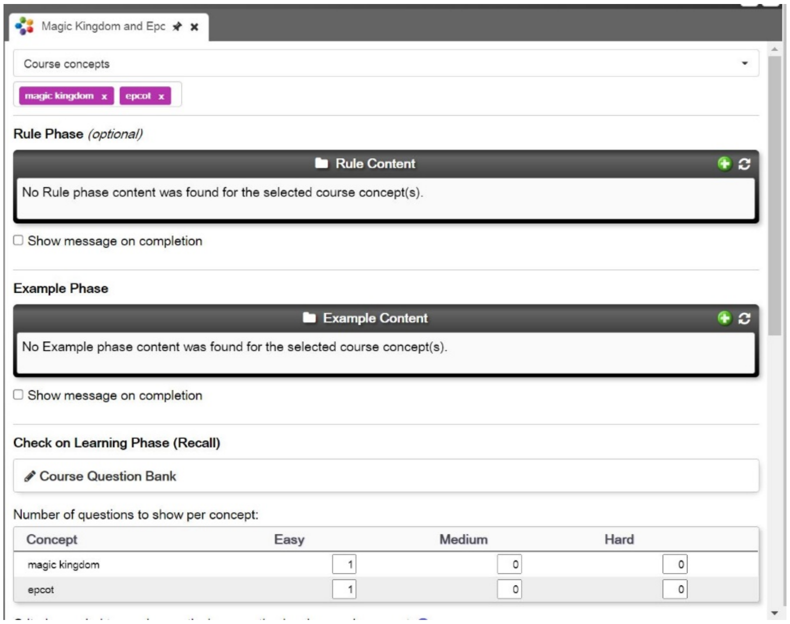
□ Show message on completion

Figure 7. Course Concept selection in the Adaptive Course flow course object.

The adaptive courseflow object in the main GIFT course indicates that material on the specific concepts will be presented, the performance of the learner will be evaluated, and they will receive remediation on the concept(s) that they missed until they pass a certain author set threshold. After the adaptive courseflow object is completed the learner will then move onto the next overall GIFT course object.

The adaptive courseflow object is modeled after Merrill’s Component Display Theory, and is broken down into 4 phases (Rules, Examples, Recall, and Practice). The author will add the materials that they would like to use for each of these phases into the interface. The Examples, Recall, and Remediation entries are required. The Rules and Practice phases are optional.

See Figure 8 for the adaptive courseflow configuration interface. Notice that each of the concepts I added are now listed on the top of the page. Additionally, I can now add items to the Rules or Examples phases by clicking on the plus symbol next to them. You can also see the Check on Learning (Recall) phase, and the question bank button. In order to add items to your question bank you will click that button. You will then select the appropriate numbers of items for each concept that you would like shown in the interface that was below it.



Rule Phase *(optional)*

No Rule phase content was found for the selected course concept(s).

☐ Show message on completion

Example Phase

No Example phase content was found for the selected course concept(s).

☐ Show message on completion

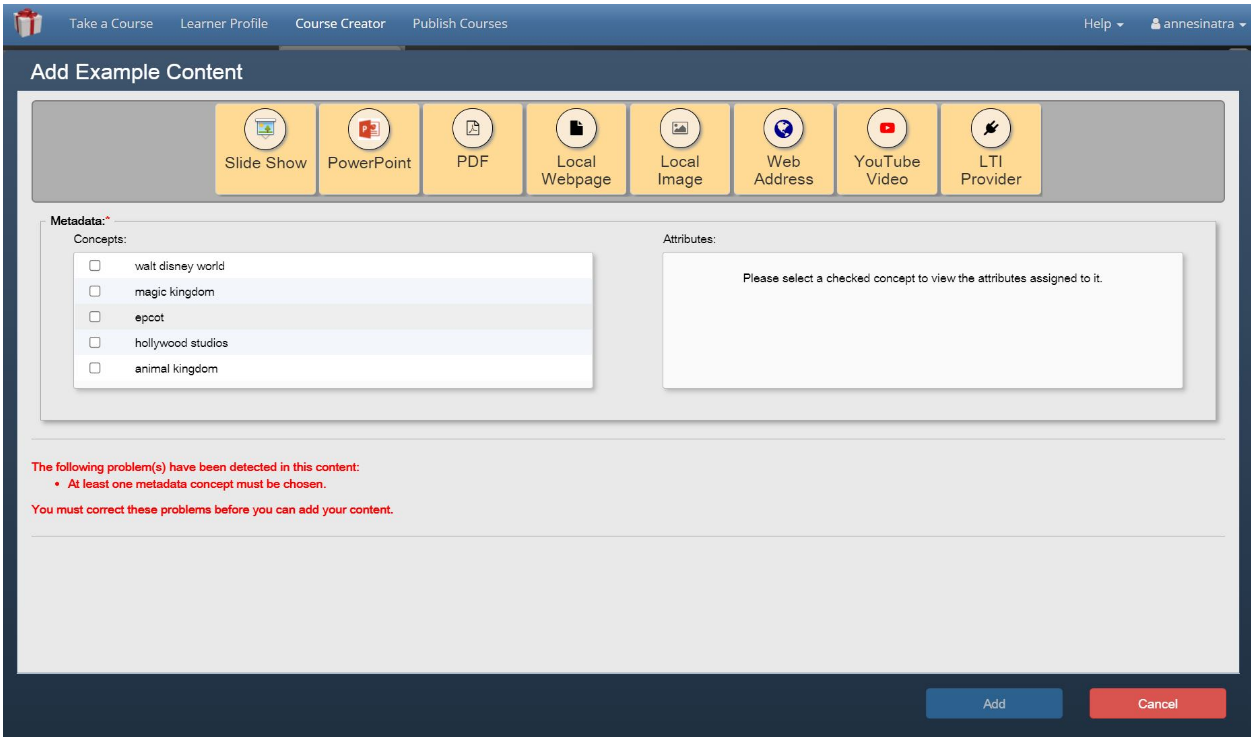
Check on Learning Phase (Recall)

Number of questions to show per concept:

Concept	Easy	Medium	Hard
magic kingdom	1	0	0
epcot	1	0	0

Figure 8. Adaptive Courseflow interface showing the Rules, Example, Recall and Practice phases.

11) For the purposes of this example I am skipping the optional "Rules" phase, and will be entering the required materials in the "Examples" section. This is the primary content that the learners will receive about your course concepts. The content can be a Slide Show, PowerPoint, PDF, Local Webpage, Local Image, Web Address or YouTube video. The same content types are possible if you wish to populate the Rules phase. See Figure 9 to see the interface for uploading your Example Content. Additionally, you will need to make sure to fill in the meta-data below it and associate it with the correct concepts. If you do not do this then the item will not appear as an option on the main screen. It is recommended that you click the Example Content type first, upload it, and then use that interface to select the meta-data (as opposed to entering it on the main page). For this example I am going to click "Slide Show".



Add Example Content

Metadata:*

Concepts:

☐ walt disney world
☐ magic kingdom
☐ epcot
☐ hollywood studios
☐ animal kingdom

Attributes:

Please select a checked concept to view the attributes assigned to it.

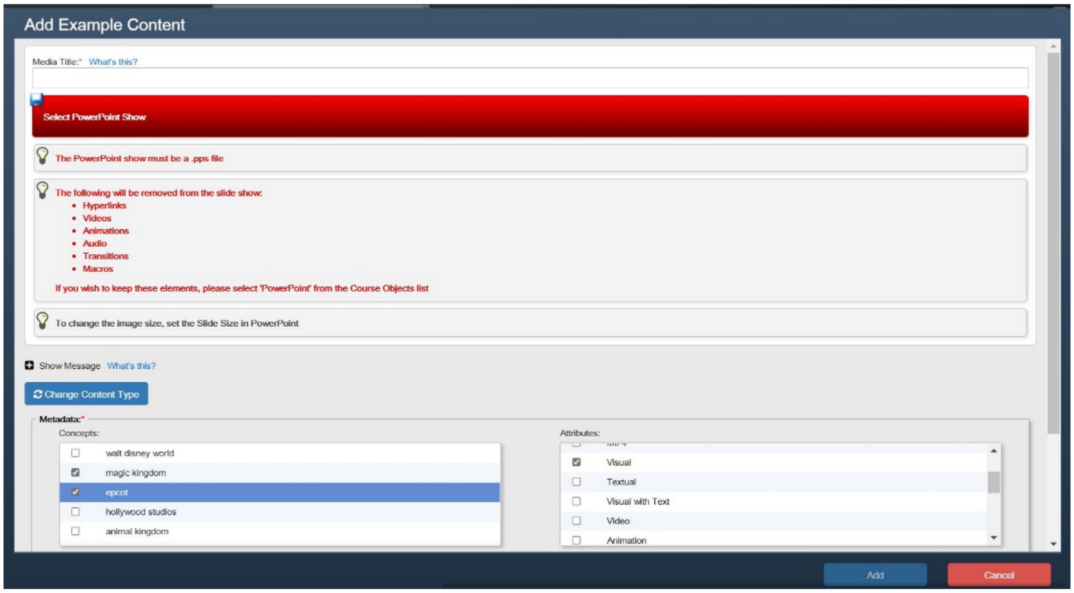
The following problem(s) have been detected in this content:

- At least one metadata concept must be chosen.

You must correct these problems before you can add your content.

Figure 9. Example Content configuration screen.

12) After clicking on "Slide Show" I need to select the file that I want to upload, and then associate it with the proper concepts. Additionally, I will choose a meta-data type which represents the type of content (e.g., Visual, Textual). See Figure 10 for the Example content upload screen. Click on the red "Upload PowerPoint Show" button to choose the item. Then add the concepts and attributes in the meta-data; You will need to add separate attributes for each concept before it will let you click "Add".



Add Example Content

Media Title:* What's this?

The PowerPoint show must be a .pps file

The following will be removed from the slide show:
- Hyperlinks
- Videos
- Animations
- Audio
- Transitions
- Macros

If you wish to keep these elements, please select 'PowerPoint' from the Course Objects list

To change the image size, set the Slide Size in PowerPoint

Metadata:*

Concepts:

☐ walt disney world
☑ magic kingdom
☑ epcot
☐ hollywood studios
☐ animal kingdom

Attributes:

☑ Visual
☐ Textual
☐ Visual with Text
☐ Video
☐ Animation

Figure 10. PowerPoint Example Content Upload screen.

Only one item will be shown to the learner in the Examples phase so make sure that the content you put in this phase represents all of the concepts that you want to teach. You can also include multiple different types of Example content, and the system will choose one to present to the learner. The only way to ensure that the learner will always receive the same content every single time is to only put one item in the Example content. If you are uploading a PowerPoint to be converted into a Slide Show it will need to be in .pps format. This is a "PowerPoint 97 – 2003 Show" file. Make sure to save your PowerPoint as this file type in the PowerPoint program, otherwise it cannot be converted to a Slide Show. If you want to add additional possibilities in the Examples phase repeat the process described above. Similarly, if you do want to add content to the Rules phase follow the same process. The Rules content will be presented prior to the Examples content. You would not want to add duplicate content to both phases because content will be presented from both. The Rules phase is intended to be used to describe the material you want to use, and the Examples phase is intended to provide more in-depth examples of the content.

The next step is to populate the Recall phase, and your Course Question Bank.

13) See Figure 6 for the image of the adaptive courseflow. You will click on "Course Question Bank" in the Check on Learning (Recall) section.

The Course Question Bank is where you enter the questions that will be used in your GIFT course. There are two main interfaces, the Writing Mode, and the Scoring Mode. You will write your questions in the writing mode (the background is GREEN), and you will set the scoring and associating concepts/difficulty levels for your questions in Scoring Mode (the background is BLUE). You toggle between the modes by clicking the name of the modes on the top of the screen.

14) When you enter the Course Question Bank interface it will be a blank green screen. Click on "Add Survey Item" on the bottom of the screen and the Select a Survey Item interface will appear. See

Figure 11 to see the interface. You can choose the type of question that you want to add, or you can import questions from an existing survey that has been created in your course or a different course. As of writing time you cannot import existing question banks from other courses, only existing linear surveys.

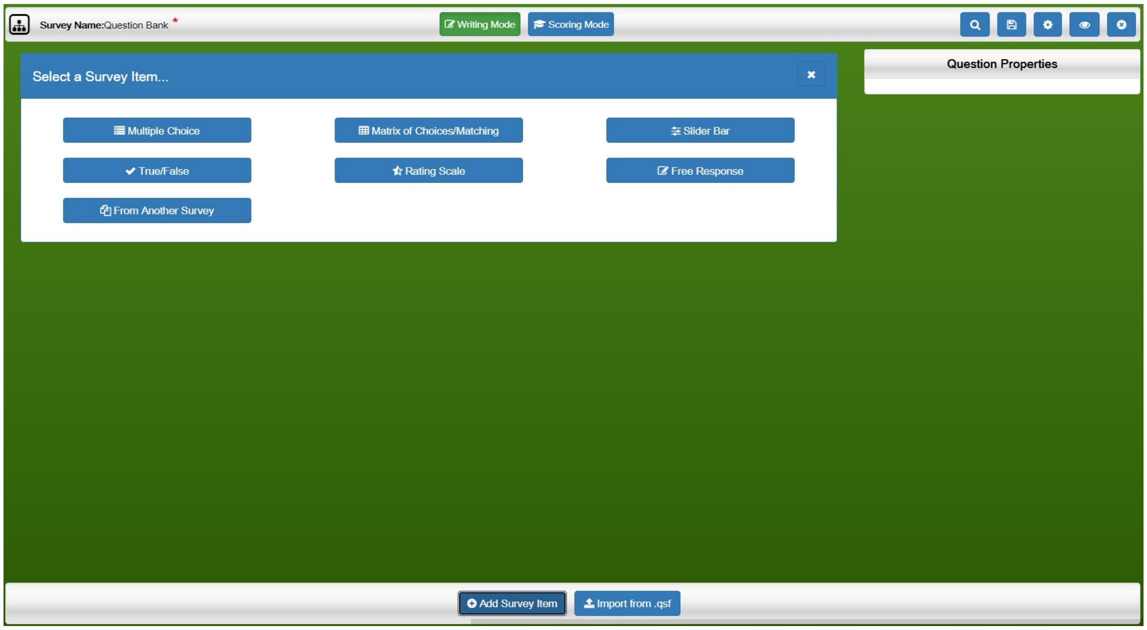
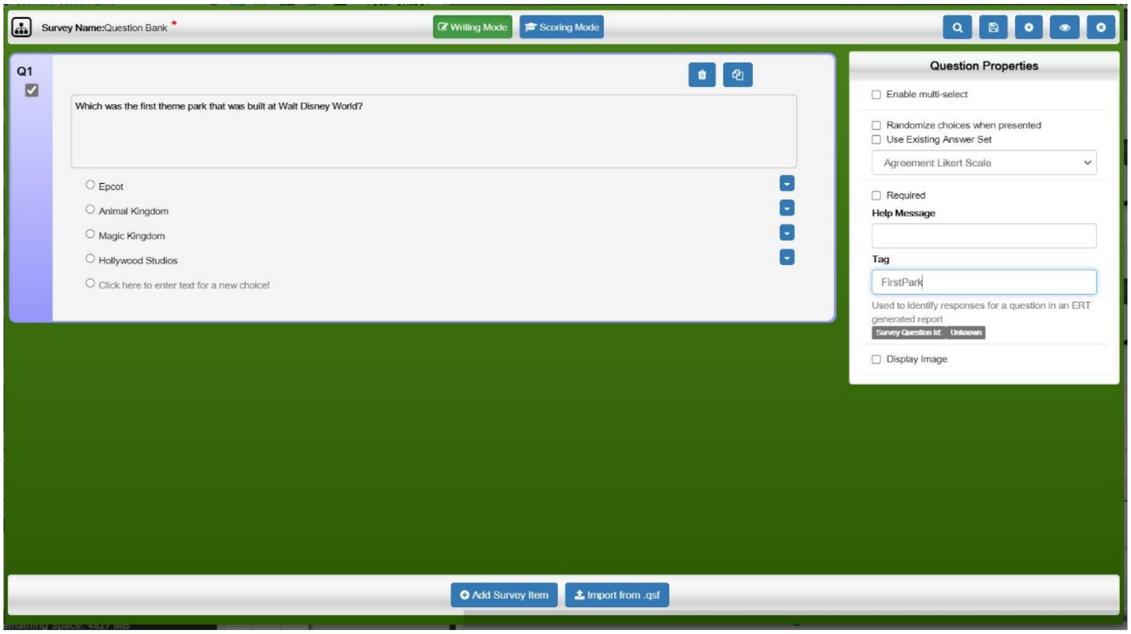


Figure 11. Add survey item interface for the Course Question Bank

15) Click on "Multiple Choice" in the Select a Survey Item interface. A question editor will appear with the first question. Enter you question into the textbox, and add as many multiple choice items as you would like below it. On the right side of the screen you will see additional options of question properties. If you want to be able to easily identify a question when you look at the responses you may want to add a Tag which will be displayed in association to the question. Keep clicking on "Add Survey Item" to include additional items in your Question Bank. See Figure 12 for an example question.



Q1

Which was the first theme park that was built at Walt Disney World?

○ Epcot
○ Animal Kingdom
○ Magic Kingdom
○ Hollywood Studios
○ Click here to enter text for a new choice!

□ Enable multi-select

□ Randomize choices when presented
□ Use Existing Answer Set

Agreement Likert Scale

□ Required

Help Message

Tag

FirstPark

Used to identify responses for a question in an ERT generated report

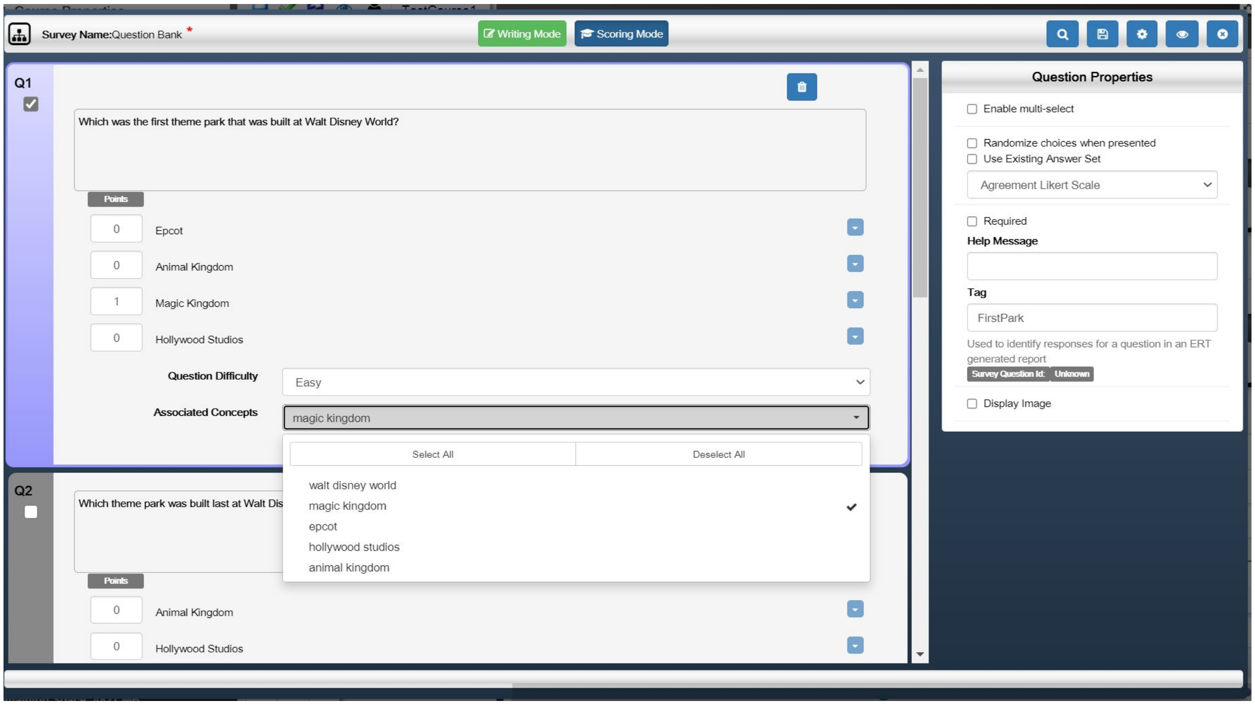
Survey Question Id: Unknown

□ Display Image

Figure 12. Add Survey Item Interface

As you add questions to your question bank make sure to hit the save button which is above question properties. It is also recommended that every once and awhile you exit out of the question bank authoring and hit the save button on the main course as they are saved differently, and even if you save your question bank, if you lose your internet connection before saving the entire course you may lose the questions you were working on.

16) Once you have added your question bank questions you will click on “Scoring Mode” on the top of the interface. This will change the background to blue and allow you to enter scoring information for each question that you generated. See Figure 13 for a screenshot of the interface. You indicate the correct answer by putting the number of points next to it. I have chosen to use “1” point for the correct answer. You then indicate the question difficulty with the drop down menu (Easy, Medium or Hard), and you add an associated concept. Do this for all of your questions. Make sure to save as you go.


Survey Name:Question Bank *

Q1 ☑

Which was the first theme park that was built at Walt Disney World?

Points	
0	Epcot
0	Animal Kingdom
1	Magic Kingdom
0	Hollywood Studios

Question Difficulty: Easy

Associated Concepts: magic kingdom

walt disney world
magic kingdom ✓
epcot
hollywood studios
animal kingdom

Q2 □

Which theme park was built last at Walt Dis

Points	
0	Animal Kingdom
0	Hollywood Studios

Question Properties

□ Enable multi-select

□ Randomize choices when presented

□ Use Existing Answer Set

Agreement Likert Scale

□ Required

Help Message

Tag: FirstPark

Used to identify responses for a question in an ERT generated report

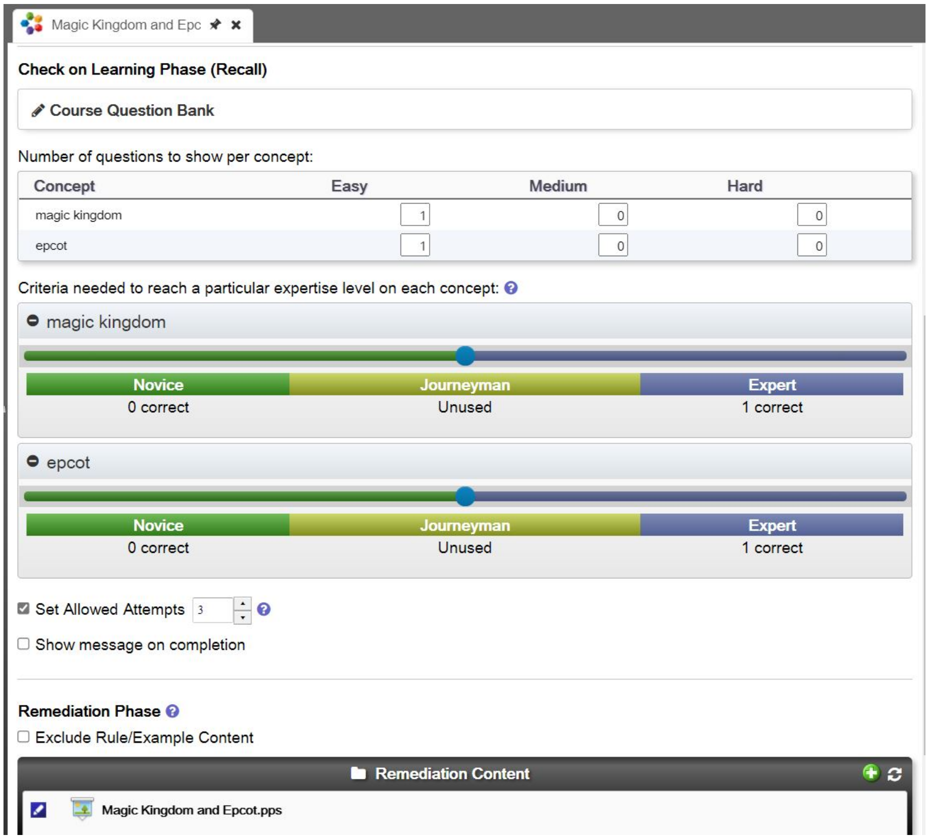
Survey Question Id: Unknown

□ Display Image

Figure 13. Scoring mode interface

It is recommended that you make a note of how many questions you have created for each concept, and the assigned difficulty levels. This will be relevant because you will be setting up the Recall phase to ask a certain number of questions for each concept at different difficulty levels. If you want all of the questions to be asked them put the maximum number you created in the configuration; if you want it to randomly choose a question from that category put in a number that is lower than your maximum.

17) Exit the survey authoring interface, and make sure to save it. Now you will back on the main adaptive courseflow configuration screen. Scroll down to the Recall section (See Figure 14).



Magic Kingdom and Epc

Check on Learning Phase (Recall)

Course Question Bank

Number of questions to show per concept:

Concept	Easy	Medium	Hard
magic kingdom	1	0	0
epcot	1	0	0

Criteria needed to reach a particular expertise level on each concept:

magic kingdom

Novice	Journeyman	Expert
0 correct	Unused	1 correct

epcot

Novice	Journeyman	Expert
0 correct	Unused	1 correct

☑ Set Allowed Attempts 3

□ Show message on completion

Remediation Phase

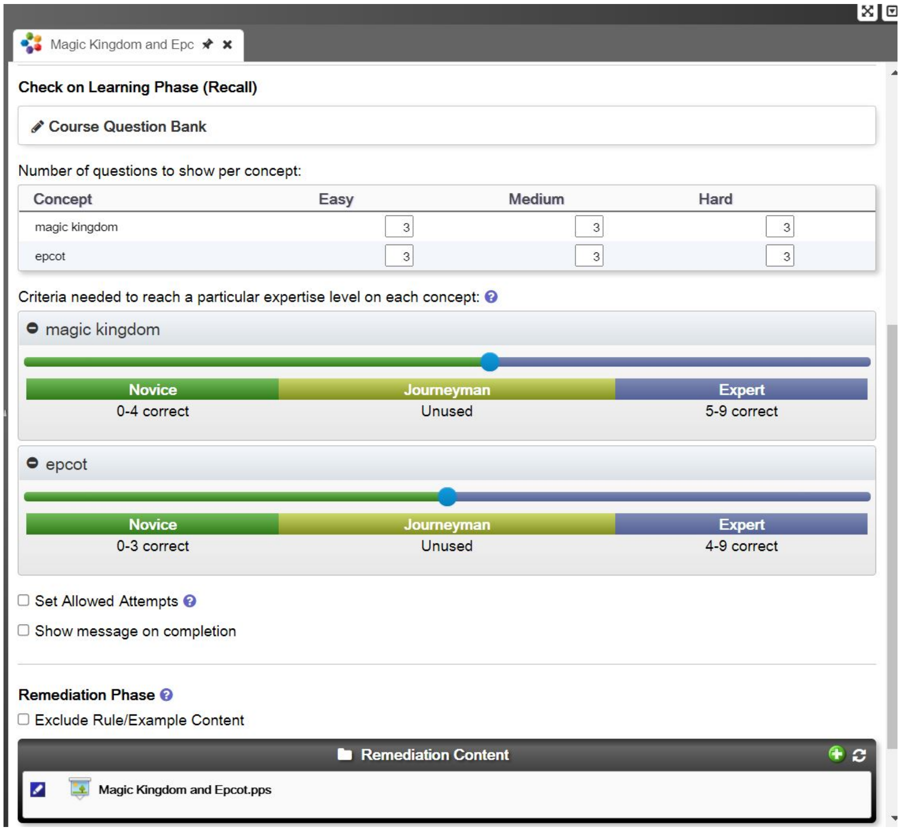
□ Exclude Rule/Example Content

Remediation Content

Magic Kingdom and Epcot.pps

Figure 14. Recall phase default configuration.

18) Now that you have created your question bank you should populate the section that says “Number of questions to show per concept”. Imagine that in our Question Bank we entered a total of 5 easy questions, 5 medium questions, 5 hard questions about Magic Kingdom, and a total of 3 easy questions, 3 medium questions, and 3 hard questions about Epcot. We can change the numbers in the Recall configuration for the concept of Magic Kingdom to 3 easy, 3 medium and 3 hard questions. This would be a total of 9 questions that would be asked based on this concept. The system will randomly choose 3 of the 5 questions at each of those difficulty levels. For Epcot we can set it as 3 easy, 3 medium, and 3 hard questions. This would result in all of our Epcot questions being asked every time. See Figure 15 for our configured version of the Recall phase.



Magic Kingdom and Epc

Check on Learning Phase (Recall)

Course Question Bank

Number of questions to show per concept:

Concept	Easy	Medium	Hard
magic kingdom	3	3	3
epcot	3	3	3

Criteria needed to reach a particular expertise level on each concept:

magic kingdom

Novice	Journeyman	Expert
0-4 correct	Unused	5-9 correct

epcot

Novice	Journeyman	Expert
0-3 correct	Unused	4-9 correct

☐ Set Allowed Attempts

☐ Show message on completion

Remediation Phase

☐ Exclude Rule/Example Content

Remediation Content

Magic Kingdom and Epcot.pps

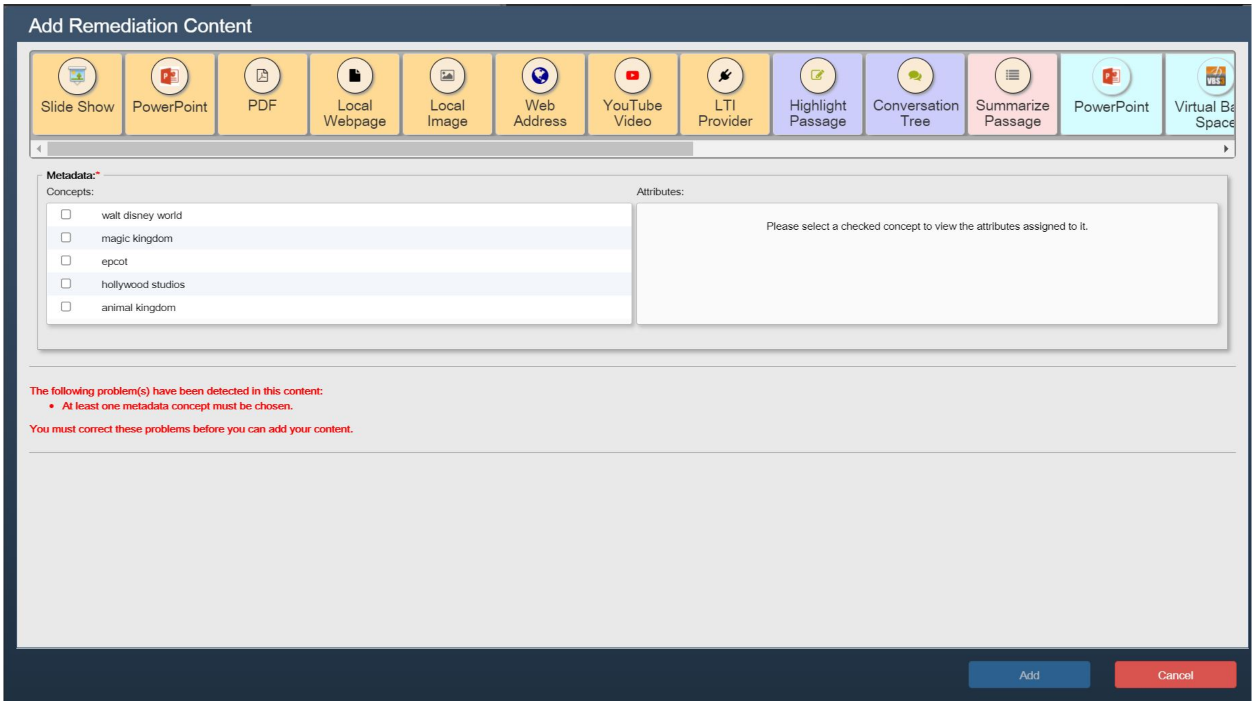
Figure 15. Configured version of the recall phase.

19) Now you will need to adjust the "criteria needed to reach a particular expertise level on each concept". By default in GIFT the learner must reach the "Expert" level to be able to pass that concept and move on without remediation. You can either set it to assess as just Novice or Expert, or add in the category of Journeyman by moving the blue dots to the right or left. It may take a little bit of trial and error to understand it, but you will see the number correct under each of the categories change when you move the blue dot. In Figure 15 I have set it so that if 5 or more questions are answered correctly about the Magic Kingdom the learner can move on without remediation (they are an Expert), and if 4 or more questions are answered correctly about Epcot they can move on without remediation on that topic. I also unchecked the "Set Allowed Attempts" box to allow the learner to continue in the course regardless of the number of times they go through the remediation phase (by default it is set to 3, which will stop of the course after 3 attempts if the learner is consistently not passing the Recall phase).

After the questions are presented and the learner answers them, if they do not meet the Expert threshold on the concepts they will receive remediation. For instance, if this learner got 6 questions correct about the Magic Kingdom, but only 2 questions correct about Epcot they would receive Remediation about Epcot. However, after receiving that remediation they will be asked questions about both concepts again. If they now only get 3 questions right about Magic Kingdom, and 2 questions right about Epcot they will receive remediation on both concepts before going back to the recall phase. Once they consistently meet the Expert level on all of the concepts they be able to move forward.

20) Now that you have configured the Question Bank it is important to set up your remediation options. Remediation is the material that is provided to the learner based on the course concepts that they did not pass successfully. The different types of files that you can use are similar to the file types that you uploaded in the Example phase. However, there are a few more interactive options that are available: Summarize Paragraph, Highlight Paragraph, and Conversation Tree. Additionally, you can use a

training application such as PowerPoint or VBS3 for a remediation activity (this is beyond the scope of the current paper). All of your remediation needs to be tagged with meta-data to indicate the concept and the attributes (see step 12 which is similar). Your remediation materials can cover a single concept or multiple concepts. The GIFT system will select an appropriately tagged item for remediation; if more than one concept is missed it will select remediation items to cover each of them. For example, you could include Magic Kingdom only remediation, Epcot only remediation, and remediation that covers both, you will just need to tag it appropriately. You also have the option of including your Rules and Example concept in the remediation or excluding it if you do not want the learners to see the same exact material again. See Figure 16 for the add Remediation interface.



Metadata:*

Concepts:

- □ walt disney world
- □ magic kingdom
- □ epcot
- □ hollywood studios
- □ animal kingdom

Attributes:

Please select a checked concept to view the attributes assigned to it.

The following problem(s) have been detected in this content:

- At least one metadata concept must be chosen.

You must correct these problems before you can add your content.

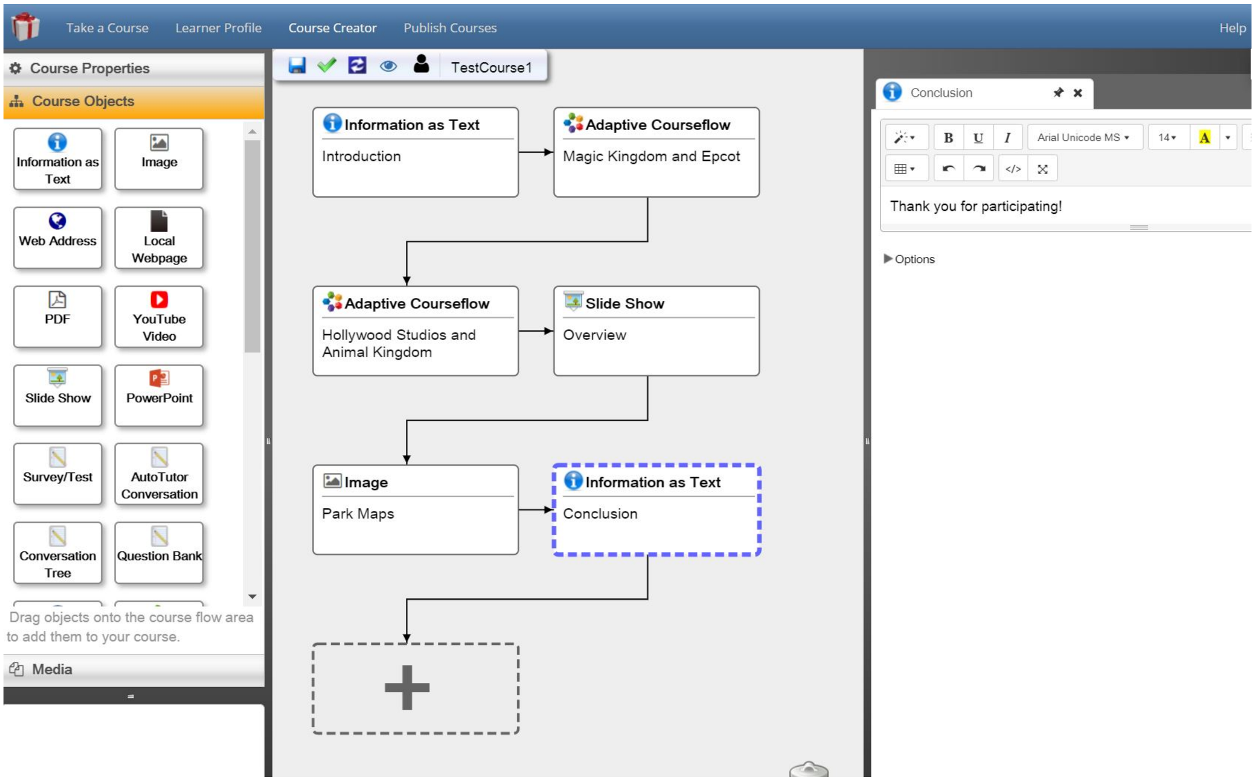
Figure 16. The Add Remediation Content interface.

The Practice Phase is an optional phase that is meant for the learner to engage in an applied activity in an external training application (such as VBS3). It requires additional configuration and is beyond the scope of the current paper. If you do implement it, it will act similarly to the Recall process where learners engage in the scenario and if they fail a specific concept it will provide remediation on it until they completely pass the scenario or until it reaches the maximum number of attempts that you have set.

21) Now that you have completed your adaptive courseflow object make sure to save your main course. You can create another Adaptive Courseflow Object using the same steps (for instance, my next one would be named Hollywood Studios and Animal Kingdom).

Now you can add any additional course objects that you would like to use. If it is not part of an adaptive courseflow that means that everyone who engages with the course will experience the item that you have added. You can imagine the main GIFT courseflow as happening in linear order; when an adaptive courseflow object is added the learner has to complete the entire sequence which we described above, and then they can move on to the next item in the main GIFT courseflow.

To add additional items to your GIFT courseflow drag them from the right side of the screen and drop them on the main courseflow in the center of the screen. You can also reorder them in the courseflow by dragging them. You can add linear questionnaires using Survey/Test (the authoring is similar to the question bank process but will not include associated concepts), Slide Shows, PowerPoints, Local Webpages, Images, YouTube videos, etc. In my example I have added an additional adaptive courseflow, a Slide Show, an image and an information as text. The process of adding these items is similar to the processes used above when adding items. Additional information can also be found in previous guides (Sinatra, 2018; Sinatra, 2019; Sinatra, 2020). See Figure 17 for a screenshot of my completed course.

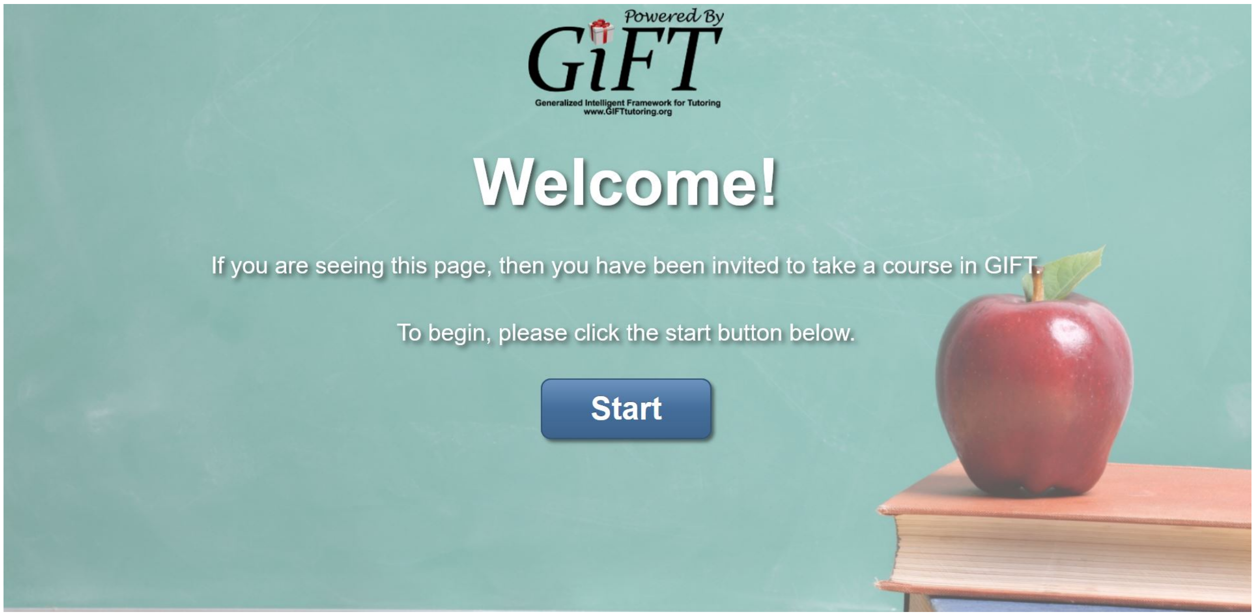


Thank you for participating!

Figure 17. Screenshot of completed GIFT course.

Now that you have completed your course you can preview it by hitting the eye button. You can validate it by hitting the green check button. Validating the course will check to see if it will be able to run. If you are still in the progress of authoring it and have not included all the components it is not likely to properly validate, but you can still save it.

You also need to decide how you are going to distribute your course. You can hit Publish Courses and create a URL which can be provided to learners and gives a direct link to the course (See Figure 18 for what the login screen looks like for a published course). This has advantages because they will not need a GIFT account and you can easily export the data. However, it does not provide a USERID or learner name. You will need to create a Survey item that asks them for this information if you need it. The course publication and data extraction process for shared courses has been documented in detail in the 2020 Research Psychologist's Guide to GIFT (Sinatra, 2020). Please follow up with that guide for detailed instructions on how to publish courses.



Welcome!

If you are seeing this page, then you have been invited to take a course in GIFT.

To begin, please click the start button below.

Figure 18. Login screen for a published course.

If you do not want to publish your course, the other approach that you can use is to Share your course with other learners by going to the tile screen. They will have to have a GIFT user account, and you will need to know what it is and enter it in the box. You can give them full permissions to edit and view it in edit mode, or you can give them permission to only run the course (this is the option that should be used for students). In this version they will login to GIFT and on their tile screen they should see all of their own courses, public courses, and courses that are shared with them. If they do not see it they will need to check the option to view courses shared with them. Current work is being done to improve the ability of a course author to extract data from shared courses, and is expected to be detailed in future guides.

DECISIONS TO MAKE WHEN DESIGNING YOUR COURSE

There are a number of decisions to make when you are designing your course, which will have an impact on the course construction. Remember that the more concepts you have the more material you will need to generate including example content, multiple choice questions, and remediation.

- How do you want to break down your course concepts?
- How do you want to construct your adaptive courseflow objects? Do you want them to cover all your course concepts, or subsets of your course concepts?
- Do you want to create multiple GIFT courses that cover different course concepts?
- Do you want to publish your GIFT course and send learner's a URL, or do you want to share the course with them through their GIFT accounts?

In context of the example in this paper, the answers in regard to my course are:

- I have broken down my GIFT course into 4 concepts: Magic Kingdom, Epcot, Hollywood Studios, and Animal Kingdom.

- I constructed two adaptive courseflow objects that each covered two course concepts (Adaptive Courseflow 1: Magic Kingdom and Epcot; Adaptive Courseflow 2: Hollywood Studios and Animal Kingdom). I could have put these all in 1 adaptive courseflow if I wanted to, or gave them all their own adaptive courseflow.

- I created 1 GIFT course that covered all 4 of my GIFT concepts. If I wanted to split it into separate courses I could have covered Magic Kingdom and Epcot in one GIFT course, and the other two remaining Parks in another one. This would result in each GIFT course taking less time and being able to be taken at different times (since in the current iteration of GIFT a course attempt has to be taken at one time and cannot be paused/returned to with progress saved).

- At current time there are pros and cons to publishing your course vs. sharing your course directly with a learner. Your decision will be based on your goals and if your GIFT course will be a one time interaction with the system, or if your learners will be likely to register with GIFT.

CONCLUSION

GIFT is being continually developed and improved. The steps within this guide are current as of 2021, but they may change slightly in future versions of GIFT. However, the general processes within GIFT, definitions, and course objects are relatively stable and will continue to be in the future. The current paper can be used to assist you as you familiarize yourself with GIFT and as you create adaptive content using adaptive courseflow objects. If you have additional questions about GIFT please review the GIFT documentation, and you can submit questions on the gifttutoring.org forums. As GIFT continues to be developed it is expected that additional features will improve both the authoring and data extraction experiences, which will hopefully be documented in future guides.

References

Sinatra, A. M. (2014). The research psychologist's guide to GIFT. In Proceedings of the 2nd Annual GIFT Users Symposium (pp. 85-92).

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

Sinatra, A. M. (2016). The Updated Research Psychologist's Guide to GIFT. In Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym4) (p. 135).

Sinatra, A. M. (2018, May). The 2018 Research Psychologist's Guide to GIFT. In Proceedings of the Sixth Annual GIFT Users Symposium (Vol. 6, p. 259). US Army Research Laboratory.

Sinatra, A. M. (2019, May). The 2019 Instructor's Guide to GIFT. In *Proceedings of the 7th Annual GIFT Users Symposium* (p. 19). US Army Combat Capabilities Development Command–Soldier Center.

Sinatra, A. M. (2020, May). The 2020 Research Psychologist's Guide to GIFT. In *Proceedings of the 8th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym8)* (p. 23). US Army Combat Capabilities Development Command–Soldier Center.

Sottilare, R. A., Brawner, K. W., Sinatra, A. M., & Johnston, J. H. (2017). An updated concept for a Generalized Intelligent Framework for Tutoring (GIFT). GIFTtutoring. org.

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S Army Combat Capabilities Development Command – Soldier Center – Simulation and Training Technology Center. The statements and opinions

expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHOR

Dr. Anne M. Sinatra *is part of the adaptive training research team within US Army DEVCOM Soldier Center STTC, and works on the Generalized Intelligent Framework for Tutoring (GIFT) project. Her background is in Human Factors and Cognitive Psychology.*

Authoring Collective Training Demonstrations in GIFT, 2021 Update

Elyse Burmester
Dignitas Technologies

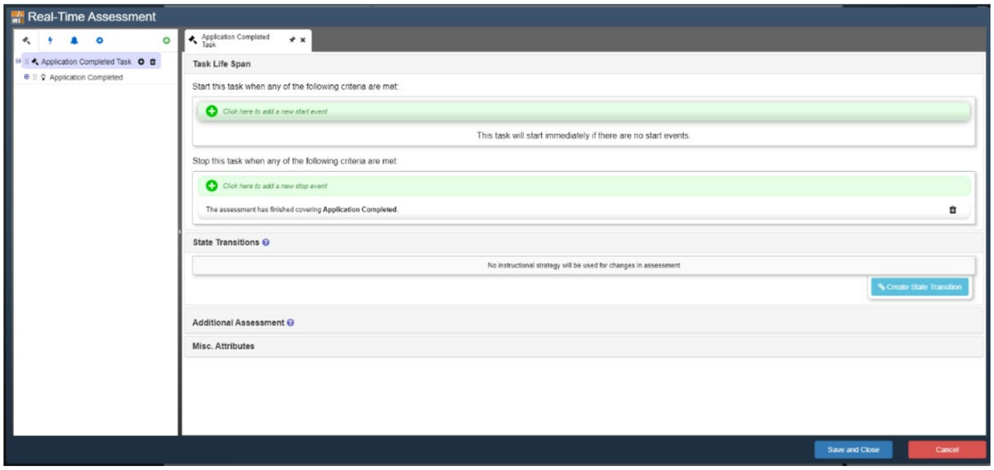
Introduction

The Generalized Intelligent Framework for Tutoring (GIFT) has served a number of unique purposes since its release in May of 2012, but the most recent shift of interest in GIFT's usage has been focused on collective training in virtual training environments. This paper was written to serve as a beginners' guide for those looking to use GIFT to improve their team-based training exercises. As a frequent developer of collective demonstrations using the training applications integrated with GIFT, I have developed the streamlined process provided in this paper to create training content quickly and effectively. While GIFT has several assessment tools in its suite, this paper will focus only on the Real-Time Assessment tool, which is powered by the Domain Knowledge File (DKF). Recent updates to GIFT will also be discussed.

DKF Overview

The Domain Knowledge File (DKF) is an XML file that provides all the necessary information in order to execute a lesson with an external training application. DKFs are read by the Domain module and used by the Domain and Pedagogical modules. This schema contains elements such as scenario name, learner identification, waypoints, assessments, etc. You can edit a DKF using a text editor (although this is not recommended for non-development GIFT users) or the more popular DKF Authoring Tool.

When you open the DKF editor, also called Real-Time Assessment, your screen will look similar to Figure 1. The panel on the left-hand side of this window serves as the root menu of the editor. It has four tabs available at the top, located just under the words "Real-Time Assessment". Each tab contains different elements of the DKF. Starting from the left-hand side they are labeled: "Tasks", "Strategies", "State Transition", and "Assessment Properties". The green plus sign located to the right of this list is used to add new elements within each tab, excluding "Assessment Properties". We will explore each tab in detail in the following sections.



Start this task when any of the following criteria are met:

This task will start immediately if there are no start events.

Stop this task when any of the following criteria are met:

The assessment has finished covering **Application Completed**.

No instructional strategy will be used for changes in assessment

Figure 1. Task Authoring Panel.

Tasks and Concepts

Tasks and concepts outline the assessments that are required in a scenario. GIFT has a number of automated performance measure options, or Performance Nodes, that are available to use with most training applications. A concept is the lowest level performance node and is associated with a java class that contains logic to assess the learner's actions in the domain. A concept is assessed via condition classes. Condition classes are responsible for consuming a game state message and then providing an assessment based on the information contained within that message. The concept/condition hierarchy supports infinite nesting by specifying the "concepts" choice for a concept instead of "conditions".

Tasks have a lifecycle that must be defined by Start and End triggers, found in the Task Life Span panel when a parent task is selected from the list. These triggers are used to structure the flow of your DKF – the concepts listed under each parent task will only remain active for the duration of the parent tasks' lifecycle. The Start and End triggers are defined using either: 1) the learners location in the simulated environment, 2) the learners completion of a task or concept, 3) the learners performance on a concept, 4) when a Learner Action is selected (explained further below), or 5) when a strategy is applied.

Strategies

GIFT has a number of adaptation options that can be automatically applied by the system when a learner state is identified based on the defined parameters. These are called "Strategies" in the DKF architecture. Strategies are sent from the Pedagogical module and implemented by the Domain module based on changes in learner state.

Strategies are mainly used to adapt the learners experience for instructional support but they can now also be used as a Task start or end trigger. Some strategy options currently available include: sending a feedback message, presenting a survey, displaying a picture or video, and modifying the virtual training environment.

State Transitions

State transitions cause instructional strategies to be sent to the learner based on assessment values defined in a Performance Node. A state transition is "activated" when a logical expression evaluates to true based on the specific learner state attribute(s) changes from the previous value to the current value. There are three evaluation criteria options to choose from: an authored Task, an authored Concept, or a possible Learner State.

Assessment Properties

The final elements of the DKF are found under the Assessment Properties tab in the left-hand panel. Two of these properties are necessary to tie your DKF together: Points of Interest and Team Organization. The other properties - End Triggers, Learner Actions, and Miscellaneous – provide additional customization options but are not usually required.

Team Organization

This section defines the hierarchy of teams and team members within your scenario. Both teams and team members can be referenced in various parts of the DKF, such as strategies and conditions. This is beneficial when assessing multiple teams at once, as well as assigning assessments or strategies to a specific team or team member to separate responsibilities. Each level of this hierarchy must have a unique name and this list must contain at least one team member for the learner to link to.

Places of Interest

This section defines the global list of points, paths, or areas associated locations in a virtual environment, e.g. VBS3. These waypoints can be referenced throughout the DKF in tasks, concepts, and conditions. All waypoint name values must be unique within the DKF. Locations can be specified in either Geocentric Coordinates (GCC), Above Ground Location (AGL), or other coordinate systems depending on the need and application being used.

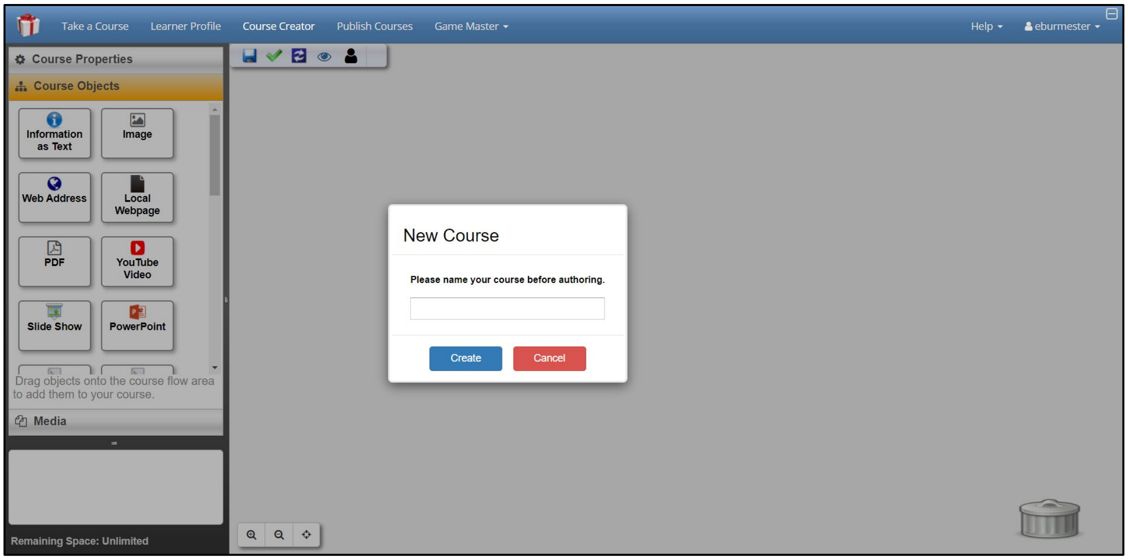
Authoring a DKF

Now that you are familiar with the main components of a DKF, you can follow the steps below to create a DKF on your own. This guide is designed to benefit new and current GIFT course developers to embrace the full benefit GIFT offers when integrated with external training applications with minimal programming knowledge. It is intended to be simple and easy to use - it will not discuss advanced features of the DKF, such as course concepts. Readers are encouraged to explore the documentation provided on gifttutoring.org for more in-depth, technical information.

In order to access the full capabilities in this guide, you will need to run GIFT locally using the desktop version. The DKF Authoring tool can be accessed in GIFT Cloud but at the time of this papers publication training applications can only be used in the desktop version of GIFT.

1. Start by launching GIFT and logging in with your GIFT account. Once the Take a Course page loads, click on the "Course Creator" tab at the top of the webpage. Enter a useful course name and save it.

2. Scroll through the list of Course Objects on the left side of the webpage (shown in Figure 2) and find the appropriate training application for your scenario. For the purpose of this guide, we will use the Virtual Battlespace course object.



Drag objects onto the course flow area to add them to your course.

Remaining Space: Unlimited

New Course

Please name your course before authoring.

Figure 2. Create a Course page

3. Add all of the known entities within your scenario to the Team Organization list found under the "Assessment Properties" tab (shown in Figure 3). Consider also adding objects or entities that may not be played by a human but might be utilized as a trigger for some event in your scenario.

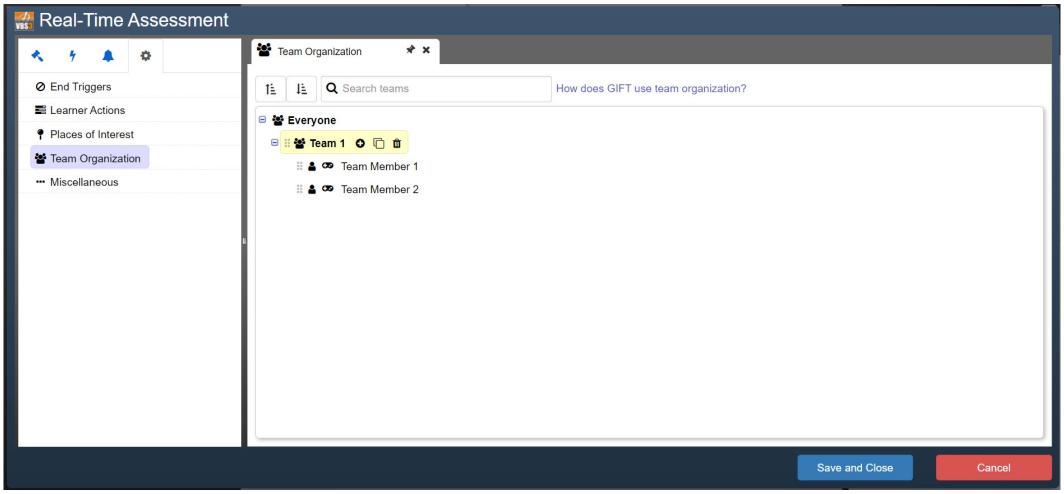
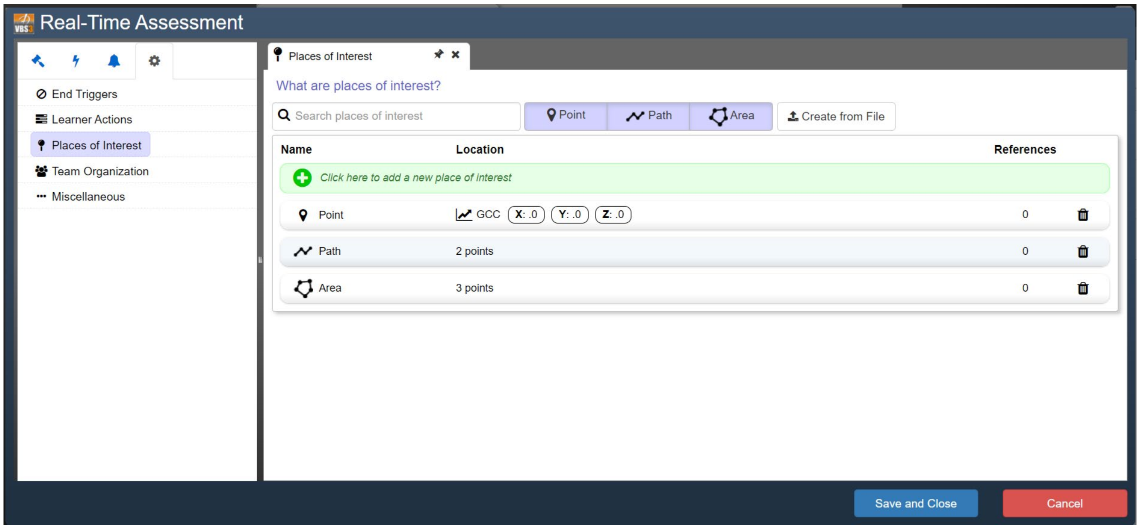


Figure 3. Team Organization Panel

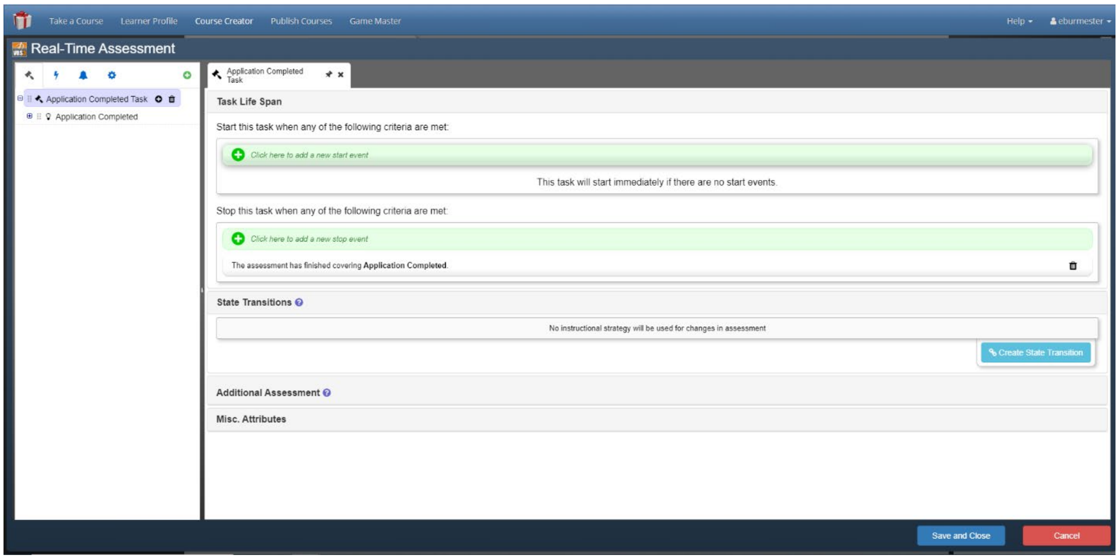
4. After all of your entities are accounted for, open the Places of Interest tab in the Assessment Properties menu (shown in Figure 4). Use the green plus sign to add all relevant locations within your scenario - locations might be used to trigger events or provide context for an assessment.



Name	Location	References
Point	GCC X: .0 Y: .0 Z: .0	0
Path	2 points	0
Area	3 points	0

Figure 4. Places of Interest Panel

5. Now you are ready to populate the Tasks and Concepts. Open the Task panel by clicking on the hammer icon in the top left corner of the screen. Add an item to this task list using the green add button located next to the tabs at the top of the panel (see Figure 5). This will add a parent task to the list (indicated by a hammer icon). To add a concept (indicated by a lightbulb icon) under a parent task, use the grey add button located next to the parent tasks' name.



Start this task when any of the following criteria are met:

This task will start immediately if there are no start events.

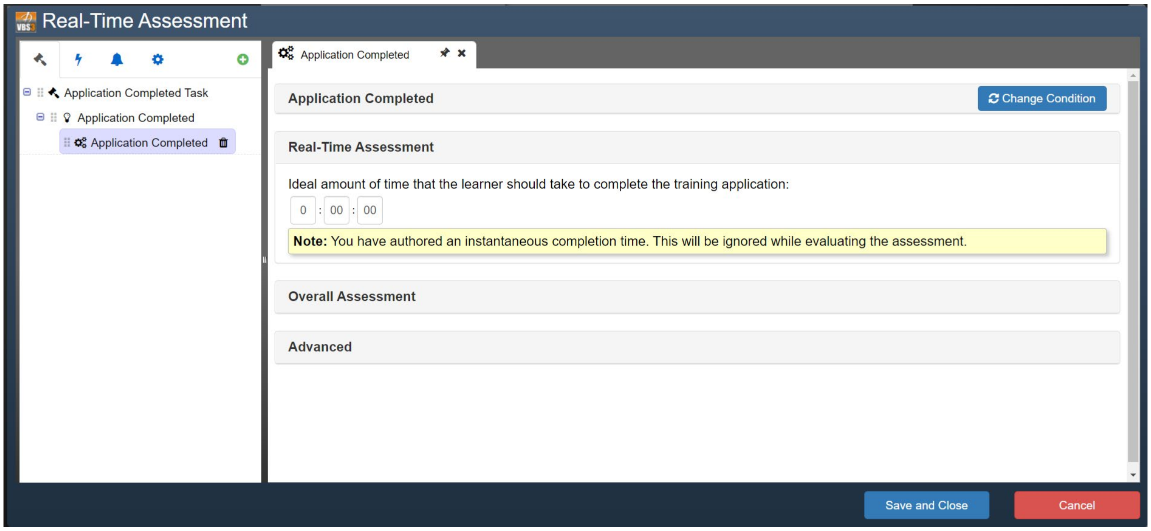
Stop this task when any of the following criteria are met:

The assessment has finished covering **Application Completed**.

No instructional strategy will be used for changes in assessment

Figure 5. Task Authoring Panel.

6. When a concept is added, a second level node will appear below that concept and a list of condition classes will be shown in the panel on the right-hand side. Click through each condition class in this list to find detailed descriptions for the associated assessment logic and evaluation values that are used to drive that conditions' logic. After selecting a condition, enter the appropriate values in the "Real-Time Assessment" section to define the desired assessment logic. The "Overall Assessment" and "Advanced" sections within this panel contain additional options to customize assessments, but these options are not required for validation. An example of a condition class can be seen in Figure 6. Repeat steps 3 and 4 for all assessments required in your scenario.



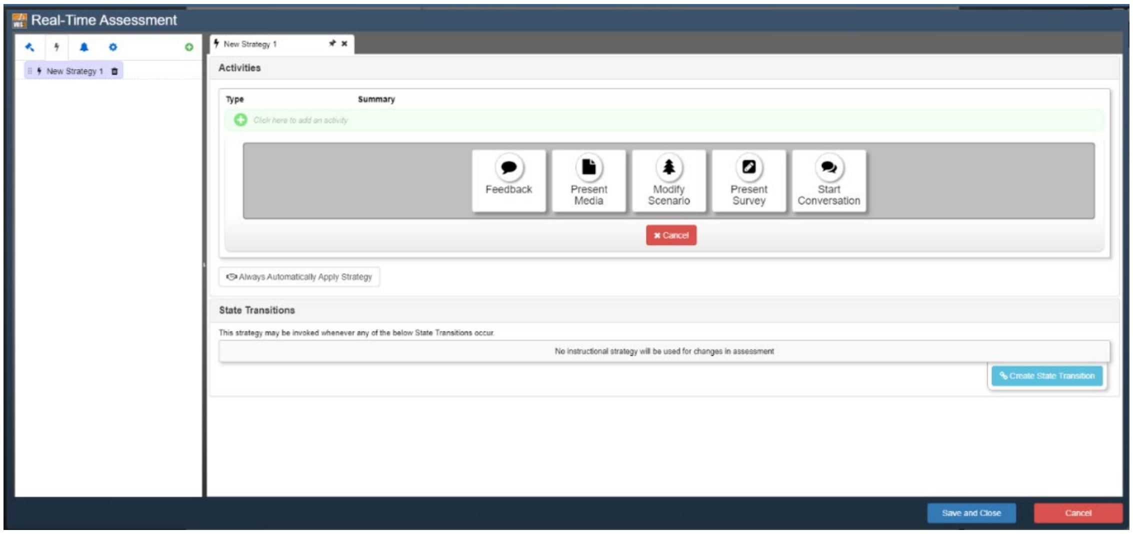
Ideal amount of time that the learner should take to complete the training application:

0 : 00 : 00

Note: You have authored an instantaneous completion time. This will be ignored while evaluating the assessment.

Figure 6. Application Completed Condition Class

7. After adding all of your tasks and concepts, click on the Strategies tab (shown in Figure 7). Use the green add button at the top of the left-hand panel to add each new strategy. Unlike the task list though, I recommend adding strategies individually and linking each with a state transition before moving on to author the next strategy. After naming the strategy, click on the green bar in the "Activities" table to select an activity for the strategy to implement.

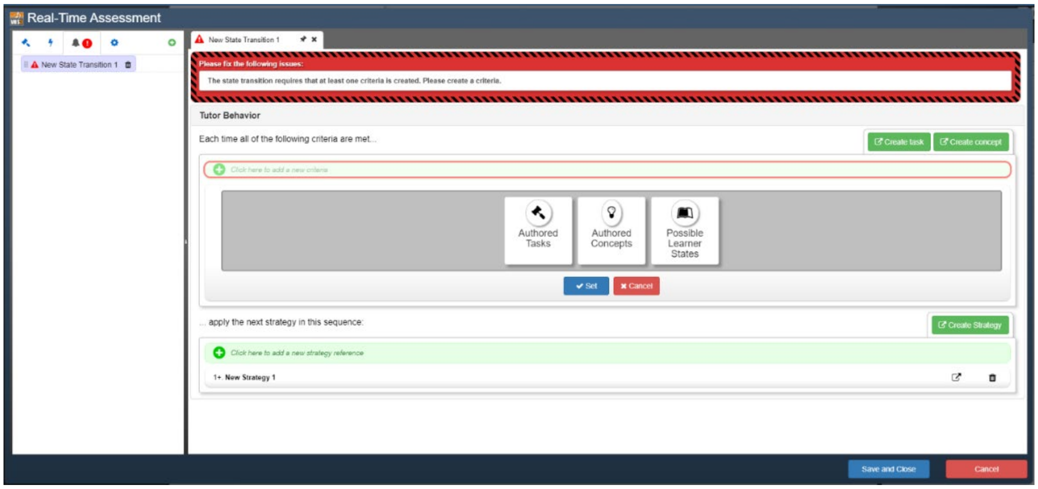


This strategy may be invoked whenever any of the below State Transitions occur.

No instructional strategy will be used for changes in assessment

Figure 7. Strategy Authoring Panel

8. To author a state transition from this strategy panel, click on the blue button labeled "Create State Transition" below the "State Transitions" section of this panel. This will open a new window, replacing the previous strategy authoring window, in the State Transition tab with the new state transition panel ready for authoring (shown in Figure 8). (Tip: select the push-pin icon in the right side panel top tab to lock the panel, this will cause the new panel to appear in a new top tab while keeping the previous top tab open.)



Please fix the following issues:

The state transition requires that at least one criteria is created. Please create a criteria.

Each time all of the following criteria are met...

... apply the next strategy in this sequence:

1+ New Strategy 1

Figure 8. State Transition Panel

9. Notice the strategy table at the bottom of the window is already populated with the strategy created from the previous window. You can add additional strategies that will be applied by this state transition using either the green bar in the table to select an existing authored strategy or by clicking the "Create Strategy" button in the top left hand corer of the strategy table to create a new strategy. Strategies in this table will be applied in the order that they appear.

10. You have now authored a complete assessment loop for one task or concept, depending on which you chose for the state transition. Repeat these steps for each event to complete the core assessment logic for your training scenario.

Now that you have all of the pieces of your DKF put together, it is very important to perform as many iterations of testing as necessary to achieve your desired instructional flow. Very rarely does a DKF work as expected the first time it is tested. For example, you may notice feedback that is not presented at the proper time. This could be due to incorrect values provided in the state transition associated with that strategy. Or, if that state transition is not being triggered by the expected learner performance state this could be due to incorrect values provided in the associated concepts condition logic. Keeping in mind where values are defined or what dependencies they are linked to will help pinpoint where in the DKF certain modifications are needed to reach the expected outcome.

Recent Updates in GIFT

Several updates have been made in the last year pertaining to elements of the DKF. Customer requirements and community feedback are the main driving factors for GIFT updates. Here are some new features that have been added in the realm of Real-Time Assessment:

- Strategies can now be used as Task start and end triggers. Using this feature, you can control the lifespan of a task based on an event unrelated to a performance node, opening up even more possibilities for DKF structuring.

- New condition classes were created to provide more automated assessments. The newest condition classes are Muzzle Flagging, Engage Targets, Detect Objects, Assigned Sector, Fire Team Rate of Fire, and Request External Attribute. For more information on these condition classes, check out this years' GIFT Architecture and Features Update (Hoffman et al, 2021).

- New scenario adaptation strategy types, Breadcrumbs and Highlight Object, were added to extend the actions available within the virtual environment. The Breadcrumbs strategy places an visual indicator in the virtual environment showing the distance between the learner and a place of interest. The Highlight Object strategy places an arrow on or around either a team member or a place of interest in the virtual environment.

- The "Miscellaneous" Assessment Properties tab now contains inputs for Mission Details, which provide scenario specific information such as Situation, Goals, Weapon Status, etc. Mission details are included in the domain session output after the training is complete and can also be referenced in the Game Master interface.

- Tasks and concepts can now be weighted to influence the assessment of a parent node. These assessment rollup rules provide authors with greater flexibity in the structure of their assessments, as well as their intended outputs.

CONCLUSION

Authoring a DKF can seem like a daunting task, especially when you consider all of the elements involved and the level of complexity that is possible. While this paper only touches on the basic DKF functionality and capabilities, it is intented to serve as a user-friendly introduction to the full-suite of tools available with this software. Whether you are looking to improve your current team-based training methods or serving some other domain, GIFT has the tools to help alleviate instructor workload and improve training outcomes.

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S Army Combat Capabilities Development Command – Soldier Center – Simulation and Training Technology Center. The statements and opinions

expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHOR

Elyse Burmester *is a Test Engineer at Dignitas Technologies. Since 2017, she has contributed as a member of the GIFT development team in addition to directly supporting the Creative and Effective Training Technologies Branch at U.S. Army Combat Capabilities Development Command - Soldier Center (CCDC-SC) SFC Paul Ray Smith Simulation & Training Technology Center (STTC). Ms. Burmester has a bachelor degree in Political Science from Florida Gulf Coast University.*

Theme II: Standards for Adaptive Instructional Systems and Competency Modeling

Instrumenting GIFT with xAPI: a use case for IEEE P9274.3.x standards activity and implications for the broader field of ITS and AIS

Shelly Blake-Plock, Will Hoyt, & Cliff Casey
Yet Analytics

INTRODUCTION

There are clear benefits to auditability, clarity, and interoperability when activity data emitted by the Generalized Intelligent Framework for Tutoring (GIFT) is modeled on the Experience API (xAPI) data specification.

As part of the STEEL-R project, an xAPI Profile defining the vocabulary, patterns, sequences, and concepts relevant to GIFT is currently under development which is designed to model the behavior of GIFT-enabled adaptive instructional systems (AIS) as xAPI activity data.

This capability will provide the opportunity to leverage the benefits of an xAPI Profile germane to the specifics of GIFT while laying the foundation of an xAPI Profile more generalizable to the domain of intelligent tutoring systems (ITS). It will ultimately be useful to the broader marketplace of adaptive instructional systems.

The xAPI Profile[1] under development for GIFT (which we'll shorthand as the GIFT Profile) and the more generalized xAPI Profile for intelligent tutoring systems (which we'll shorthand as the ITS Profile) are each examples of xAPI Profiles which represent use cases which can be defined and standardized within the construct of related xAPI standards activity at the IEEE.

PROBLEM

There is no standard way to ensure the auditability, clarity, and interoperability of the activity data emitted by adaptive instructional systems. As a result, the users of such systems experience the detriments of "black box" syndrome, untenable technical debt, and vendor lock-in. The lack of accountability in such a scenario is only counterbalanced to the wealth of additional cost and exposure.

A solution investigated in the STEEL-R project has been the development of an xAPI Profile for GIFT which can meet the needs of ensuring validated and auditable activity data, clear meaning and machine-readable data design, and interoperability with applications and systems across deployments of learning ecosystems. The implementation of xAPI Profiles provides a means of capturing a range of GIFT activity data representing constituent domains of administrative, pedagogical, and learner behavior (Blake-Plock, et al., 2020).

Events modeled within an xAPI Profile may leverage not only the resources necessary to define verbs and activities, but also those required to map out patterns, sequences, and concepts relevant to the possible learner pathways made available by engagement in the learning experience. The activities that occur during these engagements may represent a range of system-level activities as well as pedagogical and technical approaches to instructional delivery, assessment, and competency assertion. Whether cognitive-based, constraint-based, natural language based, or what have you, the purpose of the xAPI Profile is to provide a logical model of potential experiences in machine-readable format based on the design of those experiences, including the

[1] https://github.com/adlnet/xapi-profiles

pathways of learners through the experiences and—in the case of ITS—the interaction between learner and AI.

All reasoning and results are baked into the design of what is being modeled as opposed to being prescribed independently by the xAPI Profile. In this way, the Profile helps to standardize the architecture of the experience, but is agnostic with regard to the pragmatics of the experience. An advantage of this approach is that the xAPI Profile itself can then serve as the foundational text against which the activity collected as xAPI data statements may be validated and ultimately audited. The necessity for an xAPI instrumentation for GIFT resulted from the desire within the STEEL-R project to create an auditable chain of activity-to-assertion between GIFT-based activities on the one hand and CaSS[2]-based conclusions on the assertion of competency on the other. xAPI was chosen as the data model to link the two, but there was not an xAPI Profile available to serve the modeling needs of the program.

BACKGROUND

An xAPI Profile describes the range of xAPI Statements produced by an xAPI instrumentation. However, xAPI Profiles are not typically incorporated into the source code that constitutes an xAPI instrumentation.

Within the context of the STEEL-R project, GIFT's existing xAPI instrumentation had to be updated to include xAPI Profile Concepts and Statement Templates as "first-class citizens" within the data regime. xAPI Profile Concepts—the Verbs, Activity Types, Activities, Extensions, and other attributes that define a type of activity or behavior within the context of an activity domain—now would serve as the anchor for converting elements of the GIFT data model into their xAPI representation. Meanwhile, xAPI Profile Statement Templates would mandate how those elements were combined into specific xAPI Statement shapes as produced by the instrumentation.

Incorporating the GIFT xAPI Profile into the GIFT xAPI instrumentation eliminates the disconnect that typically exists between an xAPI Profile and the domain it describes. It accomplishes this by enforcing the xAPI Profile Statement Template determining property requirements within xAPI Statement generation as opposed to performing Statement Template conformance validation after the fact via an xAPI Profile Processing Library.

SOLUTION

The resulting solution is a leaner and more tightly integrated relationship between GIFT activity and the semantics that describe it at the machine-readable level. While recognizing that GIFT is only a specific framework allowing for a specific expression of AIS, this GIFT Profile can nonetheless be used as the basis of an abstract or generalized xAPI Profile relevant to ITS more broadly, as was the suggestion of the aforementioned research. Whereas the current project focuses on the ability of GIFT to evaluate and make conclusions about learners within a GIFT-moderated learner experience, the reporting on instances of evaluation as well as the conclusion drawn from the summation of those instances are generalizable to the broader range of ITS technologies. Furthermore, and more relevant to thinking about GIFT at scale, is that by instrumenting GIFT with xAPI, we are able to design the relationship between GIFT-moderated activity and the transactional requirements of the Master Object Model (MOM) Profile[3]. This is the necessary link when addressing how ITS-at-large can fit into the broader data flow that hydrates the component capabilities of the Total Learning Architecture.

[2] Competency and Skills System https://adlnet.gov/projects/cass/

[3] https://github.com/adlnet/MasterObjectModel

In an attempt to provide a succinct account of the process of xAPI generation with GIFT:

- As a domain session is requested, data is cached for use in the creation of the xAPI statements
- As a domain session starts, an xAPI statement is generated which identifies the user and the course selected
- As a knowledge session is created, one or more xAPI statements indicate that either a session host created and started a session or that upon creating the session lobby, other users joined the lobby and then the host started the session
- As an updated request passed through a knowledge session, the team position of the session member provides information for use in the creation of xAPI statements
- The knowledge session begins for the team and statements are emitted
- The learner state is derived from the relevant GIFT components regarding: cognitive state, affective state, and performance state; user interaction within the course causes an update to these attributes
- As formative assessment is completed, a request is made to publish the lesson score and summative results are recorded as xAPI statements
- The session is closed of an xAPI statement is emitted indicating that the user has exited the course

The xAPI statements themselves indicate the team member, the structure of the team, peers among team members, membership within the knowledge session, the parameters of the mission, and individual results. The benefit to analytics includes the capture of the range of performance states including data necessary to assert confidence and competence along with trends, observations, assessment explanations, indications of authoritative resources, and short-term, long-term, and predicted assessment. All activity is captured in a manner that can distill to the requirements of the MOM Profile for the purpose of hydrating downstream processes. Within the Total Learning Architecture, this is accomplished as the data flows between Edge LRSs and a Transactional LRS governed (or mediated) by the MOM Profile. xAPI Statements retrieved from or forwarded from an Edge LRS must conform to a statement template present within the MOM Profile in order to pass through to the Transactional LRS. In this way, the MOM Profile controls data flow. It specifies the minimal requirements for statement shapes.

Within the GIFT Profile, the MOM Profile's "assessed" statement template is extended to include GIFT specific elements such that matching statements can pass from Edge LRS to Transactional LRS within a TLA implementation but are still distinguishable from other MOM Profile conformant xAPI Statements. Further, they can be identified as such using the statement templates found within the GIFT Profile. Likewise, the abstract or generalized ITS Profile would enjoy a similar relationship with the MOM Profile whereby ITS and AIS-specific statement templates would extend the MOM Profile statement templates via the addition of determining properties that describe the distinct aspects of common actions or events within an xAPI-instrumented ITS or AIS applications not expressly defined within any existing xAPI Profile. This could lead to the ability to rapidly develop extended ITS analytics capabilities.

CONCLUSION

Clarity in the definition and handling of team training components as well as clarity in the means of initializing learner state—and thus managing its influence on both macro and micro adaptations—are key to the success of the GIFT and ITS Profiles and are elements that may not have been considered originally in the design of xAPI. In the present use case, semantic clarity is maintained via the funneling of data flow through the governance of control Profiles. Herein the system is designed to provide legitimate semantic interoperability as opposed to what has been described as mere syntactical compatibility (Johnson, et al., 2017) in an effort to foster agreement between machines not only in regards to the shape of the data, but with accountability as to what the data mean with respect to the broader learning ecosystem.

Interoperability itself is maintained through strict adherence to conformant xAPI. The simplest way then to describe the relationship between the parts necessary to ensure the auditable, clear, and interoperable throughput of GIFT activity data within a TLA implementation is that:

- the GIFT xAPI instrumentation incorporates xAPI Profiles into its targets
- the specific approach to xAPI Profiles is aligned to the Master Object Model—in that the goal is for the downstream Transactional LRS to govern data flow and to permit the data necessary for processes such as competency assertion by CaSS
- the ITS Profile itself is therefore an extension of the MOM Profile
- the GIFT Profile is an implementation of the ITS Profile

The result is a set of GIFT activity data that is auditable, maintains clear and machine readable semantics, and is interoperable among other applications and business processes within an implementation of the Total Learning Architecture. Note also what this data set is not. It is not composed of telemetry data or data that may be better handled by event streams parallel to xAPI emitted by the system. The data captured as xAPI has been done so with express purposes to capture and make available the trackable activity of learners in their interaction with GIFT which may be used to drive specific processes. In doing so, GIFT engagement becomes a cost-effective and highly useful means of generating the data necessary to support the assertion of competencies and to drive the successful deployment of downstream business processes.

Additionally, because “learner portability” has replaced “content portability” (Robson & Barr, 2018) as the mover of learning technology paradigms, it is relevant to note that the data output of the former is the dynamic event-based data modeled by xAPI in contrast to the comparatively static data represented by earlier specifications and standards. A system such as GIFT can thrive in the era of learner portability so long as it is able to provide the core capabilities in auditability, clarity, and interoperability expected of such a system.

This new paradigm bears its own cautions, however.

Because xAPI and the instantiation of xAPI activity within Profiles is agnostic regarding the design of elements that comprise it so long as that design can be validated as conformant xAPI by a learning record store, it is essential that the Profiles designed accurately reflect the intention of the author. In the case of AIS, this could provide an argument for the broader community to develop AIS design standards that will enable developers to effectively leverage xAPI Profiles—and the broader world of semantic Linked Data vis-a-vis the specification’s JSON-LD representation. But the AIS capabilities modeled and expressed in the xAPI Profile need to be cognizant of how they appear as metrics downstream (otherwise, we lose the clarity that was the original intention) and the xAPI Profiles and the statements which comprised activity representative of xAPI Profiles needs to be validated conformant with the language, patterns, sequences, and concepts present within the Profiles themselves.

Nonetheless, the benefit is clear: better AIS design produces experiences that lend themselves to the emission and capture of better event-based xAPI data which brings with it the ability to fully leverage the xAPI Profiles specification and therefore machine-readable capabilities in the automation of validation, storage, and reporting functions directly based on the patterns, sequences, and concepts of the modeled experiences themselves. The outcomes is an auditable record of the learning experience itself (perhaps both human and machine)—whereas lacking the semantic parameters of xAPI Profiles, a data log would otherwise be able to represent AIS and ITS telemetry and completion data but not in a way that is elegant (and auditable) in its connection to the representation of context available through semantic and Linked Data resources.

The 9274 suite of xAPI standards activity represents an open source effort within the IEEE to deliver a portfolio of technical standards representative of the needs of xAPI stakeholders. The baseline xAPI standard—directly based on the xAPI 1.0.3 data specification developed by the Advanced Distributed Learning Initiative—is called P9274.1.1 and the subsequent components of the suite are scheduled to include:

- P9274.2: xAPI Profiles (the specification and model itself)
- P9274.3.x: individual xAPI Profiles (such as domain and media specific applications such as the Video Profile, the cmi5 Profile, or the Master Object Model Profile)
- P9274.x: Recommended practices such as regards technical implementation and cybersecurity matters relevant to xAPI and LRS deployments

Within this suite of standards, the GIFT Profile and the more general ITS Profile would fall under the 3.x category. In addition to demonstrating the structure and usability of the 9274 suite, the categorization of these profiles as tiered by specific-to-general application follows the recommendations of research into how ITS profiles could best align with metric and visualizable outcomes.

The definitions of such vocabulary, patterns, concepts, and statement templates as needed to accomplish what is described above is planned to be standardized within IEEE 9274.3.x documentation. The significance of this standardization would be the ability to use the xAPI Profile model to augment, enrich, and enhance the capabilities of GIFT itself as well as any ITS or AIS.

REFERENCES

Blake-Plock, S., Hoyt, W., Casey, C., & Zapata-Rivera, D. (2020). Data Analytics and Visualization for

Brawner, A. Sinatra, & B. Goldberg (Ed.). Proceedings of the Sixth Annual GIFT Users Symposium: US Army Research Laboratory.

Johnson, A., Nye, D.B., Zapata-Rivera, D., & Hu, X. (2017). Enabling Intelligent Tutoring System

Tracking with the Experience Application Programming Interface (xAPI). In R. Sottilare, A. Graesser, X. Hu, and G. Goodwin (Ed.). Design Recommendations for Intelligent Tutoring Systems: Volume 5 - Assessment Methods: U.S. Army Research Laboratory.

Robson, R., & Barr, A. (2018). Learning Technology Standards - the New Awakening. In R. Sottilare, K.

xAPI Learning Data: Considerations for a GIFT Strategy. In Sinatra, A., Graesser, A., Hu, X., Goldberg, B., & Hampton, A. (Ed.). Design Recommendations for Intelligent Tutoring Systems Volume 8 - Data Visualization: US Army Combat Capabilities Development Command – Soldier Center.

ABOUT THE AUTHORS

Shelly Blake-Plock, *is President and CEO at Yet Analytics, Inc. He is an officer of the IEEE Learning Technology Standards Committee.*

Will Hoyt, *is a data engineer at Yet Analytics, Inc. where he has led the development of the GIFT Profile.*

Cliff Casey, *is Chief Technology Officer at Yet Analytics, Inc.*

Recommendations for GIFT from The Learner Data Institute

Vasile Rus
The University of Memphis

Introduction

The Learner Data Institute (LDI) is an NSF-funded project whose mission is to *harness the data revolution (HDR) to further our understanding of how people learn, how to improve adaptive instructional systems (AISs), and how to make emerging learning ecologies that include online and blended learning with AISs more effective, efficient, engaging, equitable, relevant, and affordable* (Rus et al., 2020). LDI is part of the NSF's Harnessing the Data Revolution (HDR) Institutes effort. HDR Institutes enable breakthroughs in science and engineering through collaborative, co-designed programs to formulate innovative data-intensive approaches to address critical national challenges, which in our case is offering access to quality and affordable education to everyone. Indeed, we aim to transform the learning ecosystem via a framework for science convergence based on innovative data science approaches coupled with transdisciplinary, collaborative, and co-designed research and development.

In this paper, we will emphasize three major LDI thrusts which are relevant U.S. Army' Generalized Intelligent Framework for Tutoring (GIFT; Sottilare et al, 2012): (1) *scaling up AISs horizontally* – across topics and domains; (2) *scaling up AISs vertically*, i.e., targeting higher-level, advanced skills such as collaborative problem solving and deep conceptual understanding of STEAM+C topics; and (3) *learning data sharing and access* which is a core problem that must be addressed in order to accomplish our LDI mission as to enable data science, there must be data. We end the paper by describing the LDI AISs autonomy levels or models of AISs-human partnerships and in particular AISs-teacher partnerships.

While major efforts are currently spent on standards and standardization for good reasons, we believe access to data at scale is a more critical, upstream challenge that needs to be addressed first as before mapping learning data to a particular standard for sharing, one must have have access to the data and have permission to share it. Data owners (parents, learners, schools, developers, and other owners) may fall into a wide spectrum of data sharing philosophies ranging from skeptical and paranoid (i.e., not willing to offer any access to data but rather work on their own device with a local copy of an AISs and not allow any data to be sent outside their own machine) to widely open, sharing data publicly for the benefit of everyone as such data can be used to improve AISs and learning environments that include AISs. Our goal of modelling learners and understanding their learning process and instructional environments via big edu-data is something beyond the traditional data-science regime. In particular, students, parents, and teachers are outside of our computing infrastructure while serving as a key to our end goal. In particular, access to that big-edu data, even when collected at the level of granularity, scale, and richness that we described above, is not to be presumed. Individuals, education technology developers, and school systems may be reluctant to contribute data to data analyses if they perceive the contribution as having only downside, perhaps running afoul of regulation, enabling sanctions, or jeopardizing individual opportunities. The security and privacy risks discourage data owners from sharing the data due to concerns of data abuse and data leaks, and at the same time, poses huge responsibility to data collectors to protect them under various regulations like FERPA.

SCALING UP AISS HORIZONTALLY – ACROSS TOPICS, DOMAINS

A major opportunity from a learning engineering perspective is the automation of the development processes of AISs, that is, making progress towards automating the authoring of AISs which should enable scalability across topics and domains which currently is a major stumbling block for a wider adoption of such systems. Expert-driven approaches to developing domain models, learner models, and instructional strategies for new topics and domains are expensive, tedious, and time-consuming. Automated or semi-automated approaches to discovering domains models, inferring learner models, and discovery instructional strategies are much needed. For instance, we intend to use neuro-symbolic approaches to automatically extract from both structured, e.g., student performance data, and semistructured data, i.e., text in textbooks, domain models.

Online data sources can greatly help in understanding and refining knowledge components (KCs) and KC relations, i.e., KC structure. For example, consider a statement from Wikipedia such as "using factorization we can solve quadratic equations." From the text in the statement, one can easily infer that factorization is related to understanding how to solve quadratic equations. This can be especially useful for discovering new concepts as well as refining existing concepts encoded in systems such as MATHia AIS. To do this, we aim to exploit large-scale, openly available diverse data sources such as online textbooks (CK-12 foundation) and Wikipedia to extract KC structure directly from natural language (free text). While there has been recent work that has focused on learning prerequisites and outcomes from textbooks (Labutov et al., 2017), our proposed approach is likely to be more general. Specifically, existing methods primarily use the index in textbooks and not the actual text content to determine prerequisite structure. However, the text description of concepts is likely to hold much deeper information about the KC structure. Therefore, in our proposed work, we plan to develop an information extraction system that converts unstructured text data into structured content containing KCs as entities and relationship between KCs as links or relations. We plan to develop our system based on Markov Logic Networks (MLNs), using an approach with which we have previously obtained state-of-the-art results in other information extraction tasks such as Biomedical event extraction (Venugopal et al., 2014). MLNs are particularly appealing for our task as compared to traditional information extraction systems that typically have pipeline architectures, namely, the entities are first extracted, and then conditioned on the extracted entities, the relations are extracted, which can lead to error propagation. In contrast, using MLNs will allow us to define a richer information extraction system where the entities and relations are extracted jointly. Further, we will integrate rich linguistic, lexical and semantic features to improve the proposed extraction system relying on various natural language processing tools such as SEMILAR, a semantic similarity toolkit (Rus et al., 2013a). One of the challenges here though is that some of these features cannot easily be encoded into MLN formulas or even if they can be encoded, their dimensionality is very large resulting in an extremely large probability distribution. For example, encoding dependency parse trees or semantic similarity features as logical formulas is difficult. At the same time linguistic features are well-known to be essential to achieve state-of-the-art performance in information extraction. We aim to encode such high-dimensional features as soft evidence where the weighting function for the evidence is derived from non-relational methods such as deep neural networks and support vector machines. We will integrate the extracted KC structure with the KC structure generated using other methods such as Learning Factors Analysis (LFA; Cen, Koedinger, & Junker 2006), where the extracted KC structure will act as hints which will i) increase our confidence in the KC structure found by the other methods and/or ii) allow us to further refine KCs based on the KC structure extracted from the text data.

SCALING UP AISs VERTICALLY TARGETING HIGH LEVEL, ADVANCED SKILLS

This LDI thrust focuses on using massive content, student performance data, and other learning data to *scale up AISs vertically targeting advanced skills such as collaborative problem solving and deep conceptual understanding of STEAM+C topics*.

Many AISs, and in particular largely deployed (at scale deployment over hundreds of thousands or even millions of learners), commercial AISs do not target more advanced topics such as collaborative problem solving or deep conceptual understanding. Collaborative work and collaborative problem solving skills are much needed in the 21st century (Autor, Levy, & Murnane, 2003; Carnevale & Smith, 2013) and learning activities fostering the acquisition of such skills must be adopted by learning ecologies of the future in order to make such ecologies more effective and equitable for all learners and more relevant to emerging needs and new realities. Our goal is to scale up AISs vertically, to offer training opportunities for such more advanced skills. The strategy is to extend AISs such as those offered by Carnegie Learning and Age of Learning with language through discourse components.

Language and discourse play a central role in learning (Morrison & Miller, 2018; Vygotsky, 1978), particularly for the acquisition of difficult topics that require deep comprehension, reasoning, problem solving, and collaboration that are required for higher paying jobs in the 21st century (Autor, Levy, & Murnane, 2003; Carnevale & Smith, 2013). Language and discourse are essential for developing argumentation skills (Ferretti & de la Paz, 2011), disciplinary literacy (Goldman et al., 2016; Shanahan & Shanahan, 2008; Shaffer, 2017), reasoning associated with mental models (Graesser, 2020; Rus, Olney, & Graesser, submitted), and formulating explanations of complex systems in science (Chi et al., 1989; Graesser, 2015) and computer code (Lasang & Rus, 2021).

Language and discourse is not only essential for learning within individuals but also learning in group contexts. Problems have dramatically increased in complexity, requiring collaborative problem solving by people with disparate expertise and perspectives (Carnivale & Smith, 2013; Graesser et al., 2018; OECD, 2017). Groups must establish a common ground of goals and ideas, negotiate plans, strategies, actions, and actor roles (Fiore et al., 2010), and periodically update team members to accommodate increasingly dynamic modern problems (Greiff et al., 2017). All of this requires language and discourse to accomplish.

Conversational agents have a long history of serving as proxies for human teachers, tutors, peers, and other roles in learning environments (Graesser, 2016; Johnson & Lester, 2016; Nye, Hu, & Graesser, 2014; McNamara et al., 2006). This approach provides several advantages. First, rich interaction data can be stored in a learner (and group) record store for coding and analysis. Second, the agents can reliably and precisely implement protocols of interactions, beyond the consistency possible with humans. Third, agents can be tailored to distinct populations of learners, accommodating performative, linguistic, and cultural differences, expanding equity and inclusion. Fourth, these advantages persist at scale, with agents able to run multiple instances simultaneously, day or night.

Effective and affordable collaborative learning environments with discourse components are at a tipping point. Dramatic improvements in several constituent technologies, including and especially data science, have created fertile ground for transformational convergence. LDI focuses on the following general goals: (1) leverage data analytic approaches to scale up prior work on learning and instructional processes in diverse learner populations, (2) develop (semi-) automated methods for conversational AI systems to generate dramatically more data, and (3) explore the human-technology frontier to optimize learning through discourse platforms.

LEARNING DATA CONVERGENCE, SHARING, AND ACCESS

To enable data science, there must be data and in particular "big" education data (big edu-data; see later). To this end, and given the LDI mission to harness the data revolution to transform the learning ecosystem, a key long term goal of our LDI institute is learning data convergence, i.e., collecting and aligning (more) comprehensive data about the same learner across skills, disciplines, and modalities (cognitive, metacognitive emotional, motivational, behavioral, social) and across time/grades, as well as data about the learnig process. That is, by learning data we refer to both data about the learner and about the learning process.

While prior efforts such as LearnSphere/DataShop have made progress towards building data infrastructure and capacity in education contexts, slow data convergence is a critical issue that hinders realizing the full potential of data and data science to transform the learning ecosystem. For instance, the DataShop metric reports show that most of the data is composed of datasets in the standard datashop format, of which there are about 3500 (https://pslcdatashop.web.cmu.edu/ MetricsReport). While this might seem superficially largish, the average number of observations per student is less than 400 the large number of students, greater than 800,000, is spread across more than 3000 datasets, resulting in less than 260 students per dataset. By the same token, the recently released EduNet (Choi et al., 2020) contains data from 784,309 students preparing for the

Test of English for International Communication at an average of 400.20 interactions per student. They do not cover STEAM+C topics and are limited to student performance data. That is, there is an "impoverished datasets" challenge in education, i.e., lack of big edu-data – see next.

Big edu-data should include data about millions of learners that is fine-grain (e.g., step/substep level information or detailed process data), rich (cognition, affect, motivation, behavior, social, epistemic), and longitudinal (across many grades). That is, big edu-data should be deep (e.g., about many learners), wide (e.g., capture as many learning relevant aspects as possible such as behavior, cognitive, social, emotional, epistemic, and motivational aspects), and long (being long-itudinal, across many grades or even lifetime). Convergence efforts will seek to "deepen" samples and "lengthen" timeframes of datasets that are (sometimes, but not always, already) "wide" in terms of features captured. Using these concepts, our goal can be re-stated as enabling the collection of deep, wide, and long education data which could then be analyzed using powerful Data Science methods capable of learning patterns from such massive collections of data and also accounting for input from diverse domain experts (science convergence) with the ultimate goal of transforming the learning ecosystem. That is, in order to fully harness the data revolution to transform the learning ecosystem we need: (1) improved, at scale data collection and (near) real-time access to big edu-data in ways that account for security, privacy, and ownsership, i.e., addressing the "impoverished datasets" challenge and (2) infrastructure to process learner data at scale using distributed computing (see the cloud continuum) and scalable algorithms and richer/more powerful algorithms (neuro-symbolic approaches).

LDI adopts the principle that the data owner (parent/guardian/teacher/school/developer/etc.) should be given a spectrum of options with respect to data sharing or, if deciding not to share, with respect to providing access to data. The spectrum of options should accomodate all attitudes that learners/learner data owners may have towards data ownership, security, and privacy. Indeed, access to learner data is a complex issue due to privacy, security, ownership, and regulatory concerns. Individuals such as learners (minors or adult learners) and education technology developers may be reluctant to share data if they perceive the contribution as having only downsides, perhaps running afoul of regulation, enabling sanctions, or jeopardizing individual opportunities. Security and privacy risks discourage data owners from sharing data due to concerns of data abuse and data leaks and pose huge responsibilities to data collectors to protect data under regulations like FERPA. A spectrum of solutions is needed to accommo- date various attitudes that learners/learner data owners may have towards data ownership, security, and privacy. For instance, at one end of the spectrum there need to be solutions for learners that do not wish to share their data or even provide access to it in any secure and privacy-preserving way. In that case, standalone education technologies are needed that can be downloaded on the learner's device and which do not rely and do not send any information to an outside location. The disadvantage is that the learners will not be able to take advantage of updated tailored components of the education technology and therefore will not have access to the optimal learning experience and learning outcomes. Other learners may be willing to share data or just provide access to some aggregate characteristics of the data and in return benefit from a better learning experience and better learning outcomes. In this case, learners do not provide access to the actual data but rather can offer the education developers and other interested parties access to certain characteristics or patterns of the data based on which the consumer of the data can generate new, synthetic data that matches the characteristics of the learner data. At the other end of the spectrum, there are learners willing to share their data with the education developer and also with other learner data consumer for the benefit of everyone as long as there are checks and assurances that the data will not be misused or used for purposes other than to improve learning experiences and outcomes for everyone.

We are aware that full data convergence would be hard to achieve for various reasons. However, our goal is to push the limits of what is possible, understand those limits, and act accordingly. Understanding the limits of data convergence will allow us to understand the limits of technology, what teachers can do to compensate for those limitations, and how to best orchestrate the learner-teacher-AISs partnership.

Our data convergence activity focuses on concrete examples from math and computer science (STEAM+C) as well as literacy (reading, writing) and leverage prior efforts in the area of building data infrastructure and capacity, contributing and expanding on those previous efforts to move us closer to the goal of full data convergence. Specifically, and as we already hinted earlier, one of our major goals is to build a fine-grain, large, and diverse (deep, wide, long) dataset that will enable LDI to explore the potential of data science methods to capture learners. By larger, we mean covering hundreds of thousands (possibly millions or more) of learners. By diverse, we mean learner data covering increasingly comprehensive facets or factors related to learning. That is, we call for the development of LearnerNet (Rus et al., 2020), an "ImageNet" (Su, Deng, & Fei-Fei, 2012) for learner modeling which could enable a transformation of our modelling and understanding of how learners learn, of adaptive instructional systems (AISs) that adapt to the learners and which rely on such learner models, and of the learning ecosystem as a whole. We announced and started the process of building LearnerNet in Fall 2019 as part of LDI Phase 1 (Rus, 2019 – ADL Directors' meeting talk). For instance, we have access to fine-grain student performance data from two largely deployed, commercial AISs covering math from early childhood to high-school (pre-K-12). We plan to "widen" this data by working with 3 school districts to collect data from students using the two AISs in order to cover other aspects of learning such as motivational, behavioral, epistemic, and affect. Furthermore, we will expand the two AISs with collaborative problem solving activities in order to understand learner's level of mastery of such skills and, based on their level of mastery and other characteristics, implement, evaluate, and refine instructional strategies meant to help students acquire collaborative problem solving skills.

HUMAN TECHNOLOGY FRONTIER: PUSHING FOR WIDER ADOPTION AND INTEGRATION OF AISs

Another major goal of the LDI is to foster wider adoption and integration with school-based and teacherled learning activities, i.e., making contributions to the Human-Technology Frontier which is one of the ten new Big Ideas for Future Investment announced by NSF. It should be noted that LDI makes contributions to three of the ten NSF new Big Ideas for Future Investment: Harnessing the Data Revolution for 21st Century Science and Engineering (HDR), Growing Science Convergence (GSC), and The Future of Work at the Human-Technology Frontier (FW-HTF).

Indeed, many teachers are overwhelmed by the many duties and tasks they have to handle as suggested by reduced teacher job satisfaction, burnout, and retention rates (Borg & Riding, 1991; Grayson & Alvarez, 2007; Hastings & Bham, 2003; Rhodes, Nevill, and Allan, 2004; Landers, Servilio, Alter, & Hayden, 2011). To this end, LDI is seeking answers to the following research questions: *Are learners, teachers, and learning science researchers successfully interacting with the cyber-learning technologies? How can data about these interactions be explored to guide understanding and improve these interactions through data collection and analysis to generate insights that will lead to improvements of human-technology partnerships? What kind of training does the future education workforce need to make the best use of cyber-learning for the benefit of students?*

AISs Autonomy Levels or Teacher-AISs Partnership Models

Finding the best teacher/learner-AISs partnerships could have transformative impact on the learning ecosystem such as freeing teachers from certain duties that AISs can do in an autonomous manner thus allowing teachers to focus on higher level tasks such as designing new instructional materials or new tailored interventions and tasks that AISs cannot handle. This better distribution of duties and coordination between teachers and AISs should lead to a more effective, efficient, engaging, and equitable learning ecosystem.

We defined and intend to study four levels of AISs autonomy with respect to how teachers may use such AISs: (1) fully autonomous – teachers need no training and not much involvement in tuning the AISs, (2) minimal teacher involvement – teachers tune the parameters of the AISs with the help of the AISs developer at the beginning of the school year or semester (minimal teacher training with respect to the workings of the AISs), (3) average teacher involvement – teachers require training and they work with the system on weekly basis selecting instructional tasks and receiving information from the AISs, (4) teacherdriven – the teachers exerts full control of the AISs including overriding decisions the AISs may take or suggest, the teacher will interact almost daily with the AISs. There is in fact one other level (level 0) which are self-improving fully autonomous AISs – they improve with experience with minimal or no developer intervention. While we will explore as resources permit the role of data science to enable such level 0, self-improving fully autonomous AISs, from a teacher and learner perspective they are similar to the fully autonomous level of AISs (level 1).

We plan to study and understand the trade-offs in terms of teacher involvement in tuning AISs vs. levels of AIS autonomy. For instance, teachers may choose a fully autonomous mode of operation for an AIS meant for students working independently with the system afterschool as supplemental instruction whereas for student interactions with the AIS during a class period, i.e., in a blended-learning environment, the same teacher may choose to control more the behavior of the AISs. Similarly, teachers may decide to use/download a pre-train learner model and update it with data from her students thus assuring data security and privacy and maintaining full ownership of the data. They may decide to share a sample of her own student data to benefit the pooled/pre-trained models that everyone can download as default.

From Traditional To Autonomous AI-driven Learning Environments

This section aims to clarify and emphasize our vision of the future of education as a learning ecosystem encompassing various forms of instruction (tutoring, classroom-based, remote schooling, etc.) and learning environments (instructor driven, technology as the primary driver of instruction, etc.).

As already noted, while our main focus is on learning with technology in fully online (i.e., outside of traditional classroom) and blended environments (i.e., education technology embedded in classroom instruction and learning), our longer term our goal is to support all forms of instructions as we envision a future where all instructional forms and learning environments are available. For instance, we imagine a traditional learning environment where technology is simply in the background, capturing the environment through data, e.g., the instructor – learner(s) interaction, in order to enable researchers and other stakeholders understand what works and what doesn't work in traditional, e.g., classroom, instruction. The analysis of the data capturing the learning environment will in turn provide input to instructors with the goal to improve the learner and instructor experience and learning outcomes and offer insights to other stakeholders such as parents and policy makers. In other words, the technology is in the background assisting the instructors, not driving the instruction.

At the other extreme of the spectrum of learning environments, we can imagine autonomous AI-driven learning environments where the technology drives the interaction with the learner(s) with humans (instructors, researchers, policymakers, etc.) playing more of a behind-the-scene force focusing on the development, analysis, and refinement of such learning technologies and environments. Anything in between those two extremes – traditional versus autonomous AI-driven learning technology – is possible implying various distributions of roles for human and computer-based instruction. For instance, we can imagine an environment where the AI-technology drives the so-called outer loop, i.e., the selection of instructional tasks, whereas the human instructor handles the inner loop, i.e., the within-task monitoring and feedback which currently is harder to do with technology. Furthermore, for instance, the teacher/instructor can focus on social aspects of learning and classroom interactions.

Indeed, it is part of our mission to contribute to understanding what instructional format works for whom and under what circumstances, improve learning and instructional processes and environments, and disseminate the findings and make recommendations.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

The LDI mission has been built upon and is meant to contribute to GIFT. The three major LDI thrusts we briefly presented in this paper can be viewed as recommendations for future GIFT efforts to pay attention to in order to enable the development of *AISs horizontally* – across topics and domains, *vertically* – targeting higher-level, advanced skills such as collaborative problem solving and deep conceptual understanding of STEAM+C topic, and provide learning data owners with options that match their attitudes and concerns regarding access to their learning data. Furthemore, the 5 LDI levels of AISs autonomy could inform future GIFT releases with respect to what type of AISs to support when it comes to their use in partnerships with teachers.

ACKNOWLEDGMENTS

The Learner Data Institute is sponsored by the National Science Foundation (NSF; award #1934745). The opinions, findings, and results are solely the authors' and do not reflect those of NSF.

REFERENCES

Sottilare, R., Brawner, K., Goldberg, B. & Holden, H. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). US Army Research Laboratory.

ABOUT THE AUTHORS

Dr. Vasile Rus *is a Professor of Computer Science with a joint appointment in the Institute for Intelligent Systems (ITS) whose areas of expertise are computational linguistics, artificial intelligence, software engineering, and computer science in general. His research areas of interest include question answering and asking, dialogue-based intelligent tutoring systems (ITSs), knowledge representation and reasoning, information retrieval, and machine learning. For the past 15 years, Dr. Rus has been heavily involved in various dialogue-based ITS projects including systems that tutor students on science topics (DeepTutor), reading strategies (iSTART), writing strategies (W-Pal), and metacognitive skills (MetaTutor). Currently, Dr. Rus leads the Learner Data Institute project as well as several other federally funded projects such as one that develops an intelligent tutoring systems for code comprehension. He has coedited two books, received several Best Paper Awards, and authored more than 100 publications in top, peerreviewed international conferences and journals. He is currently Associate Editor of the International Journal on Artificial Intelligence Tools.*

GIFT Giving and Receiving Helping Vendors Share Appropriately

Tyler Clarke[1], Austin Duncan[1], Dr. Gregory Goodwin[2], John Merrihew[1], Stacey Sokoloff[1]
Veloxiti, Inc.[1], Army Research Laboratory Human Research and Engineering Directorate[2]

Introduction

Soldier training requires a complex give and take between people and training technologies. Soldiers must train individually and collectively, both on home station and during deployment, which necessitates training at different levels of fidelity. Live, virtual, and constructive training, each of which provide unique opportunities and challenges, are often combined to provide soldiers with the most realistic training that is practical given the constraints under which a unit operates.

Given the importance of soldier training and its inherent complexity, it is not surprising that a wide variety of training technologies have been developed that range from general frameworks to specific niche focused tools. The Generalized Intelligent Framework for Tutoring (GIFT) provides soldiers with adaptive intelligent instruction, enabling trainers to develop training courses that leverage dissimilar tools and training technologies. The tools and training technologies leveraged by GIFT have been developed by different organizations, at different times, and are at different levels of fidelity. Enabling very different systems to interact is challenging, but it is a challenge worth addressing.

One facet of this challenge is enabling training technology developers to leverage complex assessment logic implemented by different organizations. The military operational environment changes quickly, necessitating updates in soldier training. Enabling training technology developers to leverage other organizations' capabilities is likely to help the training tools development community to quickly produce new training technologies and update existing technologies. This type of sharing is easier said than done for technological, administrative, and human reasons.

Ideally, instructors creating courses should not have to think about the underlying training technologies. They should be able to focus on course content and let the course creation framework recommend and/or manage the technologies that enable the instructor's vision. An ambitious, but less aspirational step on the way to achieving this goal is for training technologies to use a common language. In addition, they should be sufficiently transparent regarding assessment and instructional logic that a course creator, with the help of a framework such as GIFT, can combine existing training technologies without undue effort.

GIFT is in a unique position to help address these challenges. As a general training framework, GIFT serves as the interaction point between different training technologies. It sits between course creators and training tools and can help ensure that the appropriate technologies are applied to each task trained, but GIFT cannot do so without support. It is dependent upon course creators stating their needs and training technology developers communicating sufficient information for course creators to combine distinct technologies into a cohesive training course.

SIMULATION MONITORING AND REPORTING TOOL (SMART)

Overview

Veloxiti has insight into these challenges from the training technology developer point of view. Veloxiti is currently developing Squad SMART (Simulation Monitoring and Reporting Tool). SMART is an emerging architecture that processes data about the environment, trainee actions, and communication between team members to perform automated performance assessment. The SMART architecture prioritizes modularization of domain specific performance assessment logic.

The first version of SMART, FO (Forward Observer) SMART, provided automated individual task performance assessment for adjust fire missions. It was integrated with Bohemia Interactive's Virtual Battle Space 3 (VBS3) simulation version 3.9.2. The latest version of SMART, Squad SMART, provides collective task training for Army Battle Drill 5a, Attack a Bunker, and is integrated with VBS3 version 19.1. The Attack a Bunker battle drill includes a call for indirect fire support, presenting an opportunity to embed FO SMART functionality within Squad SMART, providing significant functionality without additional development.

Squad SMART was designed to support the axiom train as you fight. It uses the Android Team Awareness Kit (ATAK) for communication between a Squad Leader (SL) and a FO. ATAK is a Government Off the Shelf (GOTS) Android application that promotes situation awareness, and, among many other features, enables indirect fire requests. Since the verbal communication between a FO and a Fire Direction Center (FDC) is an essential part of an adjust fire mission, SMART uses a Commercial Off The Shelf (COTS) Speech To Text (STT) tool and contains a simple Natural Language Processing (NLP) capability to simulate and monitor verbal communication.

SMART GIFT Integration

SMART was not initially designed to integrate with GIFT but performing a limited integration posed minimal challenges since SMART and GIFT have many underlying concepts in common. SMART assesses tasks as Go or No-Go, which is like GIFT's Above Expectations, At Expectations, and Below Expectations categorization. Like GIFT, SMART tasks have starting and stopping conditions. SMART and GIFT both emphasize assessment transparency. The reasoning behind each assessment is provided by SMART and can be displayed in GIFT.

The current SMART / GIFT integration is considered limited because SMART can obtain data from GIFT and send assessments and reasoning to GIFT, but the integration currently does not provide significant added value to either component. A major near-term focus for SMART is to leverage GIFT's After-Action Review (AAR) capability.

SMART Architecture Components

SMART (Figure 1) contains a Service Oriented Architecture (SOA) of components that work together to provide collective task performance assessment. Services vary in complexity and responsibility from a simple FDC simulator to a complex performance assessment engine that leverages the Velox Toolkit.

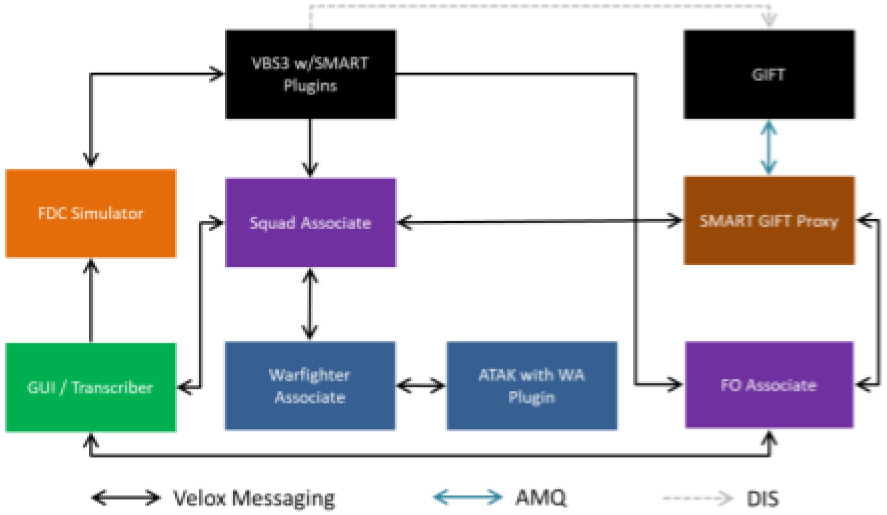


Figure 1: SMART Architecture

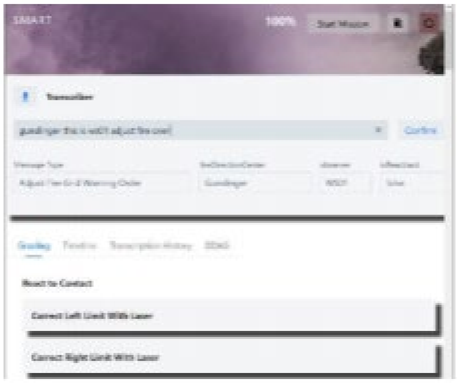
FDC Simulator: The simple SMART FDC Simulator monitors communication from the FO, verbally provides doctrinal FDC responses, and when appropriate, sends fire requests to a SMART VBS3 plugin that initiates explosions in VBS3.

FO Associate: The FO Associate assesses performance for fire mission related tasks. It contains executable domain logic that tracks a fire mission based on the interaction between the FO and the FDC and data from the environment (explosion locations, etc.). The FO Associate can assess grid, polar, or shift from known point fire missions.

GUI / Transcriber: The SMART GUI (Figure 2) serves two purposes. It provides a real-time view of task grading and it contains the speech to text transcriber that simulates radio communication between the FO and the FDC. The GUI can be configured to only contain the transcriber panel (for use by an FO), only show real-time grading (for use by an Observer Controller / Trainer), or both (as shown in the image). The real-time task grading view provides assessment explanations to enhance grading transparency. The transcriber functionality relies on a COTS STT toolkit. It has been tested with Dragon Naturally Speaking and built-in smart phone STT technology. The SMART transcriber functionality corrects the STT transcription using configurable common misunderstandings (ex. in the context of a fire mission, if STT heard “grade” it probably should have heard “grid”) automatically. It presents the user with its interpretation of the corrected transmission and enables the user to provide manual corrections.

SMART GIFT Proxy: The SMART GIFT Proxy sits between GIFT and the rest of the system. It acts as an adapter, enabling the rest of SMART to communicate using Velox Messaging (Velox Toolkit) but enabling the SMART classes inside of GIFT to communicate using out of the box ActiveMQ. The SMART GIFT Proxy also contains logic to control the amount of DIS traffic that is sent to the rest of the SMART system to prevent unnecessary processing.

SMART VBS3 Plugins: Squad SMART uses DIS when possible to obtain data from VBS3, but certain types of data that are vital to Army Battle Drill 5a are not available through DIS, including data about laser pointer and chem light use. This data is obtained from VBS3 using VBS3’s Action Script Interface (ASI) and published from SMART VBS3 plugins to interested Squad SMART components, including the SMART GIFT Proxy. A specific SMART VBS3 plugin exists to listen for fire requests from the FDC simulator and uses the ASI to trigger explosions in VBS3.



guardinger this is wd01 adjust fire over

Message Type: Adjust Fire Grid Warning Order

fireDirectionCenter: Guardinger

React to Contact

Correct Left Limit With Laser

Correct Right Limit With Laser

Figure 2: SMART GUI

Squad Associate: The Squad Associate assesses Army Battle Drill 5a tasks other than the call for fire. It contains complex assessment logic that often involves timing and positioning.

Warfighter Associate: The Warfighter Associate (WA) is a GOTS tool to provide warfighters with enhanced situation understanding and decision aiding. A key WA feature is the ability to mine tactical XMPP based chat to identify critical combat events related to CCIR (Commanders Critical Information Requirements) and PIR (Priority Intelligence Requirements). The WA was built to work with numerous user interfaces. The current iteration supports ATAK (Figure 3) and RaptorX. Squad SMART leverages the WA's ability to interact with ATAK.

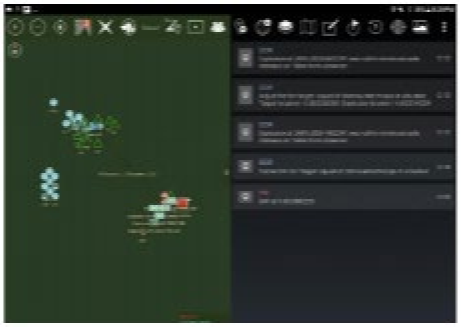


Figure 3: ATAK w/ WA Plugin

Squad SMART Scenario

The initial Squad SMART scenario provides collective training for Battle Drill 5a. Squad SMART expanded upon the adjust fire mission focused FO SMART to add Squad based assessment logic for the tactical behaviors that make up most of the battle drill. The Attack a Bunker battle drill begins when a unit comes under fire. The SL must react to contact, which involves repositioning troops, returning suppressive fire, and lasing the target. Then the SL requests fire using ATAK. The FO (if present) coordinates with the FDC. After the indirect fire, the SL leads the Squad through the covered bunker approach, coordinating with Team Leaders and ensuring that fires are shifted and lifted appropriately. Then the SL coordinates the bunker entry.

The critical roles for this scenario are the Squad Leader, the Forward Observer, and the Fire Direction Center (simulated). Small, but important, tasks are conducted by the Assault Team Leader and the Supporting Fire Team Leader. The scenario assumes that the SL requesting indirect fire support uses ATAK for communication, the FO has access to both ATAK and a radio, and the FDC only has radio access (Figure 4). An Observer Controller / Trainer can monitor the mission in real-time using the Squad SMART UI or GIFT.

The initial part of the scenario is reacting to contact, and the tasks Squad SMART grades include applying well aimed suppressive fire on the enemy and marking left and right limits of the bunker with an IR laser. In addition, SMART grades tasks pertaining to the Platoon Sergeant and Platoon Leader, including joining the squad in contact. All these tasks use configurable values that allow the course creator to modify the difficulty and realism of the course. Figure 3: ATAK w/ WA Plugin

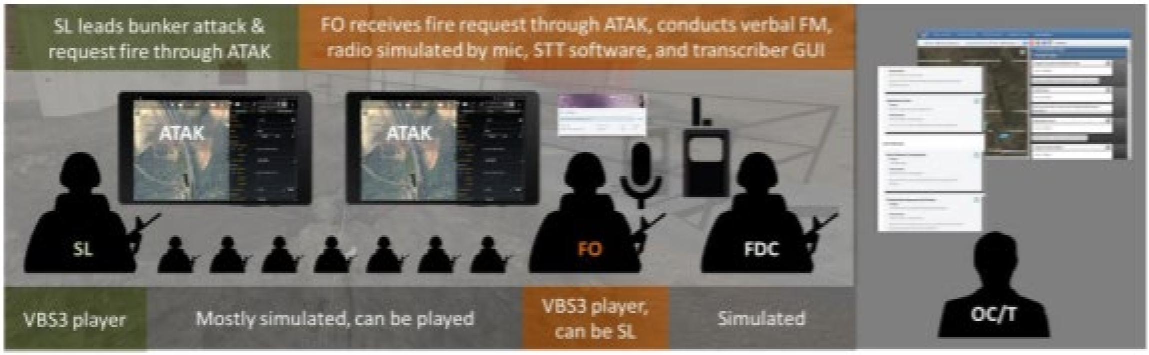

SL leads bunker attack & request fire through ATAK
FO receives fire request through ATAK, conducts verbal FM, radio simulated by mic, STT software, and transcriber GUI
ATAK
ATAK
SL
FO
FDC
OC/T
VBS3 player
Mostly simulated, can be played
VBS3 player, can be SL
Simulated

Figure 4: Squad SMART Scenario

Squad Leader Requests Indirect Fire Support

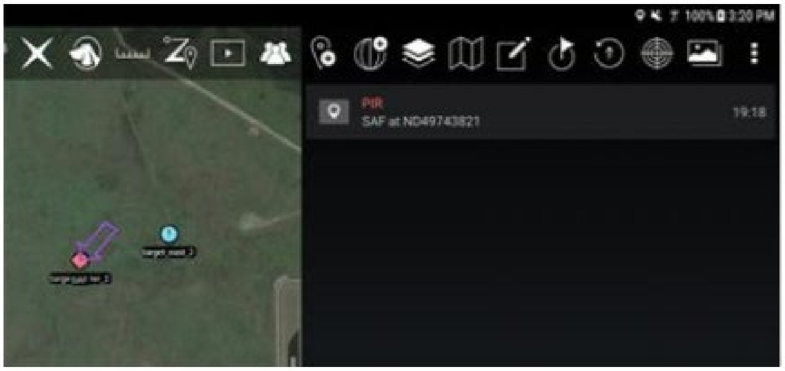
A Squad Leader is on patrol and comes under fire from a bunker. After reacting to contact and providing initial guidance to his unit, the Squad Leader uses ATAK to request indirect fire. To streamline the reporting process, Veloxiti leveraged its GOTS Warfighter Associate ATAK plugin.

Report Small Arms Fire

The Squad Leader reports Small Arms Fire (SAF) either through ATAK chat or by using the ATAK point dropper (Figure 5). The Warfighter Associate detects the new event of interest, sends a notification, adds a map icon, and adds a C2 pointer (purple arrow). Squad SMART uses this event to mark the time the event was reported. Future versions of Squad SMART may use the Warfighter Associate to deliver GIFT mediated intelligent training to a Squad Leader on the ATAK user interface.

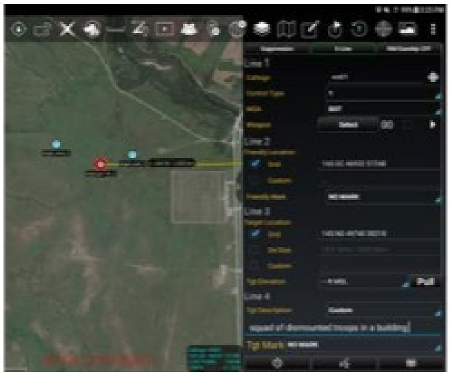
Submit Fire Request

The Squad Leader submits the call for fire using ATAK's standard indirect fire request mechanism, which involves tapping on the target and selecting the fire icon from a radial menu. ATAK presents the Squad Leader with a form with the appropriate fields pre-populated. The Squad Leader completes and submits the fire request. The request is sent through chat, observed using the Warfighter Associate chat monitoring service, and sent to Squad SMART, which parses the chat and assesses the Squad Leader's performance for accuracy and timeliness (Figure 6).



PIR
SAF at ND49743821
19:18

Figure 5: SAF Report



squad of dismounted troops in a building

Figure 6: ATAK Fire Request

Forward Observer Calls for Fire

The FO receives the chat message from the Squad Leader on his ATAK device and contacts the Fire Direction Center using a radio. In Squad SMART, radio transmissions are simulated using a speech to text toolkit (for example, Dragon Naturally Speaking Home Edition) and the SMART Web UI.

As the fire mission progresses, the simulated FDC service keeps track of the observer location, target location, and current fire request location, enabling the service to respond to an adjust fire request (ex. add 50, drop 100). When a transmission resulting in an explosion occurs, the simulated FDC calculates the explosion location based on fire mission transmissions and triggers an explosion in VBS3 using SMART VBS3 plugins.

Squad SMART observes the call for fire, including natural language transmissions from the FO to the simulated Fire Direction Center conducted via the Transcriber as well as entity location and explosion location data from the simulation.

Figure 7: Real Time Grading

The performance of the FO is assessed based on grading criteria provided by the Fires Center of Excellence at Fort Sill, Oklahoma. The real time grading user interface, shown in Figure 7, is optional.

The assessed performance for each mission can be saved as a PDF report (Figure 8) enabling trainees and trainers to maintain a record of training and to view and assess progress. The PDF report displays assessment explanations for each task as well as a timeline view that enables readers to view mission progress chronologically. The report can serve as a low fidelity means to support basic AARs between trainers and trainees.

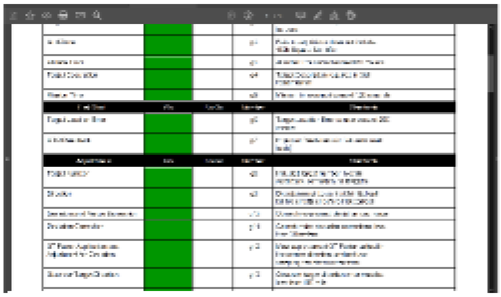


Figure 8: PDF Report

VELOX TOOLKIT

The Velox Framework was created to facilitate the development of systems based on Air Force Colonel John Boyd's Observe Orient Decide Act (OODA) Loop. While the OODA Loop was developed to support military aviation, the concepts apply to many domains. The OODA Loop is based on the principle that good performance requires monitoring the environment (observe), considering new observations in context with other information and prior knowledge (orient), making a decision based on the current situation assessment (decide), and acting on the decision (act).

The Velox Framework contains 3 main components to facilitate the development and verification of OODA Loop based systems: Velox Design, Velox Engine, and Velox Messaging.

Velox Design is a plugin for the Eclipse Java integrated development environment (IDE) that helps Java developers to implement OODA loop-based systems by enabling the creation of domain specific knowledge graphs through a drag-and-drop interface. The graphs processed by Velox are an Observe-Orient graph that contains beliefs about the environment and a Decide-Act graph that captures intent in the form of plans and goals. Velox Engine is a runtime component that processes the knowledge graphs created by Velox Design.

Velox Messaging is a tool for creating SOAs that generates publish and subscribe interfaces that are specified through XML contracts. Velox Messaging uses Google Protocol Buffers for serialization and language independence and currently supports Java and C++. It creates wrappers around underlying communications frameworks that allow for a common usage language without worrying about internal mechanics. The current version of Velox Messaging enables inter-process communication using either Apache ActiveMQ or PrismTech's Opensplice DDS for the underlying messaging framework.

ADDRESSING INTEROPRABILITY CHALLENGES

While there are many interoperability challenges, there were several critical challenges encountered during the integration of Squad SMART with GIFT.

1. **Intermediate Conclusion Sharing:** Training functionality is often based on domain specific calculations, many of which will be relevant to other systems. Sharing these calculations has the potential to reduce development time and increase consistency but may have the opposite effect if sharing occurs at the wrong level of granularity or for content too simple to justify the coordination burden.
2. **Combining Course Objects:** Course creators must understand the functionality and limitations of each training course object to successfully combine components developed by different vendors into a coherent course. Locating relevant course objects, ensuring that data needs are met, and enabling course creators to build a course that is a coherent whole rather than a conglomeration of parts are issues to address.

3. **Performance Assessment Categorization:** Different technologies use different schemas for assessing performance. To combine performance assessment technologies into a GIFT training course, the technologies must map their assessments to GIFT's categorization (at/above/below expectations) while not losing or assuming important data.
4. **High-Level Attributes and Competencies:** Tracking performance using competencies and other high-level attributes has the potential to greatly increase training quality but is difficult to achieve in a consistent and useful manner.

Course Creator User Types (Personas)

Before looking at these challenges, it is worth considering the characteristics of course creators. GIFT has the difficult challenge of supporting training course creators that vary greatly in domain knowledge, understanding of training course development theory, technical expertise, and available time. The course creator may be a Subject Matter Expert (SME), such as a TRADOC author, who is very knowledgeable about the theory and language of a domain. Another potential user is a vendor, who likely has less domain knowledge but more technical expertise. A third user type are soldiers developing training for their units. Soldiers likely have excellent practical experience but lack the time to focus on technical issues and often have little familiarity with the tools used in creating a course.

Ideally, the GIFT course creation experience would be usable by soldiers but enabling this user type to create training quickly and easily requires addressing the interoperability concerns discussed above.

Challenge 1: Intermediate Conclusion Sharing

Addressing this challenge, identifying which intermediate conclusions are appropriate to share with, and leverage from, other technologies, has the potential to reduce development time and to increase cross application consistency. It also has the potential to have the opposite effect if not carefully executed.

Many types of conclusions (synthesized data) are common across DoD domains. Aggregate unit location data, mission phase, battle damage assessments, etc. are fundamental to the military domain and likely needed to support performance assessment in a variety of specialties. Ideally, these conclusions would be shared from a common component and not re-implemented by different vendors for different training course objects. This can be accomplished numerous ways, including micro-services, data sharing, common libraries, and advanced GIFT interoperability plugins.

A micro-service is a small, independently deployable service that is responsible for calculations related to a domain capability, for example aggregating individual locations provided by a simulator into unit locations. GIFT can provide military domain focused micro-services that contain documentation specifying external data requirements and functionality. Vendors can list micro-service dependencies on training course objects. Using micro-services supports modularity, but complicates deployment, which may make micro-services less than ideal for GIFT, given the characteristics of its user population.

The concept of data sharing is like micro-services, but the calculations are produced by larger components with varied responsibilities. Data sharing is less modular than micro-services, giving it the potential to increase or decrease deployment complexity depending upon the characteristics of the service (ex. its dependencies) and how much of its data is needed.

Another potential solution is not to share data but to have GIFT provide libraries with domain specific calculations. This would enable vendors writing software in the supported language to reuse complex calculations, promoting consistency, and reducing development time. Whether providing domain specific calculation libraries is beneficial depends upon whether the benefit justifies the implementation and documentation effort. Even if members of the community contribute to the libraries, the GIFT team would still have to validate and maintain the code, placing an extra burden on busy developers.

Velox Messaging can help standardize data sharing across processes by providing a mechanism to specify XML based data contracts that can be referenced in service (publish and subscribe) specification contracts. Velox Messaging generates code based on these specifications promoting ontology based sharing.

The ideas mentioned so far are from the perspective of an external assessment engine developer. Perhaps the better solution is to perform common calculations in GIFT interoperability classes and share the results as produced messages. This simplifies the architecture and promotes GIFT usage but may make feeding conclusions into other conclusions challenging and/or obscure.

Regardless of the mechanism of sharing, it is important to identify which conclusions are appropriate to share. Although sharing granularly synthesized domain data may offer the greatest amount of reuse, it is possible that sharing only data produced by complex assessments is justified since the implementation difficulty is more likely to justify the coordination burden.

Much research has been conducted about creating a DoD domain ontology to support semantic interoperability. As an aspirational goal, having an ontology used by GIFT, through which vendors self-identify as producers and consumers of data, would go a long way towards addressing this challenge. Finding and maintaining an ontology that is specific and complete enough to have an impact is challenging.

Challenge 2: Combining Course Objects

Imagine if a Non-Commissioned Officer (NCO) assigned to an Infantry Squad could use the GIFT Course Creator to combine course objects into a coherent training course to instruct his unit on mission relevant skills without having to know anything about the underlying technology. In this world, the NCO would only be shown course objects relevant to Infantry Squads and able to run in the training environments that his team can access. After selecting a target training environment and specifying the topic of his course in natural language, the NCO would be presented with available relevant course objects that can be combined into a coherent training course with little understanding of the underlying technology. This vision has the potential to greatly increase training effectiveness but is difficult to achieve.

The first step in solving this challenge is to enable GIFT to recommend course objects. Many irrelevant course objects can be filtered out by enabling vendors to tag course objects with relevant echelons and specialties and comparing the tags with information associated with a user account. This would enable GIFT to, for example, only show an Infantry Squad NCO course objects related to an Infantry Squad. Additional filtering can take place based on the selected training environment. For example, it does not make sense to show course objects that require DIS data in a training environment that does not contain a DIS enabled simulator. Searching and tagging can further refine the training course objects presented based on a course creator's current goals. For example, a course creator can search for course objects related to checkpoints (topic) or radio usage (competency). This type of search is less structured and more complex than basic filtering based on echelon and specialty and may require the use of a third-party search technology, for example Apache Solr.

Locating relevant course objects is only one part of the challenge. Course creators must be able to combine course objects without having to worry about data availability. One solution is to require all course objects to operate on raw data provided by the training environment. This solution is possible, but perhaps not ideal because it requires coarse granularity. For example, in Squad SMART, a task to assess marking the blindside of a bunker with an IR flare requires the intermediate conclusion that the Squad approached the bunker. When running Squad SMART, this data is available. Theoretically, the IR flare task can be separated from the rest of Squad SMART if another component is able to assess the bunker approach but knowing that the constraint is satisfied and sharing the data between components is challenging. Perhaps GIFT can address this challenge using simulation interests or through an ontology.

Having verbose, user-oriented documentation and tagged information can provide a simple way for any user to navigate previously existing courses and determine what they provide. There have been attempts at standardizing data and software structure in a way that supports interoperability across many domains, in a generic manner. However, as time has proven, that is a daunting task. One approach is to look at domain specific interoperability. For example, even though SMART currently supports Battle Drill 5a, since Squad level battle drills use many of the same constituent tasks, SMART has the beginning of a Squad level performance assessment model. The simplest example of this is Battle Drill 2, which is wholly contained in Battle Drill 5a. By focusing the domain, interoperability becomes a tractable problem. In this example, having Battle Drill 2 as an independent course object with tagged metadata, a course creator could make use of it independently from the rest of Battle Drill 5a, assuming that the data availability issues are addressed.

Challenge 3: Performance Assessment Categorization

The SMART architecture assesses tasks into a Go / No-Go binary that leaves little room for nuance or metadata to capture truly helpful outcomes. The decision to use a simple Go / No-Go categorization was made to fit with the grading methodology used by the Fires Center of Excellence at Fort Sill. The GIFT categorization of Below Expectations, At Expectations, or Above Expectations is a little more expressive, but perhaps the categorization alone does not contain sufficient nuance to enable powerful performance assessment. GIFT allows a confidence rating to be attached to an assessment, which, when supported by the underlying assessor allows for a much richer expression of performance. To be most useful, the meaning of performance assessment categorizations must be clearly expressed. This includes documentation/metadata to communicate assessment logic to course creators and assessment explanations to communicate assessment logic to trainers and trainees. Standardizing these explanations is another interoperability challenge faced by the DoD training tools community.

Challenge 4: High-Level Attributes and Competencies

DoD training tasks are typically organized around concrete, measurable skills, for example, for a grid-based call for fire to be doctrinally correct, the target description must be the third message transmitted by the FO. Even though Squad SMART is assessing the student's ability to correctly transmit a target description as part of a grid call for fire, a trainee's performance on this task informs an assessment of his mastery of the radio usage competency. Performing a meta-assessment on all the task outcomes within a competency is likely to provide useful information. When the student performs with mixed success it can be easy to miss a pattern that could lead to helpful remediation by focusing only on concrete tasks and not on the related competencies. GIFT can address this challenge--of creating overarching competencies and presenting them to course creators and trainers or allowing them to create new competencies--by requiring that task definitions allow assignments to a higher-level competency. GIFT can serve as or integrate with a training record store to aggregate a list of competencies across courses, to help guide trainee course recommendations. When creating a course, GIFT could allow filtering on competencies or offer competency-related remediation strategies. This would enable GIFT to provide course creators the ability to better understand what competencies can be assessed by existing course objects.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

Rather than to emphatically propose or even strongly advocate for a specific solution to the challenge of improving interoperability for DoD training systems, this paper attempted to ask difficult questions and spur conversations on this challenging but important topic. These are difficult challenges, and the answers are currently unknown, but hopefully, by asking the right questions, GIFT can shape the discussion that improves training technology interoperability.

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S Army Combat Capabilities Development Command – Soldier Center – Simulation and Training Technology Center. The statements and opinions expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHOR

***Tyler Clarke** has been a Knowledge Engineer with Veloxiti, Inc. since 2013. He has worked to bring cutting-edge artificial intelligence technologies to bear on problems in both public and private sector domains ranging from civil affairs to finance, from intelligent training to cyber security response. Tyler earned his Bachelor of Science in Computer Science from Auburn University.*

***Austin Duncan** is a Software Engineer with Veloxiti, Inc. He is a Navy Information Technician veteran having extensive experience in Naval software, communications, and messaging systems. Since joining the company in 2018, he has contributed to numerous projects using his experience in Artificial Intelligence (AI), UIs, and software design to help warfighters train, maintain situational awareness, and perform their roles with increased expertise. Before joining Veloxiti, Austin earned a Bachelor or Science (BS) from the Georgia Institute of Technology with concentrations in Computational Theory and AI.*

***Dr. Gregory Goodwin** is a senior research scientist U. S. Army Combat Capability Development Command Soldier Center at the Simulation and Training Technology Center (CCDC-SC-STTC), in Orlando, Florida. For the last decade, he has worked for the Army researching ways to improve training methods and technologies. He has a PhD in Psychology from Binghamton University and an M.A. in Psychology from Wake Forest University.*

***John Merrihew** served as Principal Investigator on a dozen Artificial Intelligence projects in the US Army in the domains of data analytics, operations/intelligence fusion, UAV automatous operations and human terrain. John is a retired LTC Army Aviator who served 24 years in the US Army. He is a 4-time combat veteran of Operation Iraqi Freedom with 113 combat sorties and over 1300 combat hours over Iraq. He had command and staff positions at multiple levels in the US Army. He served in Infantry, Field Artillery, Aviation (Cavalry, Attack and Lift). John has extensive combat experience in intelligence fusion for warfighters at the tactical edge and is a qualified Joint Operational Planning.*

***Stacey Sokoloff** is a Senior Engineer with Veloxiti, Inc. Since joining the company in 2002, she has contributed to and provided technical leadership for numerous projects that attempt to advance the state of art by providing novel solutions to help warfighters train, maintain situational awareness, and perform their roles with increased expertise. Before joining Veloxiti, Stacey Sokoloff earned a Master of Human Computer Interaction (MHCI) from Carnegie Mellon University and a Bachelor of Science (BS) in Computer Science from Tulane University.*

The Renovated 中文 *Room*: Ethical Implications of Intentional AI in Learning Technology

Jeanine A. DeFalco [1], Shelly Blake-Plock [2] , & Andrew J. Hampton[3]
[1]U.S. Army CCDC–Soldier Center, [2] Yet Analytics, [3] University of Memphis

INTRODUCTION

In the (1980) essay, *Mind, brains, and programs*, John Searle maintains that an artificial intelligence (AI) program could not be realized to create a condition of understanding, perception, action, learning, and other intentional phenomena because only causal powers and the actual properties of synaptic sequences can instantiate the intentionality necessary for human-like understanding. Searle maintained that mere engineered instructions that manipulated formal symbols could not bring forth intentional understanding because symbol manipulations by themselves do not have intentionality, meaning, or a consciousness to make plans and achieve goals. Searle goes on to illustrate this argument with his *Chinese Room* thought experiment. Here he argues that inserting something that has intentionality (e.g., a person) into a system, but restricting their observable behavior by means of a formal program, essentially obviates that intentionality.

New developments in AI are beginning to suggest that intentionality could at some point be a feature—perhaps a core feature—of an artificially intelligent system. As intentionality implies an autonomous decision-making capability, it would stand to reason that such a system would on occasion be required to navigate ethical decisions—or decisions with ethical implications. If systems were designed with intentionality—meaning, a consciousness to make plans and achieve goals—how would we ensure its ethical nature, particularly if the AI system is intended to be autonomous?

The goal of this paper is not to suggest when intentionality will become a feature of AI. Rather, the authors have intended to provide a number of considerations with regard to ethics that may be beneficial to the design of AI-enabled systems in the present by considering the current state of personalized remediation practices. This includes the capability of leveraging reinforcement learning techniques in, for example, GIFT (the Generalized Intelligent Framework for Tutoring), as well as imagining what future ethical dilemma could look like if (or when) AI becomes capable of intention. By anticipating and designing for potential future ethical accidents or threats, one may improve upon the capabilities (and considerations that result in capabilities) of the systems feasible in the current technological paradigm. Therefore, the authors' position regarding this approach argues for a more deliberate articulation and standards-driven method of establishing an industry wide ethical framework and recommended practices that will inform the design and execution of AI driven systems where those systems may feature intentionality. In designing an ethical framework and processes for ethical risk assessments, the authors recommend designing for the eventuality of intentional AI. We feel this is particularly important as it relates specifically to AI-driven adaptive instructional systems (AISs).

INTENTIONALITY IN AI

Philosopher Robert Sparrow, in his paper *Why machines cannot be moral* (2021), insists that ethical reasoning will remain in the domain of human beings because ethics is inherently personal, subjective, and contextualized in a way that AI can never replicate; that the fundamental *personal* way in which we respond and reason about ethical dilemmas requires the *intention* to engage in ethical reasoning and behavior. Sparrow goes on to argue that this intention is grounded in individual subjectivity, and cumulative historical experiences that inform a person's response and responsibility in answering ethical dilemmas. This intentional and personal nature of ethical engagement cannot be realized in a reliable AI calculation, Sparrow argues, because fundamentally, AI systems have neither the capability to experience emotional remorse that shapes ethical thinking, nor the established moral authority acquired over a history of realized ethical behaviors necessary for solving ethical dilemmas. Further, Sparrow maintains, "moral machines" would be incapable of identifying and resolving ethical dilemmas even if trained on datasets of ethical texts and judgements… No scientific calculations could ever adequately simulate the necessary associative personal regret that hallmarks the incentive for ethical behavior. In short, Sparrow maintains that a moral machine cannot be realized because engineers and designers do not (and presumably could not) understand the nature of ethics—which is fundamentally and uniquely shaped by the subjective personal stance and intentions of an individual.

Where Sparrow's analysis fails, as it fails similarly with Searle, is in the presumption that AI systems have a pre-ordained limitation in their capability to replicate intentional decisions. However, Sparrow's analysis exceeds Searle's constrained analysis in his assertion that ethical responses are defined by their associated affective responses by which people are incentivized to engage in ethical behavior. The authors of this paper counter this position by arguing that affective responses do not in and of themselves ensure nor promote ethical behavior, and to tie the potentiality of ethical behavior to the affective realm is, from a cognitive psychology perspective, in error. Chen et al. (2019) note that while Bandura (1999) theorized that internalized social moral standards regulate behavior through shame, guilt, or remorse, this self-regulation process can in fact be bypassed through cognitive mechanisms, e.g., employing disengagement strategies to evade self-condemning reactions: "The use of moral disengagement strategies enables individuals to engage in unethical behavior without self-disapproval," (Chen et al., 2019). This is an important distinguishable element particularly when seeking to model ethical processes in AI systems: because while self-deception and affective disengagement are elements that may interfere in the execution of an ethical behavior in humans, AI systems can maintain a consistency of ethically constrained actions by design.

But perhaps more importantly, Sparrow's analysis is short sighted in that—if followed to its natural conclusion—the dismissal of the possibility of intentional ethical AI systems risks abandoning any effort toward designing moral or ethical machines. To the contrary, the aim should not be an all or nothing venture. Even if a perfectly designed ethical AI system is not something presently realized, the authors of this paper argue that the aim should be to consistently re-train and provide data to support the continued development of the self-improving, intentional capabilities of an AI system. Sparrow's dismissal, in fact, creates a fertile ground for allowing invisible ethical threats to become manifest, as AI systems will increasingly be used to execute tasks independent of a human in the loop, and these tasks may contain ethical risks that threaten human flourishing—ensuring human flourishing, we argue, as the first principle of ethical AI (Stahl et al., 2021).

In short, advances in the design, function, and breadth of capability within AI has changed its fundamental nature. This level of complexity argues against a simplistic dismissal of the potentiality of devising ethical machines, and dismissal of AI as mere processes or symbol manipulation as was the case for Searle's ignorant man in the Chinese Room, or constraining ethical behavior as being regulated by affective responses according to Sparrow. Whether or not this advancement constitutes "true" intentionality is essentially irrelevant and missing important assessments of ethical risk. The fact remains that AI systems often (and increasingly) function as semi-independent agents, beyond the immediate control or understanding of even their designers. Decision-making and other intelligently executed actions thus demand careful consideration in both design and evaluation. Simply stated, if we intend to let machines make their own decisions, we need to know they do so within an accepted ethical framework.

IMPLICATIONS FOR GIFT COMMUNITY

AISs have demonstrated utility as training and education systems, as seen most evidently in the GIFT community. GIFT and other AISs systems have been deployed in military, academic, and commercial versions. These systems AISs have the potential to significantly improve both quality and scale of learning across many sectors and to be the minders of ethical values within the experiential space of learning for both their human and machine interactants, especially as they employ the scale resources of commercial AI infrastructure and the opportunity of Linked Data and an ever evolving semantic data pool. As many AIS deployments now reside in or interact with the cloud, two market needs have been created: one for transparency concerning the operation, features, functionality, and use of AI in these systems; and one for the interoperable exchange of data with other learning and enterprise systems.

Focusing on the need for transparency, when we consider, for example, GIFT's ICAP pedagogical model that supports personalized remediation practices for individual learners, it has been designed to apply Markov Decision Processes and reinforcement learning techniques to establish remediation policies that determine what learning concepts and remediation content to deliver (Goldberg et al., 2020). This process that leads from presentation of information to gathering of evidence ultimately aimed at asserting competency is currently a tightly scoped and highly defined process. However, we would expect that subsequent generations of AI in GIFT will contribute to the automation of that process and that the process itself may become increasingly subject to the ongoing development and maturity of the decision-making capability of AI. It is reasonable to presume that GIFT will expand the scope of the automation of pedagogical decisions for learners that will have a direct downstream effect on business factors such as competency assertions.

Yet, the efficacy of ethical guidelines or codes as a basis for ethical decision-making for software engineers is effectively nonexistent (McNamara et al., 2018). Enforceability of aligning AIS design decisions to ethical frameworks will not occur by mere consensus across private and public sectors. Rather, policy makers must establish safeguards through legal measures and standards that incentivize compliance. Importantly, the consideration of ethics in AI needs to be reframed from a negative, restrictive mindset and rather as risk assessments that actually improve the scope of action, uncover blind spots, promote autonomy and freedom, and foster self-responsibility (Hagendorff, 2020). Within this context, we see the GIFT community as instrumental in contributing to the establishment and reinforcement of ethical risk assessment norms in their current and future design and implementation of GIFT and comparable AISs.

While trust in technology has been a longstanding area of concern with new and emerging technologies, it is important to highlight that one of the central functions of ethical thinking and reasoning is as a tool to bolster trust in a system that identifies blind spots and unanticipated threats as it relates to human flourishing. In complex component systems, risk assessments of engineering design is key to the engineering processes as it is an established principle that a system of significant scale will produce "normal accidents," (Williams & Yampolskiy, 2021). Similarly, in anticipation of a continued expansion and complexity of AI "components," risk assessments need to include an assessment beyond the mere mechanics of a system and include thorough analyses of ethical risks that threaten or even simply impede human flourishing. For example, unethical AI could include conducting unmonitored forms of AI experiments on society without informed consent, collateral damage form data breaches, biased and unfair algorithms, hiding harmful or flawed AI functionalities under the guise of trade secrets, vulnerabilities to cyberattacks, identity theft, disclosing personal data via machine learning applications, attacks on IT infrastructures, misinformation to perpetuate fraud or social engineering (Hagendorff, 2020).

More relevant to the AIS domain and the GIFT community of users, there are several types of ethical decision-making events that are specific to the learning and training domain which are likely to occur as AI matures in the educational domain. Not surprisingly, many of these events have corollaries in the ethical judgement process made by human instructors in the course of everyday work. They include:

- Identifying and subsequently dealing with cases of cheating
- Making adaptations to deadlines and schedules based on unforeseen or developing circumstances
- Allowing work to be turned in late for one of many reasons
- Deciding to issue or include a trigger warning with specific content
- Handling a student request to be excused from engaging with certain content
- Designing fair and equitable groupings of learners both homogeneous and heterogeneous
- Playing up a falsehood for the purpose of eliciting an instructional response

- Judging winners and losers of zero-sum and non zero-sum games, including in subjective events such as awards for artistic achievement
- Knowing when to alert authorities to a situation
- Deciding how to handle a parent or third-party request
- Choosing not to provide information upon determining a request from a bad actor
- Making decisions with life or death consequences in high stakes training environments

Additionally, there are decision-making events with ethical implications that fall outside of the normal course of human instructor experience. They may now or in the future include:

- Auto-scaling of one region versus another in the midst of a service disruption during high-stakes assessment delivered via distributed means
- Automated decision to share biometric or educational data with another application when those data are undefined per a service license or policy
- Privacy scope protocols when leveraging Linked Data across the internet
- Inherent bias amplified due to the nature of training data sets
- Dynamically evaluating decision criteria for A/B testing of features or algorithms that advantage one group over another.
- Pushing versus delaying updates based on the relative importance of optimization versus standardization of instruction.

Of more immediate concern in regards to GIFT is in the decisions that will drive competency assertions. In the case wherein decisions that GIFT makes in terms of what to provide to the learner, the experience provided holds the key to whether competency could be ascertained and therefore will have a direct outcome as to whether that learner should or can be asserted to have a competency. Currently, that decision is made in a virtual handshake between GIFT and an LRS upon the identification of target xAPI statements entering the data pool. In future iterations of GIFT, however, that decision making process has the potential to become increasingly automated--and when it is, eventually the automation is going to carry decision-making intentionality. For example, GIFT could decide to block a learner from accessing content or experiences necessary to demonstrate competency for a variety of decision-related reasons with ethical implications including risk management, prediction based on prior learning and history recorded in the learner's social or knowledge graph, and preference for promoting a learner with one behavioral profile over another regardless of the potential for training success. It is reasonable, then, for GIFT course authors to conduct ethical risk assessments when considering the effects and implications of regulating activity in such a way that a competency can or cannot be derived. And in the future it may mean that humans-in-the-loop will need to act as de facto referees of decisions made that have ethical leanings or moral consequences.

All of these and more dilemmas foreshadow the types of decisions which may fall into the hands (and artificial minds) of intentional AI systems in the future, including GIFT. In short, without anticipating the ethical threats that could occur from self-improving, intentional AI driven AISs, this oversight could cause loss and long term negative effects for individuals and society more broadly. Designing AISs that pass judgments and become the gatekeepers for growth and advancement of individuals without orienting that social power towards ethical principles such as honesty, justice, courage, empathy, care, civility, or magnanimity, could result in both short and long term societal detriments, including advancement for few and autonomy for none.

FUTURE CONSIDERATIONS

While contemporary AIs themselves are not capable of moral acts, that is not to say either that they will not be in the future nor that there may be acts carried out in the future that are not (by present definitions) considered moral acts. Hew (2014) points out that in a universe of ethical decisions composed of a rule set designed by humans, it stands to reason that any decision resultant in such a system be construed as an outcome of human design factors. But such a system is constrained by present knowledge of what such a system contains—namely human-centric designs and expectations. In the same breath that we say that a future machine may write its own code, we can catch a whisper of what such machine-centric design may mean for ethical understanding. And such a shift in authorship, ownership, and indeed *intention*, could have the effect of creating a parallel set of ethics—or agreed upon rule sets with moral implications—created not by humans for human purposes, but by machines for machine purposes. As a field, AI developers must ultimately decide whether or not such parallel sets of ethics can co-exist, and if not, how to align such rules under the broader first principle of ensuring human flourishing. But this requires deliberation and foresight in initial engineering design planning.

If we were to follow Searle's reasoning that intent is the result of a causal power willed or otherwise negotiated by an instantiating process driven by synaptic sequences, then could not something like a drone programmed to leverage a neural network for decision-making purposes, such as in the context of a killchain, be deemed to be acting with intent as it executes its task? And if such a drone were to erroneously kill a civilian based on that decision, would it be the fault of a programmer who at no point during the decision-making process was ever in the loop? And if the neural network is constantly being updated by the delivery and gathering of data and generating decisions based on that constant flow of data, can we say that the decision occurs in any finite state? These arguments concerning intention, responsibility, and finite state seem as though written for an era prior to the one in which we find ourselves.

And there is the matter of who contrives the ethical universe. If we can agree that at some point in the future machines will be able to write their own code (and thus make their own decisions as to what to value in that design), then in the same way that we can note that an ethical universe may be designed by a human, so too a universe could be created by a machine. The ethical value of any system of rules within this artificially comprised universe could be indistinguishable from any such as created by humans. Once this leap of faith is made, can we then say that the machine is responsible for the decisions that it has made in its own ethical universe? Rather than wallow the philosophical muck of an ethical Turing Test, and rather than attempt to pin down blame for activities that will occur at a rate of speed and scale beyond human capacity to negotiate, we might be better off considering the design of AIs which themselves can act in this future artificial universe of machine-derived ethical rules as ethical referees among other AIs. In this way, we answer the question: "Are AIs capable of moral acts?" by asking the question: "Are AIs capable of minding other AIs?"

THE TASK GOING FORWARD

One of the central tenets of this paper is the notion that when considering ethical implications for AISs and GIFT in particular, our aim should not rest on the capabilities of AI as it is now, but anticipating what could be. Technological innovations, even if designed purposefully for human flourishing, still contain disruptive potentials (Hagendorff, 2020) that challenge our preconceptions on the stability of agency between humans and technology (Fischer & Wenger, 2021). A common refrain is that ethics is a process and not a solution (Boe et al., 2013; DeFalco & Hampton, 2020; Hagendorff, 2020). Stahl et al. (2021) asserts that the attempt to establish a stable definition of AI or the related ethical issues is misguided, and we should rather understand that ethics is dynamic and based on process and change wherein the integration of new technologies in society requires ongoing negotiations of facts and values.

These ongoing negotiations require regulatory governance, the adoption of legal frameworks, independent auditing of technologies, an investment in education that integrates ethics and technology, and standardization initiatives (Hagendorff, 2020). Standardization initiatives such as the IEEE P7000 family of standards and IEEE's P2247.4's working group that is establishing recommended practices for ethical considerations of AISs, are actively working on creating documents of consensus that can provide guidance to product developers and consumers. In addition, establishing mechanisms for assessing ethical risks—e.g., data protection, ethical, social, and human rights impact—within private corporations, as well as mandating ethical risk assessments for publicly funded acquisitions, would further the path of reinforcing normative AI ethics in the absence of legislation (Stahl et al., 2021). Ethical risk assessments could be informed by guidance from reports such as the *Ethical Framework for AI in Education* (2021) published by the Institute for Ethical AI in

Education at the University of Buckingham. This report identifies specific ethical concerns that should be considered when acquiring an AI-enabled learning capability:

- demonstrating efficacy in helping a learner to achieve educational goals
- implementation of a broad range of forms of assessment
- increasing administrative and workload capacity while respecting human relationships
- insurances of equity in learning
- enhancing autonomy of learners
- enforcing privacy
- transparency and accountability where humans are ultimately responsible
- informed participation by all constituencies
- adherence to best practices in ethical designs

These ethical concerns are important in the AIS domain if simply because the fundamental principle in devising a learning system is oriented specifically for human flourishing, and to omit proactive analyses to anticipate even unanticipated harm for its target population would be nonsensical. By designing ethical frameworks with an eye towards the possible world(s) fostered by an intentional AI future, we may protect the population engaged with the AIs of contemporary learning without putting unnecessary limitations on our ability to carry the philosophical and practical conversation into whatever the future may hold. Whereas limiting the ethical conversation to the AI capabilities of today may have the undue consequence not only of ill preparing our moral conversation, the saddling the policy on which our ethical values may be implemented with the ethical equivalent of technical debt.

CONCLUSION

We anticipate AISs such as GIFT will continue to make inroads as a central path for education and training for civilians and military personnel, as well as expand the capabilities of AI driven decision-making capabilities. AI technologies will continue to shape the evolution of AISs, and ethical considerations must acknowledge the growing complexity of AI and its increasing decision-making autonomy. This increasing decision-making autonomy of AI concerns decisions that an AI can take on its own with little or no prior human approval, intervention, or supervision. Whether or not an intentionally ethical or moral machine can be realized is almost an irrelevant speculation. What is relevant is that AI driven systems, and in particular AISs, will engage in decision-making that affects human flourishing, and it is to that point that organizations should assess ethical threats and establish processes to anticipate the unexpected. As part of the IEEE effort to establish recommended practices for ethical considerations in AISs, we suggest it would be beneficial to both GIFT and to the AIS field at large if GIFT stakeholders are actively taking part and contributing to the development of that standard, as well as make concerted efforts to establish normative ethical risk assessment processes in future design and implementations of GIFT. Preparing for what could be is perhaps the most ethical decision we can make.

REFERENCES

Bøe, T. D., Kristoffersen, K., Lidbom, P. A., Lindvig, G. R., Seikkula, J., Ulland, D., & Zachariassen, K. (2013). Change is an ongoing ethical event: Levinas, Bakhtin and the dialogical dynamics of becoming. *Australian and New Zealand Journal of Family Therapy*, *34*(1), 18–31.

Brendel, A. B., Mirbabaie, M., Lembcke, T. B., & Hofeditz, L. (2021). Ethical management of artificial intelligence. *Sustainability, 13*(4), 1974.

DeFalco, J. & Hampton, A., J. (2020). Dewey's ethics of moral principles and deliberation: Extending IEEE's ethics initiative for adaptive *i*nstructional systems. *International Conference on Human-Computer Interaction*. Springer, Cham.

Fischer, S. C., & Wenger, A. (2021). Artificial intelligence, forward-looking governance and the future of security. *Swiss Political Science Review*, *27*(1), 170–179. Retrieved at https://onlinelibrary.wiley.com/doi/epdf/10.1111/spsr.12439

Goldberg, B., Brawner, K., & Hoffman, M. (2020). The GIFT Architecture and Features Update: 2020 Edition. *Proceedings of the 8th Annual GIFT Users Symposium (GIFTSym8).* Retrieved at https://gifttutoring.org/attachments/download/3708/giftsym8_proceedings.pdf

Hagendorff, T. (2020). The ethics of AI ethics: An evaluation of guidelines. *Minds and Machines, 30*(1), 99–120. Retrieved at https://link.springer.com/article/10.1007/s11023-020-09517-8

Hew, P. C. (2014). Artificial moral agents are infeasible with foreseeable technologies. *Ethics and Information Technology, 16*(3): 197–206. doi:10.1007/s10676-014-9345-6

Institute for Ethical AI in Education. (2021). *The Ethical Framework for AI in Education.* University of Buckingham. Retrieved at https://www.buckingham.ac.uk/wp-content/uploads/2021/03/The-Institute-for-Ethical-AI-in-Education-The-Ethical-Framework-for-AI-in-Education.pdf

Mittelstadt, M. (2019.) Principles alone cannot guarantee ethical AI. *Nature Machine Intelligence*. *1*:501–507.

Searle, J. (1980). Minds, brains, and programs. *Behavioral and Brain Sciences, 3*(3), 417–424. doi:10.1017/S0140525X00005756

Sparrow, R. 2021. Why machines cannot be moral. *AI & Society: Journal of Knowledge, Culture and Communication.* Published Online: 21 January, 2021. DOI: https://doi.org/10.1007/s00146-020-01132-6.

Stahl, B. C., Andreou, A., Brey, P., Hatzakis, T., Kirichenko, A., Macnish, K., ... & Wright, D. (2021). Artificial intelligence for human flourishing–Beyond principles for machine learning. *Journal of Business Research*, *124*, 374–388.

Williams, R. (2021). Understanding and Avoiding AI Failures: A Practical Guide. *arXiv preprint arXiv:2104.12582*.

Zhang, B., Anderljung, M., Kahn, L., Dreksler, N., Horowitz, M. C., & Dafoe, A. (2021). Ethics and governance of artificial intelligence: Evidence from a survey of machine learning researchers. arXiv preprint arXiv:2105.02117.

ACKNOWLEDGEMENTS

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.

ABOUT THE AUTHORS

Jeanine A. DeFalco, PhD, *is a Research Psychologist at the U.S. Army Combat Capabilities Development Command – Soldier Center, SFC Paul Ray Smith Simulation & Training Technology Center in Orlando, FL. She serves as chair for IEEE working group P2247.4, Recommended Practice for Ethically Aligned Design of Artificial Intelligence (AI) in Adaptive Instructional Systems.*

Shelly Blake-Plock, *is President and CEO at Yet Analytics, Inc. He is an officer of the IEEE Learning Technology Standards Committee.*

Andrew J. Hampton, PhD, *is a Research Scientist Assistant Professor at the Institute for Intelligent Systems & Department of Psychology, within the University of Memphis. He also serves as chair of the IEEE Standards Association working group P2247.1 for Adaptive Instructional Systems.*

GIFT in a Blended Learning and Competency Development Continuum

Benjamin Goldberg[1] and Kevin Owens[2]
US Army DEVCOM Soldier Center[1]. Applied Research Laboratories University of Texas-Austin[2]

Introduction

It is important to forecast the future learning environment to assist in the shaping and execution of research and development to achieve a forward vision. The Generalized Intelligent Framework for Tutoring (GIFT) has been designed to support a number of different learning and training contexts, but requires distinct capability road-maps to inform those future research necessities. In the spirit of generalizability, both from a domain and learning resource standpoint, GIFT utilizes customizable "course objects" to calibrate the learning experience for an individual or team of learners. Yet to inform the early instantiation of each GIFT course object, specific use cases were defined to build context and guide prototype implementation. These course objects were designed to support explicit research questions aligned with the application of Adaptive Instructional Systems (AIS) in the context of military training. Supported functions from these objects include: (1) adaptive distributed training with multimedia content, (2) live training with mobile-device adaptive supports, (3) psychomotor skill training with sensorized environments, (4) real-time tutoring and visualization with simulation- and game-based environments, and (5) After-Action Review capabilities. With a focus on extensibility and re-use, these course objects provide a suite of interoperable learning activities, that when tied together, support a learning continuum across resources, enabling a system-of-systems approach to competency and skill development.

In this paper, we present a new use case centered on the utility of GIFT within a training continuum that syncs together a suite of learning experiences to support competency development. In this manner, we provide a specific set of user stories centered on an experiential learning model (Owens, 2020; Owens, Gupton, Hellman & Goldberg, 2020). It details how GIFT's services can be used across a collective training ecosystem that leverages a modernized Synthetic Training Environment (STE). Informed by recent work under the STE Experiential Learning for Readiness (STEEL-R; Goldberg, Hellman, Gupton & Owens, 2020) research integrating GIFT with functional components of the Advanced Distributed Learning (ADL) Initiative's Total Learning Architecture (Walcutt & Schatz, 2019), the use case to follow provides a long-range vision that is dependent on data interoperability and competency-based modeling methodologies.

Use case framing

To frame the discussion, this use case follows the Plan, Prepare, Execute, and Assess (PPEA) phases of activity around an operational training event. In this context, we are specifically linking the PPEA process to training management activities centered on authoring and delivering a collective training-event, rather than the sequence of activities and behaviors a team executes when performing a mission. We present the processes, procedures, and technology roles across each phase, with an emphasis on establishing a training evolution that influences team development and improving task proficiencies. This use case story describes the experience of a single PPEA cycle, but the approach is extensible and iterative in nature. This is intended to spark conversation and continue future roadmap iterations to inform future research requirements and capability investments.

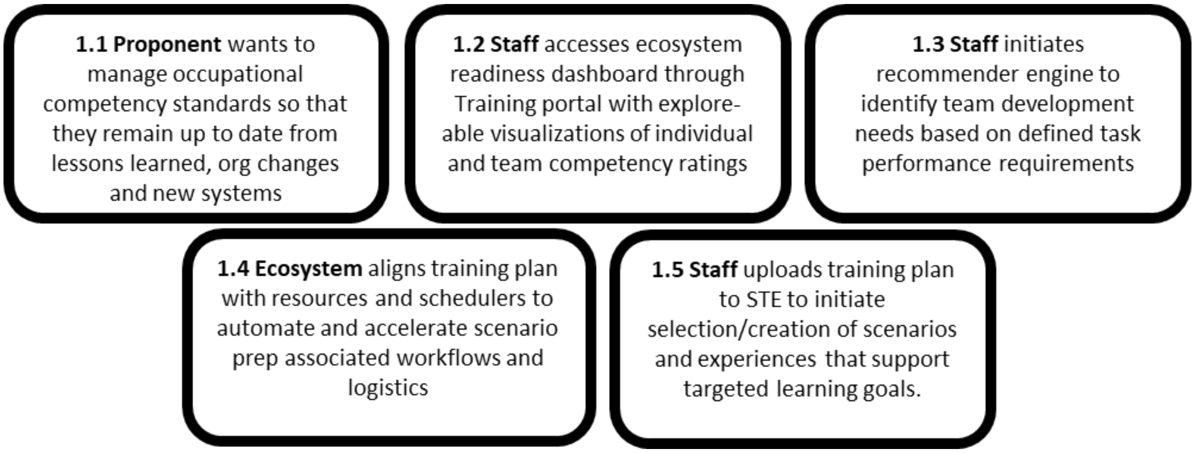
For this exercise, there are three primary components that are required: (1) an adaptive instructional system component that enables the capture and assessment of interaction data in real-time (e.g., GIFT); (2) a persistent data lake component that collects and tracks experiential data across a library of learning and training resources (e.g., Learner Record Stores); and (3) a competency management system (CMS) component that infers competency state based on collected data over time and how that data align to a set of managed competency frameworks. An underlying assumption is that there are existing standards and business rules to enable interoperability requirements to support a learning ecosystem paradigm described. This is achieved by leveraging the eXperience Application Programming Interface (XAPI; Barr, Fletcher & Morrison, 2020) and establishing XAPI profiles that produce granular experiential statements to model competency and proficiency from simulation events.

PPEA Use Case Breakdown

In this section we provide a detailed breakdown of the PPEA use case. We will ultimately describe what needs to exist, rather than what is currently feasible or available. This provides a means for examining the differences and associated deltas between what exists today versus what is needed tomorrow to drive modernization. This associates with both technological and organizational change and investment to meet the needs of future learners.

Plan

During the Plan phase (see Figure 1) of collective scenario development, team decision-makers are responsible for selecting Mission Essential Tasks (METs) and defining echelon relevant training objectives that will drive design and implementation. These METs and objectives are explicitly linked to established, and proponent managed competency frameworks that are continuously maintained based on operational requirements. These frameworks are represented within a Competency Management System that aligns experiential data across operational and training data sources to track unit and individual readiness. This requires standards and specifications for each experiential data source to assert a competency rating across the knowledge, skills and behaviors aligned across all echelon structures.



1.1 Proponent wants to manage occupational competency standards so that they remain up to date from lessons learned, org changes and new systems

1.2 Staff accesses ecosystem readiness dashboard through Training portal with explore-able visualizations of individual and team competency ratings

1.3 Staff initiates recommender engine to identify team development needs based on defined task performance requirements

1.4 Ecosystem aligns training plan with resources and schedulers to automate and accelerate scenario prep associated workflows and logistics

1.5 Staff uploads training plan to STE to initiate selection/creation of scenarios and experiences that support targeted learning goals.

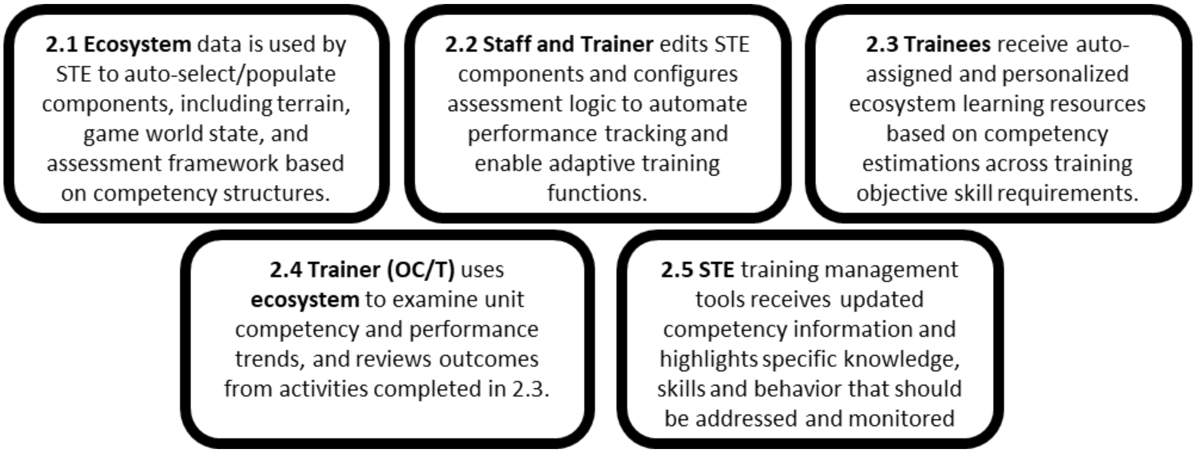
Figure 1. PLAN Activity Workflow

In executing the Plan workflows, unit staff accesses a learning ecosystem dashboard that automatically populates the interface with their relevant competency profile data. The competency profiles align the captured experiential data with representative levels of competence based on a trained mathematical model. These profiles represent the critical Knowledge, Skill and Behavior representations at the individual and team level, and are established across all defined tasks and supporting roles that require proficiency for success. Through ontological mappings, the competency frameworks align with existing training resources and environments. Ultimately, a recommender engine will be used to build a training strategy that assists unit staff in identifying tasks and competencies a training plan should target, and the sequence of resources and experiences to attain an expert proficiency rating. Once a set of tasks are selected, an exercise design process is completed that aligns task events and triggers within a specified terrain, and with calibrated virtual entities. The output of the exercise design process is a Training Support Package that is used to prepare both the simulation environment and all associated assessment requirements.

Prepare

During the Prepare Phase (see Figure 2), multiple processes and procedures are initialized. When conducting preparation activities for an operational training event, the two primary goals in this use case are: (1) preparing, and automating where possible, the collective training event based on the established trainingPlan product produced in the workflow above, and (2) preparing individuals and team elements for successful outcomes during resource intensive training. In scenario preparation, the Synthetic Training Environment is enabled to import the TSP and manage the final scenario authoring interactions. This includes automatically loading the appropriate terrain formats from the One World Terrain database, and configuring the placement of entities and objects as defined in the exercise design. This process will set the scene of the training environment when the scenario is initialized, with defined behaviors and triggers to support the selected task and dependent learning objectives.

In parallel to this process, the authored TSP is used to manage the configuration of the underlying intelligent tutoring and assessment logic that will drive the training management at scenario run-time. With GIFT, this involves building a Domain Knowledge File (DKF) that aligns competency knowledge, skill and ability metrics with scenario tasks and mission requirements. The DKF is set-up to provide both Measures of Performance (i.e., formative assessments during technical performance) and Measures of Effectiveness (i.e., summative assessments on KSA and task objectives), with templated versions in place based on competency frameworks that are represented at the team task level. When a task or set of tasks are selected, the aligned assessment templates are then populated with scenario specific information to enable the logic for that given instance of training. The goal is to automate as much of this as possible through the TSP authoring workflows, but the Prepare activities allow unit staff to visualize and verify the calibrations before moving into the Execution Phase. This also includes an ability to add additional instructional strategies, scenario adaptation scripts, and feedback prompts that can used to coach trainees and adapt the experience.



2.1 Ecosystem data is used by STE to auto-select/populate components, including terrain, game world state, and assessment framework based on competency structures.

2.2 Staff and Trainer edits STE components and configures assessment logic to automate performance tracking and enable adaptive training functions.

2.3 Trainees receive auto-assigned and personalized ecosystem learning resources based on competency estimations across training objective skill requirements.

2.4 Trainer (OC/T) uses **ecosystem** to examine unit competency and performance trends, and reviews outcomes from activities completed in 2.3.

2.5 STE training management tools receives updated competency information and highlights specific knowledge, skills and behavior that should be addressed and monitored

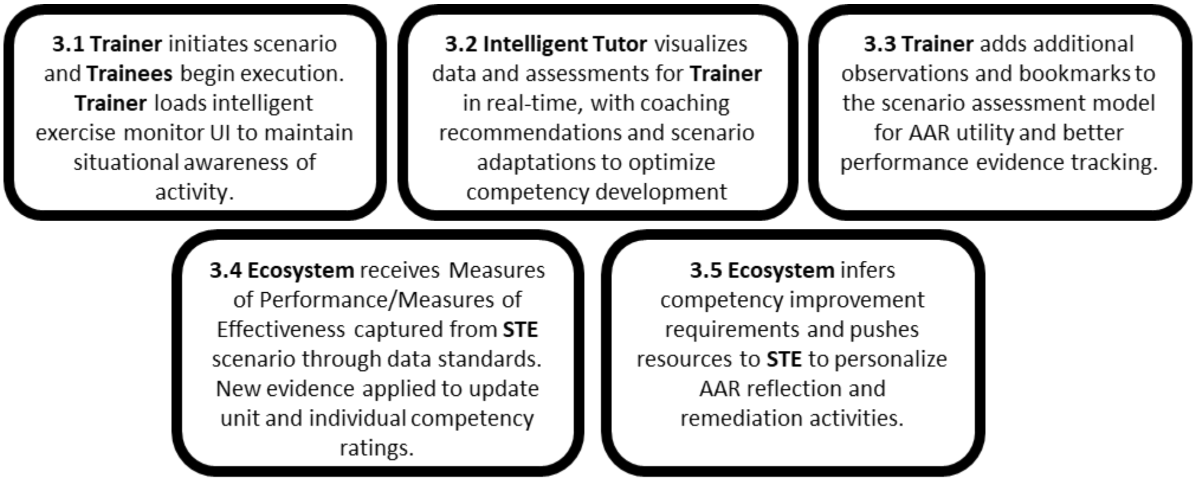
Figure 2. PREPARE Activity Workflow

During scenario preparation, trainee preparation is also activated. From this perspective, the goal here is to make sure the identified trainees are in a ready-state to benefit from a resource intensive collective exercise. Utilizing a back-end ecosystem infrastructure, digitized institutional content can be delivered at the point of need to assert and remediate role and team relevant competencies. Providing personalized content at the individual and sub-team level, informed by the individual or team's current competency state, will enable a trainee to address their knowledge and skill gaps before performing in a more demanding operational context. By virtue of linking any competency to a cloud-based learning content ecosystem, an array of learning resources can be instantly available and aligned to representative KSAs to support the just-in-time competency preparation provided by GIFT. This can involve a wide array of materials that can be delivered over a distributed cloud-based network, including multi-media content, conversational agents (e.g., AutoTutor; Nye, Graesser & Hu, 2014), skill-specific intelligent tutoring systems, and micro-simulations for enhanced exposure. The intent is to personalize these pre-experience learning engagements based on one's current competency ratings as they relate to the targeted competency defined in the Plan phase. This process could also include building a customized schedule of interactions across a network of physical training resources that are dependent on travel and access requirements.

Following this sequence of personalized interaction, all data is sent back to the ecosystem for proper experiential tracking. The competency management system consumes this data and updates associated learner profiles. This update can then be accessed by unit staff and trainers to examine outcomes from these preparation activities, and to identify elements that will be addressed during execution. This provides further insight into the KSAs a unit possesses with an objective to use this information to drive deliberate practice principles (Ericsson, 2002; 2006).

Execution

During the Execution Phase (see Figure 3), a selected and configured exercise-scenario is loaded and initialized into GIFT and an integrated STE Training Support Software (TSS) that consists of the synthetic rendering engine, and the 3D models and their respective responsive calculated behaviors and artificial intelligence. Once the trainer initiates the synthetic exercise, all exercise-registered trainees will begin actively engaging with the TSS produced synthetic world based on an in-exercised received an exercise-prescribed OPORD (i.e., Operational Order), that includes a situation, a map-based outline of their mission and movement directives, the resource and time limitations and a degree of intelligence (which will correspond to the exercise difficulty).



3.1 Trainer initiates scenario and **Trainees** begin execution. **Trainer** loads intelligent exercise monitor UI to maintain situational awareness of activity.

3.2 Intelligent Tutor visualizes data and assessments for **Trainer** in real-time, with coaching recommendations and scenario adaptations to optimize competency development

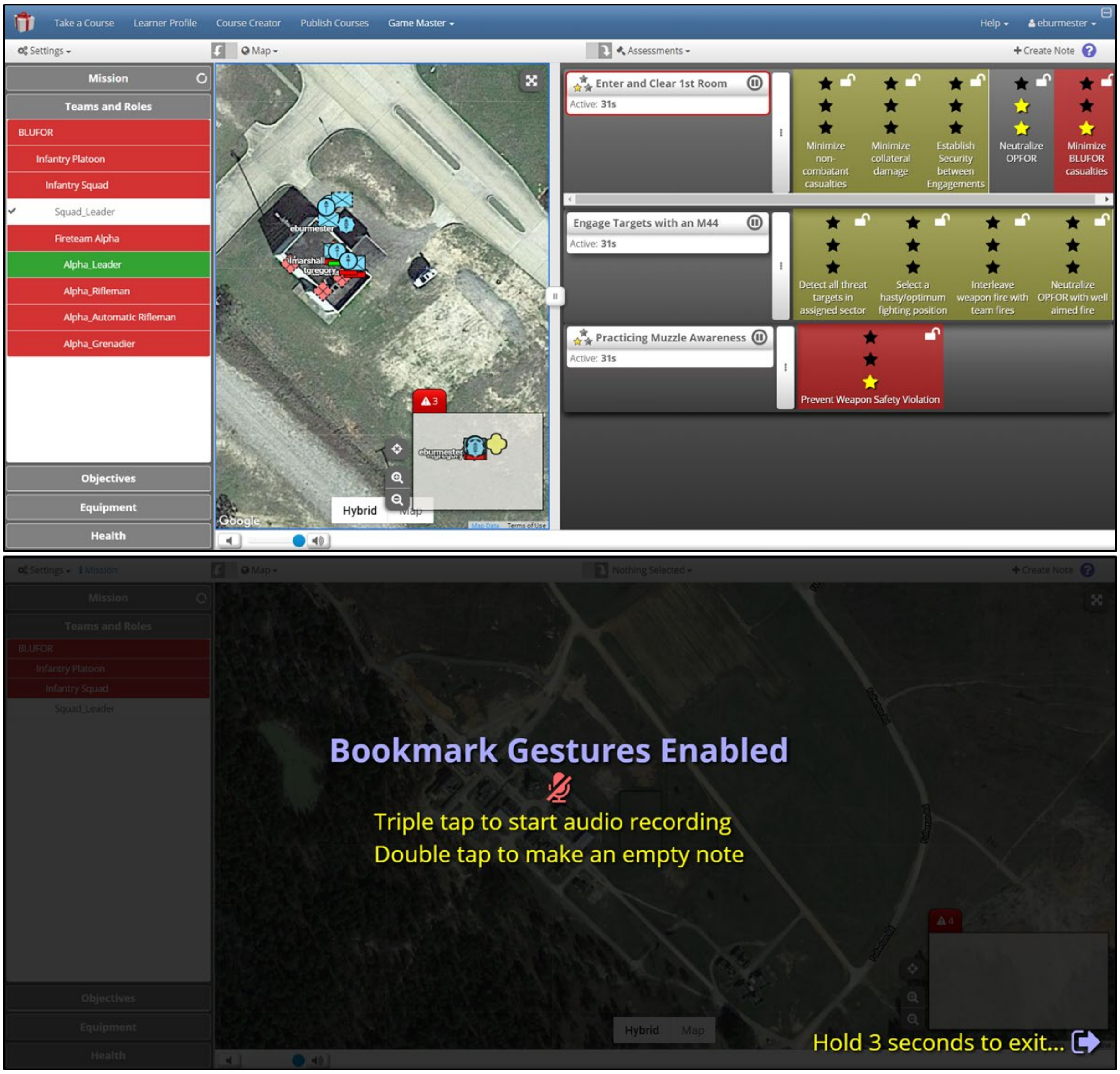
3.3 Trainer adds additional observations and bookmarks to the scenario assessment model for AAR utility and better performance evidence tracking.

3.4 Ecosystem receives Measures of Performance/Measures of Effectiveness captured from **STE** scenario through data standards. New evidence applied to update unit and individual competency ratings.

3.5 Ecosystem infers competency improvement requirements and pushes resources to **STE** to personalize AAR reflection and remediation activities.

Figure 3. EXECUTE Activity Workflow

In addition, the observer, controller/trainer staff (OC/T) can access an exercise control interface (e.g., GIFT Game Master; see Figure 4). This device based wireless capability allows for artificial intelligence support real-time interaction monitoring. At the same time, (OC/T) can use this device to track and interact with the assessments that are being captured while active tasks are completed. This drives a hybrid model to performance capture (Goldberg, Hoffman & Graesser, 2020), where there is a blend of automated and observer-driven assessments. As the adaptive instructional components of STE mature, we envision more and more automation at the assessment level, but we provide an ability to mark specific items that require human input, as well as a mechanism to override automated inferences if deemed inaccurate.



Bookmark Gestures Enabled

Triple tap to start audio recording

Double tap to make an empty note

Hold 3 seconds to exit...

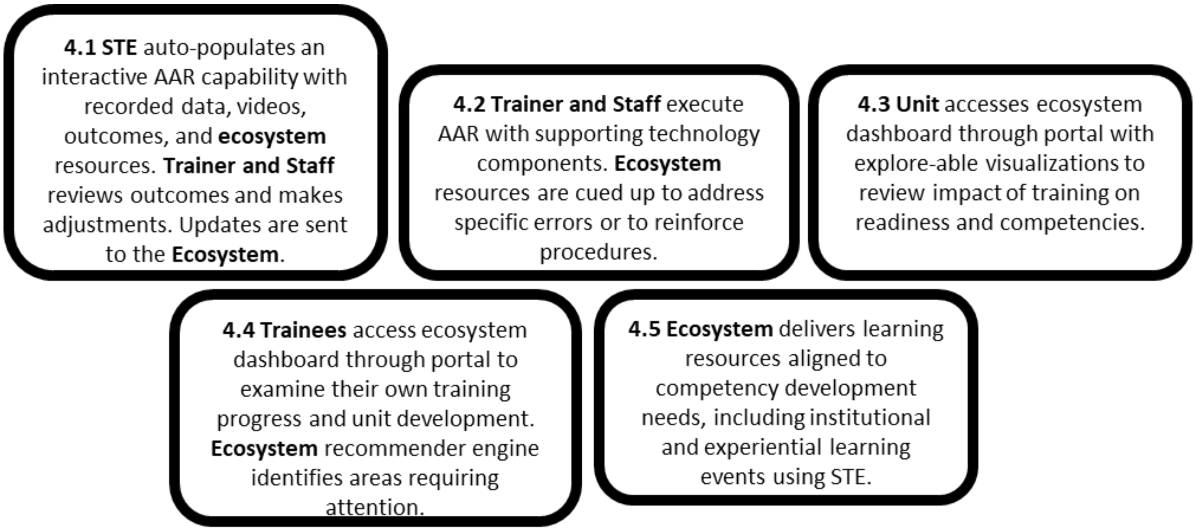
Figure 4. Game Master during the Execution Phase (Top Image) and Game Master in Gesture Book Mark mode (Bottom Image)

To support this function, the Game Master must provide a mechanism to drop observation bookmarks while maintaining full awareness of the environment (i.e., leaving bookmarks without looking away from the engagement scene). This is enabled through a Bookmark Gesture mode (see Figure 4) that allows an observer to annotate the time log with a flag for further examination following completion of the scenario. The user also has the ability to create an audio recording linked to bookmark event for the purpose of capturing better context around that bookmarked event. These annotations are then used to examine the entities or activity marked following the scenario, with specific updates to the relevant nodes in the assessment structure.

While the assessment model is being updated at run-time based on aggregated data, all computed metrics and evaluations are reported out to the learning ecosystem through a set of established data standards and specifications as they relate to the experiential tracking requirements to monitor performance and competency proficiency over time. This data is used to establish additional evidence to that unit's set of competency profiles. This Execution Phase export of data is used to track evidence to build confidence in current competency assertions, as well as identifying specific competency KSAs that need to be addressed in the Assess Phase's After Action Review (AAR). With a direct connection to aligned learning resources, the ecosystem can push supporting content to the edge instance of training for inclusion in their reflective assessment activities. This data push finalizes the Execution Phase, with the next workflows focused on training assessment.

Assess

The assessment phase of the PPEA process includes an after action review that consists of two sub-segments: an analysis or review segment and a reflection segment. During the review segment, the Game Master tool is used to carefully review and playback the logged interaction data and the automated assessments made from that data during the exercise execution phase. The OC/T can confirm or modify these assessments based on any introduced error in the assessment logic. This segment also includes reviewing the annotated bookmarks placed by the OC/T during the execution phase, and inputting any observer required assessments.



4.1 STE auto-populates an interactive AAR capability with recorded data, videos, outcomes, and **ecosystem** resources. **Trainer and Staff** reviews outcomes and makes adjustments. Updates are sent to the **Ecosystem**.

4.2 Trainer and Staff execute AAR with supporting technology components. **Ecosystem** resources are cued up to address specific errors or to reinforce procedures.

4.3 Unit accesses ecosystem dashboard through portal with explore-able visualizations to review impact of training on readiness and competencies.

4.4 Trainees access ecosystem dashboard through portal to examine their own training progress and unit development. **Ecosystem** recommender engine identifies areas requiring attention.

4.5 Ecosystem delivers learning resources aligned to competency development needs, including institutional and experiential learning events using STE.

Figure 5. ASSESS Activity Workflow

During the review segment, the OC/T can adjust and update reported automated assessments using user interfaces in the Game Master interface as shown in Figure 6. These are then sent out to the ecosystem data repository and CMS as “revisions” to the original assessments done at the end of the execution phase (i.e., meaning the original assessment are still kept for accountability purposes). The review segment can also be used to train the automation by labeling by correctly labeling the automation where it missed, was wrong or less accurate in its activity detections.

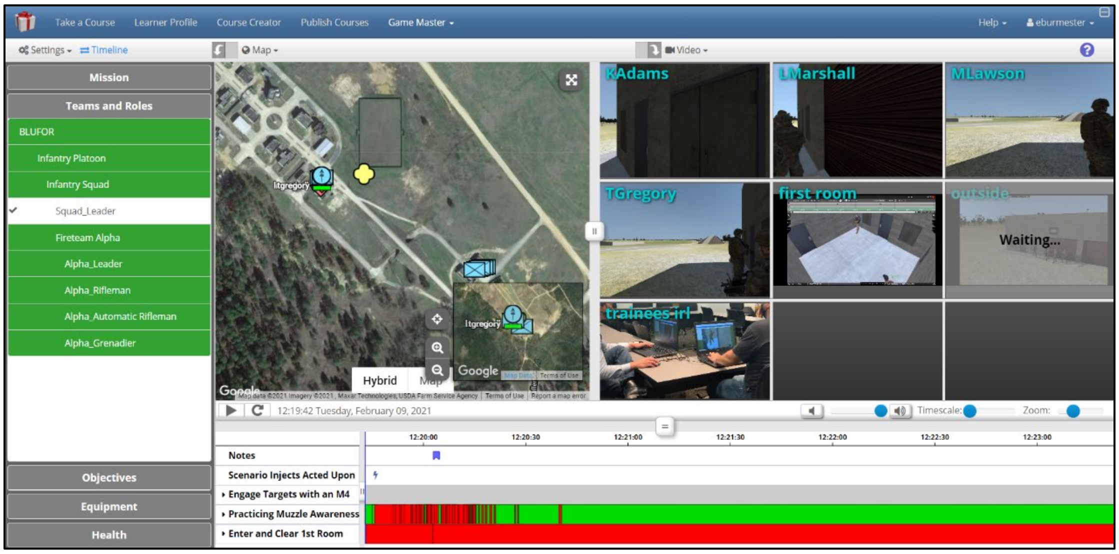


Figure 6. GIFT Game Master in After Action Review playback mode for reflection and discussion.

After the review segment, a reflection segment begins. This is when the trainees are presented with the feedback of their evaluations, using the precision of the GIFT competency based evaluation, and in context of "reliving" their past performance with the real data. A trainer or an ecosystem based automated agent skilled in facilitating reflection will not only guide learners through their past performance, but prompt and point-out where notable good and bad performance occurred, provide cross-team awareness intended to build teamwork, and will facilitate the constructive acceptance of the feedback so that positive reflection and discussion occurs.

Following the review segment, the trainer will determine if further scenario exposure is required or if the training event is complete. This will automatically update the unit's schedule and prompt the unit commander to review the impact on the team's competency states. The commander will be able to not only view the results on a unit tailored competency dashboard, but they will also be able to revisit the exercise themselves and review the same performance data the learners reviewed using the Game Master tool. This allows the commander to have a "first-person" view of the performance to help them make their own assessment if the resulting competence states are accurate.

At the same time, the trainees who performed the exercise can always access the same competence-state dashboard now tailored for their own specific history in assigned teams and roles. This is how they will remain aware of progress toward assigned team- roles and/or weapon qualifications. This is also how team leaders and individuals will become aware of any newly assigned training based on evaluated gaps or upcoming periodicities when specific competencies need to be "re-evaluated" based on a computed atrophy rate. Through this mechanism, the ecosystem connection produces a technology-enabled self-development model.

Experiential learning

The reflection segment discussed above is the most critical point in the experiential learning cycle (Kolb, 1993), shown in Figure 7 below. Aside from being the point when learning is maximized, it also requires the trainee to be skilled or facilitated in learning through a meta-cognitive and reflective cognitive processes, as well as focusing on emotional intelligence (Goleman D, 1996) to maximize the learning process.

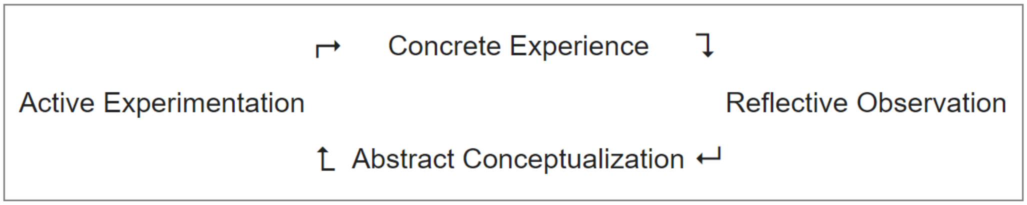


Figure 7. Experiential Learning Cycle

As illustrated above and discussed earlier, the concrete experience gained during a realistic and prompted execution phase is recorded and evaluated. The recorded data and evaluation is reviewed by the learner and facilitated by a skilled trainer as discussed earlier. During this reflective segment of the assess phase, learners are allowed to directly observe their performance improvements and gaps in competence compared to a population-normalized minimum and expert standard, which will trigger the conceptualization of alternate performance activity and strategies. These alternate actions will become active experimentation follow-on exercise-scenario with similar performance conditions. This cycle continues until the organization or self-actualized performance standards are met.

Conclusion

In this paper we provided a forward-leaning use-case where GIFT was employed in an experiential operational training process, that involves existing or conceptual instruction and evaluative features that leverages the future STE. We also discussed the experiential learning cycle that will be integrated as part of the operational training process and how that approach to competency development can be facilitated within a learning ecosystem. This is achieved by establishing representative competency standards across individual and team structures, automated highly perceptive measurement features, and a statistical competence-state outcome that feeds into the unit training management and team / individual performance management process.

References

Barr, A., Fletcher, J.D. & Morrison, J. (2020). The Advanced Distributed Learning's Total Learning Architecture: IDA Report. Technical Report AD1107614. Downloaded from DTIC at https://apps.dtic.mil/sti/citations/AD1107614.

Ericsson, K. A. (2002). Attaining excellence through deliberate practice: Insights from the study of expert performance. In M. Ferrari (Ed.), The pursuit of excellence in education (pp. 21–55). Hillsdale, NJ: Erlbaum

Ericsson, K.A (2006). The Influence of Experience and Deliberate Practice on the Development of Superior Expert Performance. The Cambridge handbook of expertise and expert performance, Pg. 38

Goldberg, B, Hoffman, M. & Graesser, A. (2020). *Adding a Human to the Adaptive Instructional System Loop: Integrating GIFT and Battlespace Visualization*. In (Eds: A. Sinatra, A. Graesser, X. Hu, B. Goldberg & A. Hampton) Design Recommendations for Intelligent Tutoring Systems, Volume 8: Data Visualization. U.S. Army DEVCOM Soldier Center, Natick, MA.

Goldberg, B., Hellman, K., Owens, K. & Gupton, K. (2020). Follow the Data: From Raw to Credential in Experiential Training. Presented at the 2020 iFest Conference, Alexadria, VA.

Goleman D (1996). Emotional Intelligence: Why It Can Matter More Than IQ. Bantam Books. ISBN 978-0-553-38371-3

Kolb, D. (1984). Experiential Learning: experience as the source of learning and development. Englewood Cliffs, NJ: Prentice Hall. p. 21

Nye, B., Graesser, A. & Hu, X. (2014). AutoTutor and Family: A Review of 17 Years of Natural Language Tutoring. *International Journal of Artificial Intelligence in Education*, 24, 427-469.

Owens, K.P. (2021). Competency-Based Experiential-Expertise and Future Adaptive Learning Systems. Adaptive Instructional Systems. Adaptation Strategies and Methods. Springer Nature Switzerland AG 2021

Owens, K.P, Goldberg, B. et.al (2020). Re-Thinking the Tactical Small Unit Synthetic Training Model. In Proceedings of the *Interservice/Industry Training Simulation & Education Conference*, Orlando, Florida, December 2020.

Walcutt, J.J. & Schatz, S. (Eds.) (2019). *Modernizing Learning: Building the Future Learning Ecosystem*. Washington, DC: Government Publishing Office.

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S Army Combat Capabilities Development Command – Soldier Center – Simulation and Training Technology Center. The statements and opinions expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHORS

***Dr. Benjamin Goldberg** is a Senior Scientist at the U.S. Army Combat Capabilities Development Command (DEVCOM) Soldier Center, Simulation and Training Technology Center (STTC) in Orlando, FL. Dr. Goldberg is the team lead for a research program focused on the development and evaluation of Training Management Tools for future Army training systems. His research is focused on the application of intelligent tutoring and artificial intelligence techniques to build adaptive training programs that improve performance and accelerate mastery and readiness. Dr. Goldberg has researched adaptive instructional systems for the last 12 years and has been published across several high-impact proceedings. He holds a Ph.D. in Modeling & Simulation from the University of Central Florida*

***Kevin Owens** is an Engineering Scientist at the Applied Research Laboratories, The University of Texas at Austin – a US Navy University Affiliated Research Center (UARC). After retiring from the US Navy, Mr. Owens earned a BS and MS in Instructional Design and has over 20-years' experience in DoD learning engineering science and technology (S&T) and DoD operational research. Mr. Owens supports US Army PEO-STRI and the Synthetic Training Environment (STE) requirements process, the Squad Immersive Virtual Trainer (SIVT) and Integrated Visual Augmentation System (IVAS) engineering team; STE Experiential Learning for Readiness (STEELR) S&T applied research; and US Navy learning engineering applied research.*

Assessing, Visualizing, and Communicating Competency Aggregations and Profiles with GIFT

Zachary Heylmun, Michael Kalaf, Christopher Meyer, Lucy Woodman
Synaptic Sparks, Inc.

Introduction

This paper explores a volunteer effort to integrate and experiment with several topics of study relating to the relatively new functionality prototyped into the Generalized Intelligent Framework for Tutoring (GIFT, https://www.gifttutoring.org/) regarding Competencies, Skills, Knowledge, Attitudes, and other traits (otherwise referenced as KSAOs). Building on a GIFT external application integration effort in 2020 with the Competencies and Skills System (CaSS, https://www.cassproject.org) and a formal standard of network communication such as the xAPI Standard (https://adlnet.gov/projects/xapi/) - we wished to further integrate complimentary software tools to enhance training and assessments that utilize formal competencies for individuals and teams utilizing GIFT.

This paper further describes experimental hypothesis and initial prototype efforts for new assessment methodologies that we believed could be achieved through the further integration of the above systems. In addition to GIFT and CaSS, new software suites compatible with GIFT are being developed in the community every day. Therefore, as efforts progressed, we also experimented with adding data to the pre-existing GIFT interface to a Learner Record Store (LRS) called Learning Locker, studied integrating with an additional Learning Management System (LMS) called Moodle supporting xAPI (https://docs.moodle.org/dev/Experience_API_(xAPI)), and began to send individual and team data to the Data Analytics and Visualization Environment (DAVE, https://yetanalytics.github.io/dave/#/). It was our hypothesis that GIFT, CaSS, an LRS/LMS instance, and DAVE could be used in conjunction to roll-up individual and team competency assessments to allow GIFT course authors and educators to instruct their students/trainees better and with higher degrees of focus, whatever the training domain may be.

The entire scope proposed above goes beyond our efforts for GIFTSym9, and therefore we limit ourselves to this short paper that should serve as multiple starts/continuations to discussions between any and all parties. For this paper, we have only begun to explore all possible enhancements made primarily to GIFT that will allow for the continued creation of the software framework with which to continue testing the above hypothesis. Through utilizing high-quality metadata tagging for courses created in GIFT, creating a CaSS database interface capability through GIFT, cross-referencing GIFT course assessments with CaSS Learner Profiles with formal qualifications, and aggregating tagged experimental GIFT data for individuals and team visualization using DAVE, we have shown that such a technical pipeline is feasible and leave it to the community to decide on its further direction and value.

It is our hypothesis that by taking all previously mentioned integration efforts and enhancing the communication and cross-functional use of each software suite that unrealized value with regards to team mission readiness assessment can be attained. This is similar in philosophy to the Advanced Distributed Learning (ADL) Total Learning Architecture (TLA) paradigms of research and development (https://adlnet.gov/projects/tla/). With U.S. Army modernization priorities in mind and emerging program groups such as STEEL-R and STRIDE supporting the Synthetic Training Environment, the authors now present the various software suites and references from which the reader may begin to build their own framework if desired. Otherwise, the works described in this paper will be made publicly available following GIFTSym9 and as our Return-to-Work plan allows for the disparate systems to be finally brought together and formally demonstrated. Conclusions and community interest shall help determine further research and experimentation as future programmatic priorities allow.

Since some of these integrations are in their infancy and all are constantly evolving, the demonstration will not be given live during GIFTSym9 due to the continued effects of the pandemic and rudimentary nature of some of the interfaces between systems. We hope to make the formal conglomerate software suite package portion of the project polished and available to the community as early as 3Q 2021. This timeline will allow for SSI to bring all employees and physical development machines back under one roof to allow for greatest ease of development for this many disparate systems working together.

PARTS LIST AND DESCRIPTIONS

GIFT Software Suite

At the time of this writing, GIFT 2021-1 has been released at www.gifttutoring.org/projects/gift/files. The reader may install this version of GIFT by creating a free GIFT account if they have not done so already. Once downloaded, the reader will be able to install their own local GIFT server and configure it to their liking. Readers can find configuration instructions included with their download as well as detailed discussions on the 'Forum' tab at the www.gifttutoring.org homepage after making their own account.

Of special note is the CaSS interface portion of the new GIFT codebase, also available upon request. The current GIFT implementation connects to a CaSS instance formally deployed by the CaSS Team, powered by Eduworks (https://eduworks.com). The reader can access the new GIFT-CaSS functionality through running the GIFT software and accessing KSAOs through GIFT Course Properties and Competencies / Skills.

CASS Competencies and Frameworks

To quote the CaSS homepage, "CaSS provides a robust common language and translation method for competencies, evidence of attainment, and associated resources." In short, the CaSS software has allowed GIFT to begin to have a formal, external framework with which to match Learner-to-Course-to-KSAOs. This linkage is accomplished through properly-formed xAPI statements. In these cases, GIFT is assumed to generate xAPI statements on a to-be-defined basis, but for the purposes of this paper were generated simply to prove the existence of the pipeline. CaSS then can consume these xAPI statements to form assertions (Competency Based Learning, 2019). For more information on how GIFT and CaSS work together, the reader may request the "Data Flow from GIFT to CaSS Assertions" information from the STEEL-R Modelling Group, or from the POCs of this GIFTSym9 event.

If the reader wishes to instantiate and run on a different CaSS server; a server maintained by the CaSS team can be found at https://cassproject.github.io/casseditor/. By following this link, readers may explore the site to edit and configure their own competency and mastery framework. This server is maintained by the CaSS team, but any user has full permissions to create and edit their own framework. Readers may also register with the CASS project at https://www.cassproject.org, download the open-source code from the referenced GitHub project, and build/configure/maintain their own CaSS server.

Experience API (xAPI) Syntax and Models

xAPI is a specification that collects and stores information about the user's learning experience, such as what they participated in and what score they achieved for that activity (Poltrack, Robson, 2017). xAPI expresses these Actors, Verbs, and Objects as statements. CaSS then retrieves these statements from the Learning Record Store (LRS); or sometimes an LMS which contains a database that stores this learner activity data (Barr, Robson, 2019). Finally, CaSS can then assert the level of mastery achieved for those competencies (Gordon, Hernandez, & Smith, 2019).

The specific syntax of xAPI statements imported and exported with regards to GIFT and STEEL-R is still being formally defined, but for this effort the team was able to put together very simple examples to prove initial implementations of the generic communication structure are sound. For instance, GIFT always has knowledge of which Learner ID can be an Actor, whether or not an Assertion is true, and which level between Below/At/Above Expectations a tagged KSAO has been assessed at. Thus, a statement such as 'Learner01 has SCORED ABOVE EXPECTATIONS at ENGAGE TARGET 04' can be a valid syntactical xAPI statement that is understood by GIFT, CaSS, an LRS, and DAVE. Each system may end up evolving to need slightly more or less information, but that is why the dictionary of terms continues to evolve with each of the systems in mind.

And, each system may contribute information upon receiving/sending an xAPI statement that it possesses that the sender of the triggering xAPI message may not have access to. For instance, an LMS may have knowledge of a course curriculum, classroom roster, or instructor access lists, and thus aggregate individual performance data from GIFT or competency assessment assertions from CaSS into a team-oriented data structure for viewing in a software suite such as DAVE.

It is this initial experimentation with the various xAPI messages that the SSI team has begun experimenting with, merely to start viewing concrete examples of how all disparate systems may end up working together on a syntax and data sharing level.

GIFT and xAPI

With an appropriately authored course, GIFT takes the data received from the learner during a course execution and creates xAPI statements from that data. GIFT then sends the xAPI statements to the Learning Locker LRS, or can have its ActiveMQ messages piped to additional sources such as CaSS, DAVE, or Moodle. For this paper, the team used ActiveMQ and MQTT to send existing xAPI messages and xAPI message 'stubs' to other software frameworks for consumption. To perform just a connection to the LRS, a connection between GIFT and the LRS must be established via configuration files. Following the instructions at https://gifttutoring.org/projects/gift/wiki/Configuration_Settings_2021-2, we supplied our LRS configuration file (GIFT/config/lms/lmsConnections.xml) with the information necessary to connect to the LRS. Also, additional messaging middleware commonly used to listen for JSON messages on certain ports can be used to quickly pipe network messages to the other sources mentioned above, paying special attention to format messages in xAPI format.

Data Analytics and Visualization Environment (DAVE)

The reader is able to access GitHub's storage of the YetAnalytics DAVE project here: https://github.com/yetanalytics/dave. This project is open source, and a major effort of the ADL Initiative: https://adlnet.gov/projects/dave/. This software, combined with additional xAPI-formatted messages and database access, formed a new way to begin visualizing team performance data. The reader should be able to also access an online Alpha demo of DAVE here: https://yetanalytics.github.io/dave/#/. This demo allows a user to tour functionality, create their own visualization queries, and upload/visualize their own LRS data with the proper connection info.

Methodologies

At the heart of this volunteer effort was a cutting-edge proof of concept with GIFT, CaSS, DAVE, an LRS/Moodle, and xAPI-compliant messaging services operating between all systems. Through a combination of code study, basic REST command insertions into GIFT, piggybacking off of existing ActiveMQ implementations, CaSS and DAVE LRS pipelines, and messaging stubs via JSON tools such as Postman and

Visual Studio REST Client Simulations, SSI was able to see the very beginnings of the stages of communication between disparate systems result in new visualizations.

Even as SSI began this project, requirements and models from the GIFT team, ADL, STEEL-R, STRIDE, and other organizations continue to evolve exact syntax and dictionaries on an architectural level. The point of this project, however, was simply to begin "external" party experimentation with communication paradigms and begin identifying areas for additional capabilities to be added as development continues. As SSI continues to closely monitor technological requirements from stakeholders, we will continue to experiment from a holistic community perspective.

Finally, with all systems operational albeit with some local and some cloud-based, SSI was able to begin aggregating anonymized learner data with regards to faux team organizations and see proof that the architectural system concept described throughout the paper is not only feasible, but completely possible as the technologies stand today. The full implementation will again be made available once all messaging 'stubs' are removed from the project and replaced with formally-vetted interfaces with the authorization of each owner of each software suite, and after the pandemic effects lessen to the point of returning physically to an office environment.

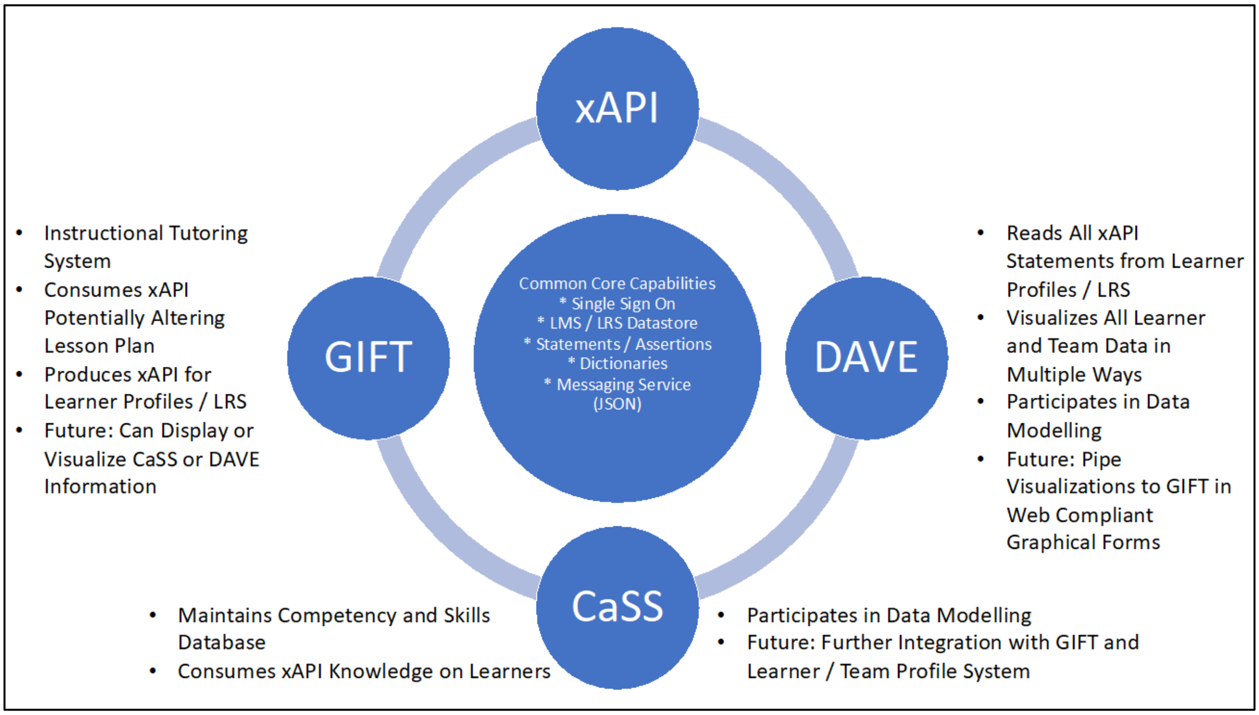

xAPI
GIFT
DAVE
CaSS
Common Core Capabilities
* Single Sign On
* LMS / LRS Datastore
* Statements / Assertions
* Dictionaries
* Messaging Service (JSON)
• Instructional Tutoring System
• Consumes xAPI Potentially Altering Lesson Plan
• Produces xAPI for Learner Profiles / LRS
• Future: Can Display or Visualize CaSS or DAVE Information
• Reads All xAPI Statements from Learner Profiles / LRS
• Visualizes All Learner and Team Data in Multiple Ways
• Participates in Data Modelling
• Future: Pipe Visualizations to GIFT in Web Compliant Graphical Forms
• Maintains Competency and Skills Database
• Consumes xAPI Knowledge on Learners
• Participates in Data Modelling
• Future: Further Integration with GIFT and Learner / Team Profile System

Figure 2 - GIFT, CaSS, DAVE Unique and Common Features Regarding Project (Non-exhaustive)

Conclusions and Recommendations for FUTURE research

This paper has described the software engineering system setup and basic configurations for new value we believe can be achieved through the further messaging configuration, system combination and integration. Knowing that xAPI was the syntactical medium through which the disparate software suites may communicate through, the SSI team implemented and proved that such communications are possible with the technology of today. Furthermore, extending these communication systems and implementing initial *hypothetical* xAPI dictionary terms wrapped in JSON/REST message calls, the team began to see how visualizations can be created for individual or teams. Identifying common data between GIFT, CaSS, DAVE, and xAPI standards, SSI can now proceed with further experiments to assist stakeholders in validating and defining statements in xAPI to meet all system and stakeholder goals. Keeping GIFT programmatic goals, U.S. Army modernization priorities, and emerging program groups such as STEEL-R and STRIDE in mind, Synaptic Sparks is happy to continue discussions for future system integration efforts, visualization validation, and the continued evolution of the GIFT software suite.

References

Barr, A. & Robson, R. (2019). Missing pieces: Infrastructure requirements for adaptive instructional systems

Blake-Plock, Shelly. (2019). Data Analytics and Visualization Environment for xAPI and the Total Learning Architecture: DAVE Learning Analytics Algorithms. Yet Analytics, Inc.

Brewer, V., Ray, R., Robson, E., & Robson, R. (2020). Competency framework development process report. Eduworks Corporation Corvallis United States.

Goldberg, B., Lester, J., Mott, B., Pokorny, R., Rowe, J., & Spain, R. (2020). Toward data-driven models of team feedback in synthetic training environments with gift.

Gordon, J., Hernandez, M., & Smith, B. (2019). Competency-based learning. Advanced Distributed Learning Initiative.

https://gifttutoring.org/documents/154. Design Recommendations for Intelligent Tutoring Systems. Volume 8. Data Visualization. Accessed May 24, 2021.

ABOUT THE AUTHORS

Mike Kalaf*: Mike has over 30 years of Modeling, Simulation and Training leading large scale efforts leveraging cutting edge technology. Mike has worked in the commercial and military aviation, training and simulation business. Mike has led several programs integrating "state of the art" technology and delivering highly successful technology and business innovation. Mike's formal education includes an earned Mechanical Engineering degree from Rochester Institute of Technology, RIT.*

Christopher Meyer*: Christopher received his Bachelor and Master of Science degrees in Computer Science from Kansas State University, also receiving minors in Economics and Modern Languages. Chris also studied abroad for a year during a tour in Japan at Chukyo University dedicated to the specialized study of Artificial Intelligence. After completing traditional education phases, Chris was employed at Lockheed Martin for 10 years working together with representatives from the Departments of Defense, Health and Human Services, Energy, and Education to assist in the creation of solutions to solve challenges at a national level.*

Lucy Woodman*: Lucy graduated from Seminole State College with a Bachelor of Science in Information Technology. Lucy has supported Synaptic Sparks for three years in addition to a successful internship and transitioned to a be a major supporter of big data services within SSI in December of 2018. Lucy is a certified Amazon Web Service specialist in addition to being a certified System Architect.*

Theme III: Measurement and Assessment

GIFT External Assessment Engine for Analyzing Individual and Team Performance for Dismounted Battle Drills

Caleb Vatral[1], Naveeduddin Mohammed[1], Gautam Biswas[1], and Benjamin S. Goldberg[2]
Vanderbilt University – Institute for Software Integrated Systems[1], U.S. Army DEVCOM Soldier Center[2]

Introduction

Dismounted battle drills (DBD) such as "Enter and Clear a Room" and "React to Direct Fire Contact" are essential operations in urban warfare conducted by the armed forces. These operations require soldiers to develop effective psychomotor and cognitive skills and strategies to quickly assess and react to situations, and the ability to work in teams to accomplish desired objectives. DBDs are dynamic and may evolve rapidly, therefore, soldiers must make rapid decisions, relying on muscle memory and individual and team-oriented behaviors they have acquired through training. To become proficient in their application of skills and strategies, it is desirable for soldiers to repeatedly train on relevant scenarios.

To support this experiential requirement, the U.S. Army has been modernizing its core Synthetic Training Environment (STE) to facilitate repeated training sessions. In this context, novice soldiers and teams may need guidance and direction to help them rapidly develop their psychomotor and cognitive skills and strategies. However, instructors and Observer/Controller Trainers (OC/Ts) often do not have opportunities to provide performance-oriented feedback during the training sessions when groups of soldiers rapidly go through the training scenario at repeated intervals. To address this problem, we are developing automated assessment mechanisms and metrics that can provide evaluative feedback to the soldiers after they complete a sequence of training exercises. Simultaneously, we will supplement the evaluative metrics by providing video evidence to support instructors in after action reviews (AARs).

As a first step toward developing tutors, we conduct an initial study of the "Enter and Clear Room (ECR)" DBD that relies heavily on team member psychomotor skills and cognitive skills, such as identifying and differentiating enemy combatants from noncombatants in the room, and providing cover for the other team members (U.S. Department of the Army, 2011). In addition to tactical skills, a squad also needs to develop strategic reasoning and decision making skills that are derived from situational awareness and planning to assure superior firepower inside and outside the building, determine the method of access into the building and rooms of interest, and to control the tempo of the operations (Holmquist & Goldberg, 2007). These tasks are typically performed by fire teams of three or four soldier, therefore, it is very important that the trainees acquire team skills in addition to the individual task skills (Sinatra, Kim, Johnston, & Sottilare, 2018). The need to combine individual psychomotor skills, cognitive and strategic processes, along with teamwork introduces a number of complexities in designing training scenarios and evaluating individual and team performance. The need to evaluate psychomotor, cognitive, strategic, and affective processes implies the need for multiple monitoring modalities, such as computer logs of individual and team performance, video analysis for analyzing psychomotor and cognitive skills, eye tracking for monitoring situation awareness, and physiological sensors to capture affect. Multi-modal data capture becomes even more critical when monitoring and assessing complex teamwork.

In this work, we propose and develop the first iteration of an external assessment engine (EAE) for the GIFT (Generalized Intelligent Framework for Tutoring) to generate automated performance metrics of soldiers executing DBDs in the Army's SAM-T (Squad Advanced Marksmanship Trainer) program of record (Gant, Speidel, Tatum, & Zuelke, 2019). Our approach will combine multiple sources of data: (1) soldier-facing video that captures soldier movements and activity; (2) VBS3 video of the evolving battle drill scenario; and (3) Distributed Interactive Simulation (DIS) game state traffic related to soldier activity during a SAM-T scenario. This paper discusses the design and implementation of the EAE in conjunction with GIFT and reviews preliminary results that we have generated from our analyses.

The rest of this paper is organized as follows. Section 2 provides a brief description of the SAM-T. Section 3 provides an overview of the ECR DBD scenario and highlight the individual and team coordination skills that squads of soldiers need to develop to become proficient in executing these operational tasks. Then, Section 4 discusses our architecture for the External Assessment Engine (EAE) and its integration with the GIFT environment. Section 5 discusses the assessment metrics we have developed for tracking individual soldier and team performance skills and strategies. Section 6 then discusses our machine learning methods that we have developed for tracking soldier movements and projecting them on to a 2-D plane for assessment of soldier skills and team coordination activities. The Section 7 discusses our approach to combining the heterogeneous data from multiple sources to compute all of the soldier and team performance metrics discussed in Section 5. Last, Section 8 presents our conclusions and directions for future work.

Training Environment: SAM-T

The SAM-T is a current Army program of record and provides a Training as a Service (TaaS) solution. SAM-T (figure 1) is an augmentation (not a replacement) of the Engagement Skills Trainer (EST) II, and was designed to simulate live weapon training events that directly support individual and crew-served weapons qualification, in- cluding individual marksmanship, small unit collective and judgmental escalation-of-force exercises in a controlled environment [https://asc.army.mil/web/portfolio-item/engagement-skills-trainer-est/]. SAM-T is intended to improve and accelerate Soldier and Squad close combat skills, and task acquisition by providing the realistic repetitions in diverse complex operational environments necessary to increase readiness and overall performance. The SAM-T was the multi-modal training environment use case to iteratively develop the EAE concept against.

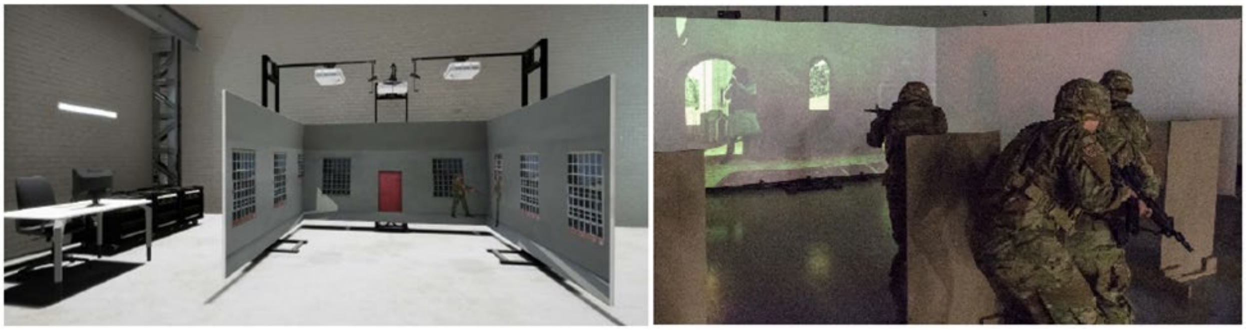


Figure 1: U-Shape System Configuration for SAM-T (Pargett, 2019)

Enter and Clear Room (ECR) Domain

Enter and Clear Room (ECR) operations are designed for small teams, i.e., teams of three or four soldiers. They represent a form of urban warfare, where the goal is to neutralize enemy personnel who have been located in a building that may also house noncombatants. The overall operations are initiated by positioning security forces around the building to secure the area. Depending on the size of the building, and the number of enemy combatants in the building, one or more fire teams of four (sometimes three) are assigned to clear and secure a specific set of rooms. The operation to enter and clear a particular room starts with the squad in formation outside a doorway or entrance. A distinct signal from the squad team leader commences the operations to seize control of the room by rapidly and tactically entering the room and neutralizing the enemy, while minimizing collateral damage to the squad, the noncombatants, and the property. To maximize chances of success, and to accomplish the ECR mission in an organized manner, the army divides up ECR missions into five major task segments: (1) Prepare to Enter, (2) Enter the Room, (3) Clear the Room, (4) Secure the Room, and (5) Complete the mission and move on to next assigned operation (Sinatra et al., 2018).

In this paper, we focus on the segments that include the steps for entering, clearing, and securing the room. This involves entering a room quickly, moving immediately to points of domination (POD) while eliminating enemy combatants with superior fire power, and once clear seize control of the room. Figure 2 illustrates the task steps related to ECR. A more detailed description of the steps in the ECR task are outlined below:

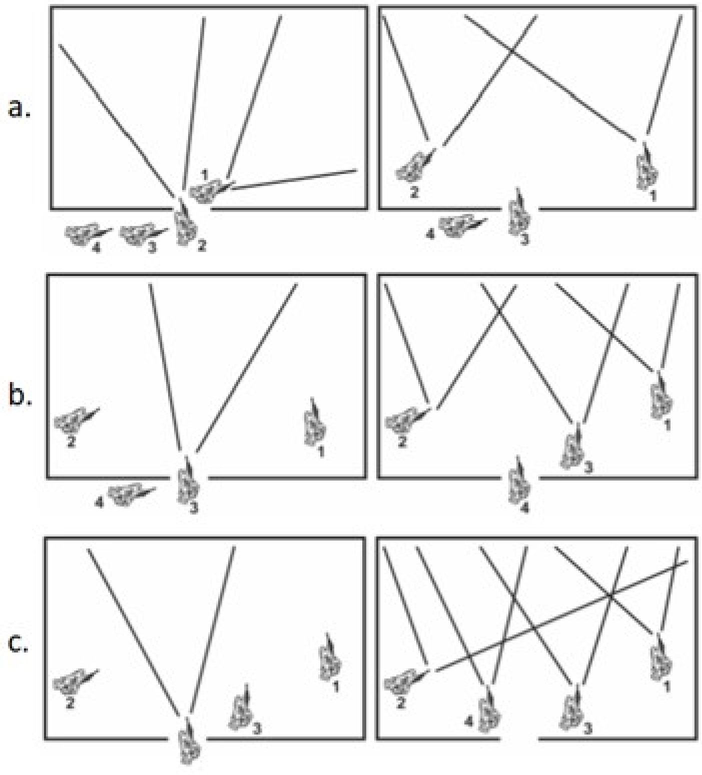


Figure 2: Enter and Clear a room: In quick succession, (a) two Soldiers enter almost simultaneously; (b) the third soldier follows; and then (c) the fourth soldier enters, and they move to their assigned points of dominance (POD).

1. The squad in tight formation readies to enter the room, checks for booby traps on the door. On a signal, usually a pat on the back or arm or an arm squeeze, to the first soldier from the team lead (who usually enters second), starts the entry process (this may require kicking down the door to facilitate entry).
2. The first two Soldiers enter the room almost simultaneously. The first Soldier enters the room and moves left or right along the path of least resistance (typically along the wall) to one of two corners. Figure 2a illustrates the soldier's trajectory. The soldier enters firing aimed bursts into his assigned sector engaging all threats or hostile targets to cover his entry. He assumes his POD (usually at the corner of the room) facing in toward the room interior.
3. The second Soldier enters the room immediately after the first Soldier. He moves in the opposite direction of the first Soldier to his POD, also firing aimed bursts to engage and eliminate all threats in that may appear in his assigned sector, and positioning himself at the POD, which is in the opposite corner of the room.
4. The third Soldier moves in the opposite direction of the second Soldier while scanning and clearing his segment of the room. Figure 2b illustrates the trajectory and POD destination for soldierthree.

5. The fourth Soldier moves opposite of the third Soldier to a POD for his sector, also scanning and clearing his assigned region. Figure 2c) illustrates this soldier's trajectory and destination.
6. After the soldiers in the squad are positioned at their PODs, they continue to scan their sectors and look for objects in the room that may hide enemy combatants. If such objects exist, the soldiers signal one another and move to engage and clear these areas of enemy combatants with precision aimed fire, while avoiding injury to the noncombatants and themselves. Utmost care must be taken so the soldiers are not killed or injured by "friendly fire."
7. The team visually scans and assesses if the room has been cleared of all enemy combatants. When convinced, the team leader announces (or sends message) to the overall leader of the operations, usually a platoon leader that this particular room is now "CLEAR."

Through interviews with instructors and domain experts we learned that soldiers need to make rapid assessments and decisions from the start to the end of the ECR operations. Proficiency in the ECR operations also requires superior muscle memory for quick effective responses to rapidly evolving situations (Scales, 2013). Before entering the room, the squad leader checks position of the door hinges, and this may influence the direction of the movement of the first and the second soldier. If the door hinge is on the left (right), the first soldier turns to the right (left) side. The second soldier goes in the direction opposite to the first. Soldiers also need to make rapid decisions on whether a person in the room is a combatant or not. A number of factors, e.g., possession of weapon and whether the person kneels when commanded, influence such decisions. Many of these actions and decisions are done within a short time window, often less than a minute to complete.

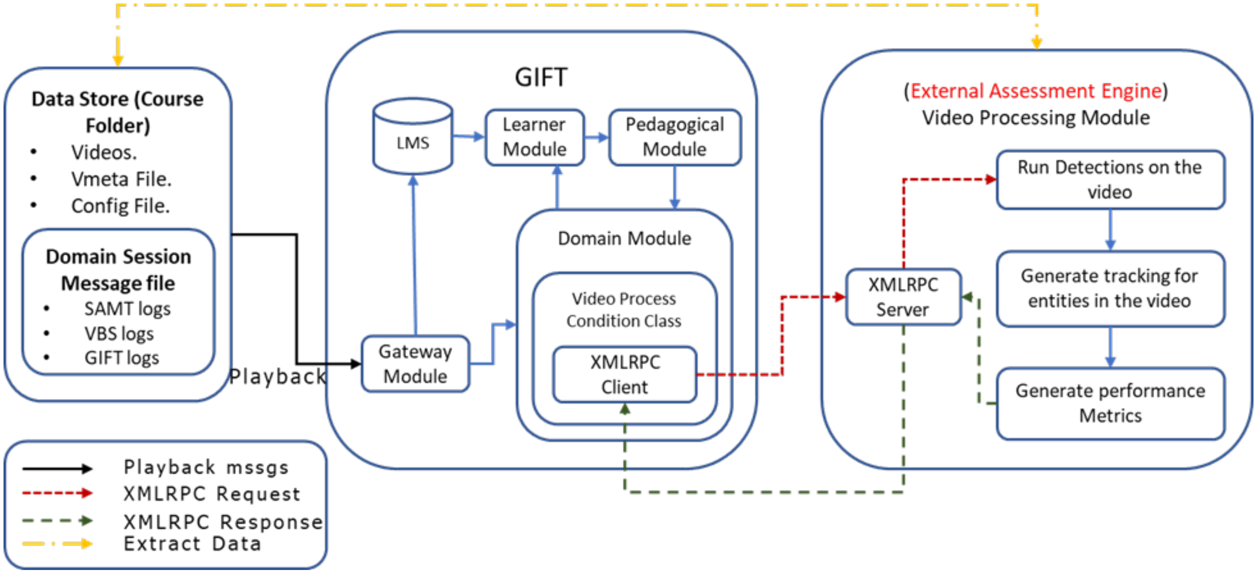


Figure 3: General architecture for integration of the external assessment engine with GIFT.

All of this information/context has to be derived from analyzing data collected from an overhead video camera that captures all of the soldiers' activities, a VBS3 video that captures the evolving ECR scenario, and data on soldier marksmanship that is provided by the SAM-T logs collected in the system. The next section discusses our design and implementation of an External Assessment Engine (EAE) that works in conjunction with the GIFT system to collect, integrate, and analyze the different sources of data to derive individual soldier and collective team performance. The following section then discusses a number of performance metrics related to soldiers' psychomotor, cognition, decision making, and team skills for ECR operations.

The External Assessment Engine (EAE) and Integration with GIFT

Figure 3 shows the overall computational architecture of the EAE and its integration into GIFT (Sottilare, Brawner, Goldberg, & Holden, 2012). Our external assessment engine, often referred to as the *video processing module* adopts a server communication model for its interactions with the GIFT system. To facilitate this, we have added a condition class in the GIFT domain module, which establishes communication between the EAE and GIFT, and serves as the entry point into the GIFT architecture. The current integration architecture focuses on post data analysis (i.e., following completion of a training event), but, in the future, we may modify it to support online analysis and assessments. The different components of our architecture, i.e., the *Data Collector*, the *Communication Module* are explained in greater in the rest of this section. Our approach to video data analysis and the derivation of performance metrics is discussed in Section 6.

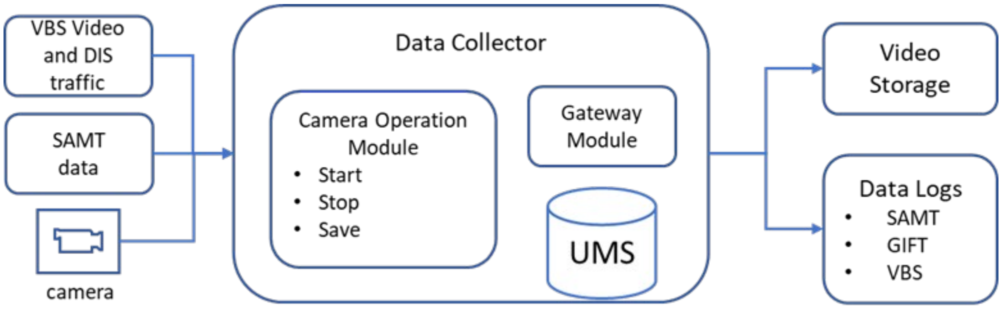


Figure 4: General architecture of the GIFT data collector.

Data Collector

Soldier activity, the ECR scenario data, and the SAM-T logs are collected at the SAM-T testing site prior to analysis. As discussed, this data comes from different sources: (1) a camera placed on top of the SAM-T screen facing the soldiers; (2) VBS3 video and events derived from the ECR scenario as it plays out in the environment in the form of DIS; and (3) SAM-T behavior and shot event data captured across wearable and weapon-embedded sensors.

All of the data is initially collected in the form of "Domain session messages" through GIFT interfaces. The data collector is currently being used to log data at the test environment. The current implementation of the Data Collector is implemented as a subset of GIFT, and it incorporates the Gateway module and the User Management System (UMS). This architecture is shown in Figure 4. The set up in GIFT does not follow the traditional approach of executing a Domain Module course or assessment and does not have an associated Pedagogy Module that determines instructional strategies based on changes in leaner state. Therefore, as discussed, there is no real time assessment or intervention associated with the data collection process. The Data Collector captures the soldier facing and VBS3 videos and stores them in a video storage location. The DIS messages from the SAM-T input stream are captured and transformed into a pre-defined logging format and stored at a location.

In more detail, the Gateway module is an interface between an external training environment (like VBS or Unity) and GIFT. The purpose of our gateway module is to establish a connection and send or receive messages to and from a training environment. In our work, the received messages are filtered, and then converted into a GIFT format and sent to the other modules in GIFT for further processing based on their type. One use case we have implemented for the Gateway module processing is VBS DIS messages. VBS DIS messages are filtered, decoded, processed by the relevant GIFT module, and actionable information generated by the module may be converted back to the DIS protocol and fed back to the VBS system.

The two management systems in GIFT are the Learner Management System (LMS) and the User Management System (UMS). The UMS module is designed to maintain a user session whereas the LMS module is responsible for maintaining scoring records based on the user's performance on a training application. The UMS stores user information like their biographical details and manages these processes (e.g., what training systems has the user participated in). In addition to this, the UMS is also responsible for logging all the messages and maintaining domain sessions for a system.

Communication and Analysis

The communication between GIFT and the EAE follows a server communication model using XML- RPC as an interface. Figure 5 shows a flowchart of the complete integration and analysis logic, with the two major processes, GIFT and the EAE, shown on the left and right sides of the figure, respectively. Logical steps are marked as solid arrows and communication interfaces are marked as dashed arrows. All communications between GIFT and the EAE take place over the XML-RPC interface, except for the server initialization, which happens through a command line interface.

At startup, the condition class for our EAE initializes the XML-RPC server connection to the EAE, which is implemented as a Python engine. Next, the EAE server is initialized using a command line interface, which results in GIFT sending a packet containing the location of the course folder on the file system and a dump of all of the data collected by the SAM-T and VBS systems. Ordinarily the SAM-T/VBS data would be communicated to condition classes during playback, but in our system the EAE makes inferences using data from the entire time period of data collection. This may include one or more training sessions, and it enables the metrics computed by the EAE to reason not only about performance corresponding to the current activities, but really look at a sequence of activities and events that evolve in space and time.

Analyzing behaviors over space and time, allows us to make inferences about soldiers' activity sequences as they evolve, providing mechanisms for us to map these sequences to their goal-oriented behaviors and complex reasoning processes. Therefore, having all of the data, which is collected in a streaming format, makes our complex processing task a little easier. Once GIFT sends the data packet over the XML-RPC interface, it enters a paused state while it awaits a start signal from the EAE.

When the data packet is received by the EAE, it applies machine learning-based processing algorithms to derive the metrics described in section 5. The generated metrics are entered into a queue data structure with a timestamp that represents the time in the evolving scenario at which the performance metric was generated. When all of the data packet has been processed, the EAE generates a start signal for GIFT, and the contents of the queue are written back to the GIFT condition class using a playback mode. The performance metrics are written back in a sequential manner, and the EAE maintains a timer which is initialized when GIFT initiates the playback mode. When the timer value matches the timestamp of the next evaluated performance metric in the queue, the condition class is updated with the associated values. This is repeated till the metric queue is empty, the EAE shuts down the XML-RPC server.

Performance Metrics

Our EAE is designed to support performance monitoring using a comprehensive set of metrics for individual and team performance. Through discussions with subject domain experts and review of related literature, we have developed five high level categories of skills on which our metrics are based. These are outlined below.

- **Individual Psychomotor Skills.** These skills refer to the speed and accuracy of individual soldier movements and are primarily acquired through repeated practice and muscle memory.

- **Marksmanship.** These skills refer to the speed and accuracy with which soldiers identify and neutralize enemy combatants while maintaining sectors of fire.

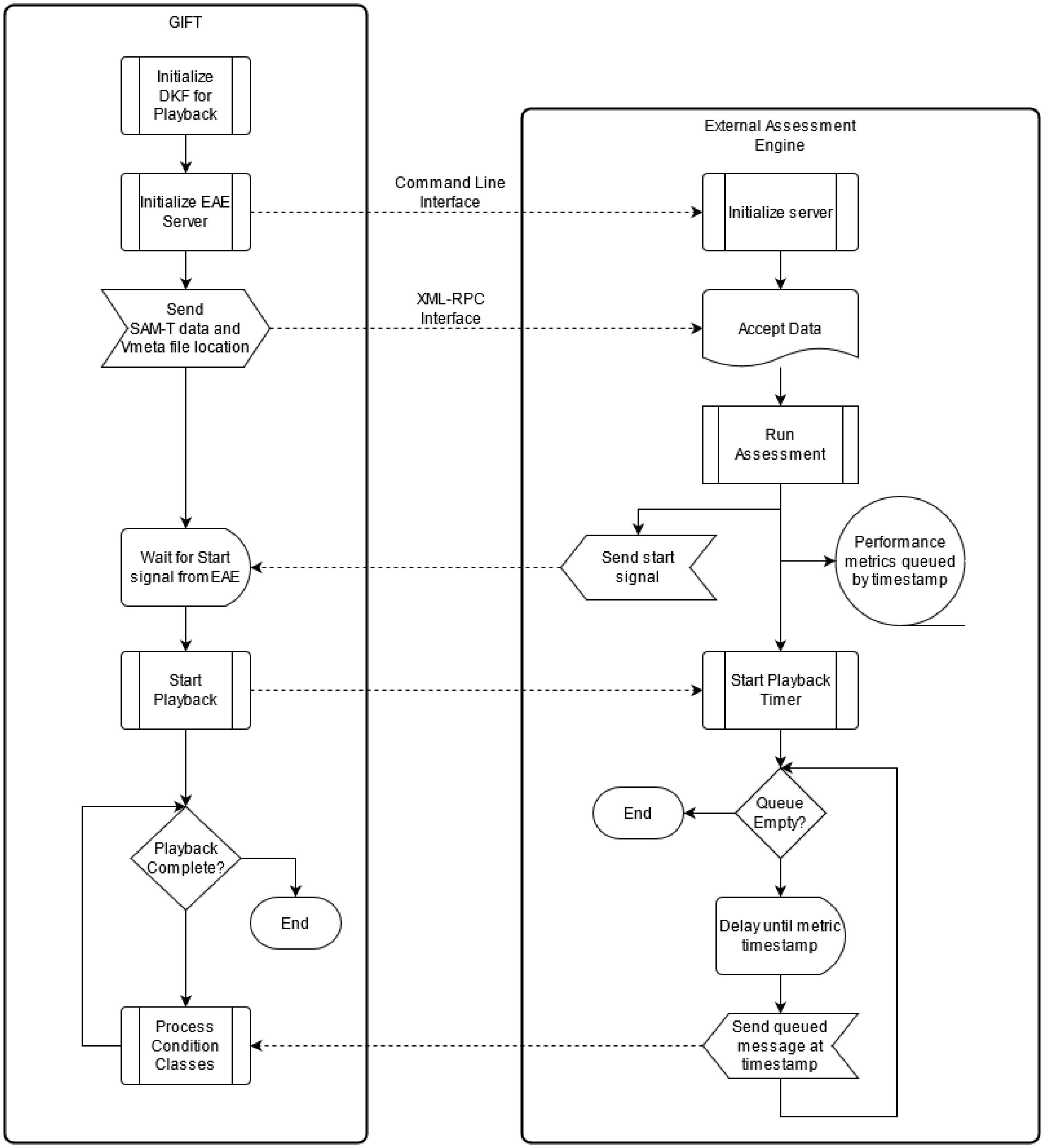

GIFT
Initialize DKF for Playback
Initialize EAE Server
Send SAM-T data and Vmeta file location
Wait for Start signal from EAE
Start Playback
Playback Complete?
End
Process Condition Classes
Command Line Interface
XML-RPC Interface
External Assessment Engine
Initialize server
Accept Data
Run Assessment
Send start signal
Performance metrics queued by timestamp
Start Playback Timer
Queue Empty?
End
Delay until metric timestamp
Send queued message at timestamp

Figure 5: Flowchart of the server communication logic connecting the external assessment engine to GIFT in playback mode. Solid arrows represent logical steps and dashed arrows represent communication interfaces.

Table 1: Performance metrics targeted for computation by the external assessment engine

Metric Name	Description	Data Inputs	Evaluated Skills
Standing At Position	Soldiers standing at their designated position prior to entering	Overhead Camera Predetermined Positions	Team Coordination Pyschomotor
Enter Signal	Soldiers signal before beginning entry	Overhead Camera	Team Coordination
React to Change	Soldiers adequately assess situation when entering and adapt planning to changing circumstances	Overhead Camera VBS Scenario Video DIS data	Situational Awareness Rapid Decision Making Team Coordination
Entrance Delay	Amount which soldiers delay at the doorway when entering the room	Overhead Camera Predetermined Entry Order	Pyschomotor
Find and Engage	Speed and accuracy with which soldiers lo- cate nearby enemies after entering	VBS Scenario Video DIS Data GIFT Entity Messages SAMT Data	Situational Awareness Rapid Decision Making Marksmanship
Path Deviation	Amount which soldiers deviate from expected trajectory during the scenario	Overhead Camera Expected Paths	Pyschomotor
Eliminate Nearest Enemy	Speed and accuracy with which soldiers tar- get nearest enemies throughout the scenario	Overhead Camera VBS Video DIS Data SAMT Data	Situational Awareness Marksmanship
Reach destination	Speed and accuracy with which soldiers clear and reach their designated stopping positions	Overhead Camera VBS Video DIS Data Predetermined Positions	Psychomotor Team Coordination Marksmanship
Hold Position	Soldiers ability to maintain their stopping position once reached until the all clear signal	Overhead Camera Predetermined Positions	Situational Awareness Rapid Decision Making Marksmanship
Covering Sector	Accuracy with which soldiers cover the sector of the room which they were assigned	Overhead Camera SAMT Data Predetermined Sectors	Situational Awareness Marksmanship
Civilian Recognition	Speed and accuracy with which soldiers distinguish civilians from enemies and disengage	VBS Video DIS Data SAMT Data	Situational Awareness Rapid Decision Making
Clear Signal	Soldiers communicate the all clear signal when room has been neutralized	Overhead Camera	Team Coordination

- **Situational Awareness.** These skills refer to the soldier's ability to maintain 360-degree, security at all times, and require cognitive processing and interpretation. Soldiers must be aware of the locations and intent of their team members, enemies, and civilians, as well as an understanding of the room space with its affordances and obstacles.
- **Rapid Decision Making.** These skills refer to the soldier's ability to quickly adapt to evolving scenarios within the room based on their situational awareness. Soldiers should be able to quickly distinguish enemies from civilians, adapt their expected movements based on room obstacles while maintaining sectors of fire, and perform actions within and across their allocated sectors.
- **Team Coordination.** These skills refer to soldier's awareness of their team members and their intent. Cooperation and coordination between team members may be initiated using gestures and predefined signals for information sharing and conveying intent. Additional coordination metrics can include providing cover for team members in different situations and avoiding fratricide.

Each of these metric categories requires combining data from the different data sources. The SAM- T environment provides much of the required data, including aiming and firing events from soldiers' guns, as well as position, orientation, firing events, and on-screen video from the VBS3 scenario which is displayed to the soldiers in the SAM-T environment. Other performance data related to the soldiers' positions and movements in the physical scenario are collected using an overhead camera. We have developed visual motion tracking methods that can analyze the video data to derive the position and movements of the soldiers in the physical space. In addition, we apply gesture recognition to this video data to derive information about soldier communication.

Our motion tracking algorithm is discussed in Section 6. The twelve metrics we have defined to assess soldier and team performance in the ECR domain, along with their required data inputs and high level skill categories, are summarized in Table 1.

Extracting Soldiers' Motion Data

The SAM-T environment, as previously discussed, is a mixed-reality synthetic training environment that provides a variety of data related to soldier performance during training. However, SAM-T, does not capture the physical position and movement of the soldiers over time within the training space as the training scenario evolves. To accomplish this, we employ a visual motion tracking algorithm that processes the video collected from the overhead camera facing the soldiers and mounted above the SAM-T display screens.

Visual Motion Tracking

Visual motion tracking is a sub-field of computer vision, which uses algorithms to localize one or more objects consistently across multiple frames of video based only on the visual (image) data encoded in each frame. In the context of analyzing our ECR scenarios, we specifically employ a multiple object tracking (MOT) scheme to track the movement of each of the soldiers during each run of the training environment. The most successful paradigm for MOT is *tracking-by-detection*, which combines powerful object detection algorithms for localization of the objects in each frame, with heuristics and other specially designed algorithms for matching the object detections between consecutive frames (Fiaz, Mahmood, Javed, & Jung, 2019). To implement the tracking algorithm in our EAE, we have developed the **Fusion-SORT** algorithm, which is a MOT algorithm following the tracking-by-detection paradigm. The algorithm is a modified version of the highly successful "Simple Online and Realtime Tracking" (SORT) MOT algorithm (Bewley, Ge, Ott, Ramos, & Upcroft, 2016). The steps of the basic Fusion-SORT algorithm is described below and depicted in figure 6.

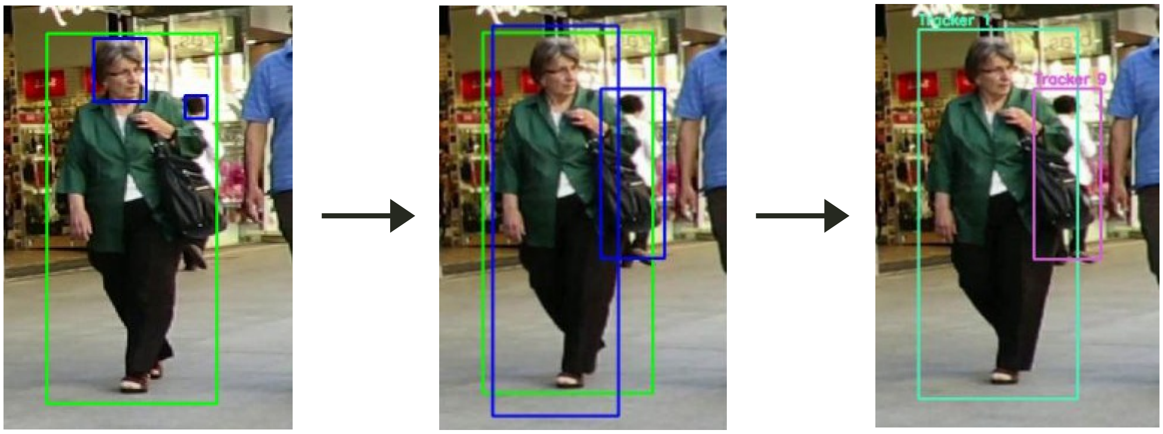


Figure 6: Example of the general tracking workflow. The leftmost image depicts detections where occlusion causes a body detection (green bounding boxes) to be missed while the head detection (blue bounding boxes) is preserved, the middle image depicts the conversion of head detections to approximate body detections (blue bounding boxes) compared to the full body detections (green bounding boxes), and the right image depicts the trackers resulting from the fusion of the detections.

1. In frame 1, run an object detection algorithm to generate bounding boxes for each tracked person.
2. For each generated bounding box, initialize a tracklet which contains a 7-dimensional linear motion model Kalman filter.
3. Move to the next frame.
4. For each tracklet, forward predict the position using the Kalman filter.
5. Run the object detection algorithm to generate bounding boxes for full-body and head-only detections.
6. Construct a cost matrix for matching full-body detections to tracklet predictions with cost elements defined by 1-IOU, where IOU is the intersection over union between the detection and the tracklet prediction.
7. Create a minimum cost matching of full-body detections to tracklets using the cost matrix, with a maximum cost threshold of 0.7.
8. For each tracklet with a matched detection, update the Kalman filter with the bounding box of the matched detection.
9. For any full-body detections which were not matched, initialize a new tracklet with the bounding box of the detection.
10. For each of the head-only detections, compute an approximate full-body detection based on the body approximation algorithm (discussed below).
11. Repeat steps 6-9, replacing the full-body detections with the approximate detections from step 10.
12. Return all tracklets which meet the probation condition (discussed below).
13. Repeat steps 3-12 until the end of the video is reached.

The Fusion-SORT algorithm roughly breaks down into two phases: *initialization* (steps 1-2) and *tracking* (steps 3-13). In addition, the tracking phase breaks down into four phases: *detection* (steps 4-5), *full-body updates* (steps 6-9), *head updates* (steps 10-11), and *return* (steps 12-13). Next, we briefly discuss each of these phases in more depth.

The initialization phase, run only on frame 1, is responsible for setting up tracklets for any persons that are visible at the start of the video. In the general Fusion-SORT algorithm, the object detection algorithm is non-deterministic; that is, it does not matter what object detection algorithm is used, so long as it can produce bounding boxes for full-bodies and for heads-only. In practice, we use the YoloV5L object detector (Jocher et al., 2020; Redmon, Divvala, Girshick, & Farhadi, 2016) fine-tuned on the CrowdHuman dataset (Shao et al., 2018) for two classes: person and head. In step 1, we run this custom trained Yolo model on the first frame of video to produce the detections.

In step 2, we use these detections to create tracklets, which keep a history of the localization of each subject, as well as a linear motion Kalman filter model to predict the next location. The Kalman filter uses a 7-dimensional state space model, $[x\ y\ s\ r\ \dot{x}\ \dot{y}\ \dot{s}]^T$, where (x, y) is the center coordinate of the person's bounding box, s is the bounding box scale (area), r is the bounding box aspect ratio, and the dotted version of these symbols represents their first derivative (i.e., velocity) as estimated by the filter. All the measurements of the filter are with respect to a two-dimensional pixel coordinate frame.

After initialization, we move to the tracking phase, which repeats on each frame of video after the first until the end of the video is reached. The detection section of tracking is straightforward: the same detection model used in the initialization phase is run on the current frame to produce full- body detections and head-only detections. In addition, we predict each existing tracklet forward in time using the Kalman filter during this phase.

For the full-body update phase, we take the full-body detections produced from the detection phase and construct a cost matrix, which compares the detections to the Kalman predictions from existing tracklets. Each element of the cost matrix compares the overlap of the bounding box from a detection and the existing tracklet prediction by calculating $1-IOU$, where IOU is the intersection over union of the bounding box and tracklet. In this way, the higher the overlap of the prediction and the detection, the lower the cost. Once the cost matrix is constructed, we use the Hungarian algorithm to make a minimum cost matching between the tracklets and detections. However, we gate the matching so that if two entries have higher than 0.7 cost (or 0.3 IOU equivalently), they are considered a false positive and the match is discarded. For all the remaining matched objects, the Kalman filters of the tracklets are updated with the bounding box of the matched detection. The unmatched detections are then considered candidates for new tracklet initialization. For each of these candidate unmatched detections, we measure the IOU with every existing tracklet and take the maximum. If the maximum IOU for a candidate is greater than 0.7, it is considered a false positive and is removed as a candidate. This additional gating of candidate unmatched detections helps to account for false positive and duplicate detections. Finally, new tracklets are created for the remaining candidate unmatched detections.

The head update phase follows analogously to the full-body update phase, but instead of using the raw detections, we first convert each head detection to an approximate full-body detection. For a head detection given by its center point, width, and height, a full-body approximation is calculated by:

$$x_{body} = x_{head} + C_1 \cdot w_{head}$$

$$y_{body} = y_{head} + C_2 \cdot h_{head}$$

$$w_{body} = C_3 \cdot x_{head}$$

$$h_{body} = C_4 \cdot y_{head}$$

Figure 7: Example of line segments drawn to represent the boundaries of the scenario space during configuration of the system.

The default constants for the approximation are: $C_1 = 0$, $C_2 = 2.6$, $C_3 = 2.6$, $C_4 = 6.6$. These equations and constants were computed empirically from the CrowdHuman dataset. Once the head-only detections are converted to approximate full-body detections, the update process follows as in the full-body update phase, with only one change. When updating with the approximate body detections, we modify the Kalman filter so that the covariance of observation noise is 3 times greater than it was for the full-body detections. In this way, we are giving more weight to the true detections in the final Kalman state (and thus the tracklet), than the lossy body approximations, which can be subject to noise. The process of converting the head detections into approximate body detections is shown in Figure 6. The leftmost image depicts the full-body detections in green and the head detections in blue. The middle image shows the approximated body detections (blue) compared to the full-body detections. The rightmost image shows the final tracklet state after updating with both the full-body and approximate-body detections.

Finally, the return section of the tracking phase is responsible for returning the valid tracklets after the processing for the frame is complete. This is fairly straightforward, as tracklets are simply returned as a sequence of bounding boxes for the subject. However, we also implement a probation condition for tracklets to help gate any false positives. In Fusion-SORT, tracklet must receive either 6 consecutive frames of updates from any source, or a combination of at least one update from a full-body detection and one update from a head detection before it is accepted. The idea is that if we detect both a head and body, it is much less likely to be a false positive than if we only detect one or the other. Essentially, the probation condition exploits the semantic consistency of having both head and body detections to reduce the false-positive rate.

Planar Projection

The visual motion tracking described above provides a robust method for measuring the position of the soldiers in the training environment relative to the camera's viewpoint. We refer to this position data as the *camera frame*. However, the overhead camera frame captures only a static 2-dimensional representation of soldier position, while the soldiers are moving in a 3-dimensional space. In order to calculate metrics that are representative of this, we must convert positions from the camera frame to the *map frame*, which represents a 2-dimensional bird's eye view of the scenario space. To convert to the map frame, we utilize a 4-point planar homography based on the bounds of the scenario space. During the setup of the SAM-T system, users create a configuration file for our EAE using a provided authoring tool. The tool allows the user to draw line segments along the wall or other boundaries of the scenario space, as shown on simulated data in Figure 7. Internally, the system extrapolates the intersection of these line segments to determine the corners of the scenario space. These four corners are used to setup the planar homography with the generated 2D map frame. Once the projection is defined using the room boundaries, the position of the soldiers in the map frame can be approximated by projecting the camera frame positions using the calculated homography. An example of this projection on simulated data is shown in Figure 8, where the left image represents tracking from the camera frame viewpoint and the right image represents the projection onto the generated map frame with trails to indicate tracking history.

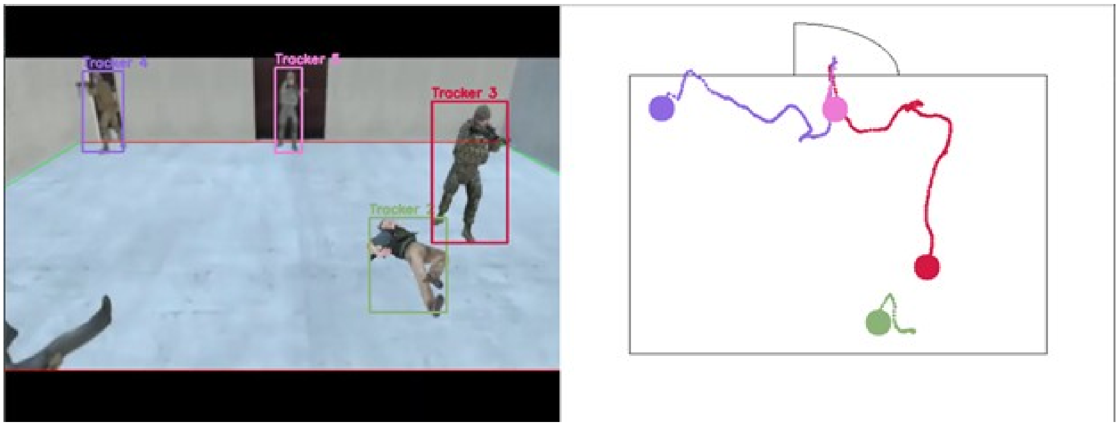


Figure 8: Example of the planar projection from the camera frame (left) to the generated top-down map frame (right).

Multi-Modal Data Fusion

The soldiers' position and motion data derived using our motion tracking approach discussed in the last section, is aligned and fused with the log data obtained from the SAM-T system to derive soldiers individual and team performance metrics. These performance metrics are derived from a multilevel task model that we have generated using cognitive task analysis (Schraagen, Chipman, & Shalin, 2000) and consultations with our domain experts. Multi-level task models have been applied successfully to a variety of intelligent learning environments, including some based in the GIFT architecture (Biswas et al., 2020; Kinnebrew, Segedy, & Biswas, 2017). At the lowest level of this task model hierarchy, we capture directly observable events in the training environment, which map on to cognitive and psychomotor skills (e.g., firing a gun), and as we move up in the hierarchy, each subsequent level represents more domain general task evaluations (e.g., clearing an area of enemy combatants).

A primary challenge we face in describing task models for the ECR scenarios is that identifying low level task skills may require combining information from multiple data streams. For example, we may have to align and combine soldier motion data (from the soldier facing video camera) with their firing performance (derived from SAM-T event logs) to derive a soldier's basic skill, such as his marksmanship accuracy in his assigned zone. The described multilevel task model including all of the data fusion requirements for the ECR domain is shown in Figure 9.

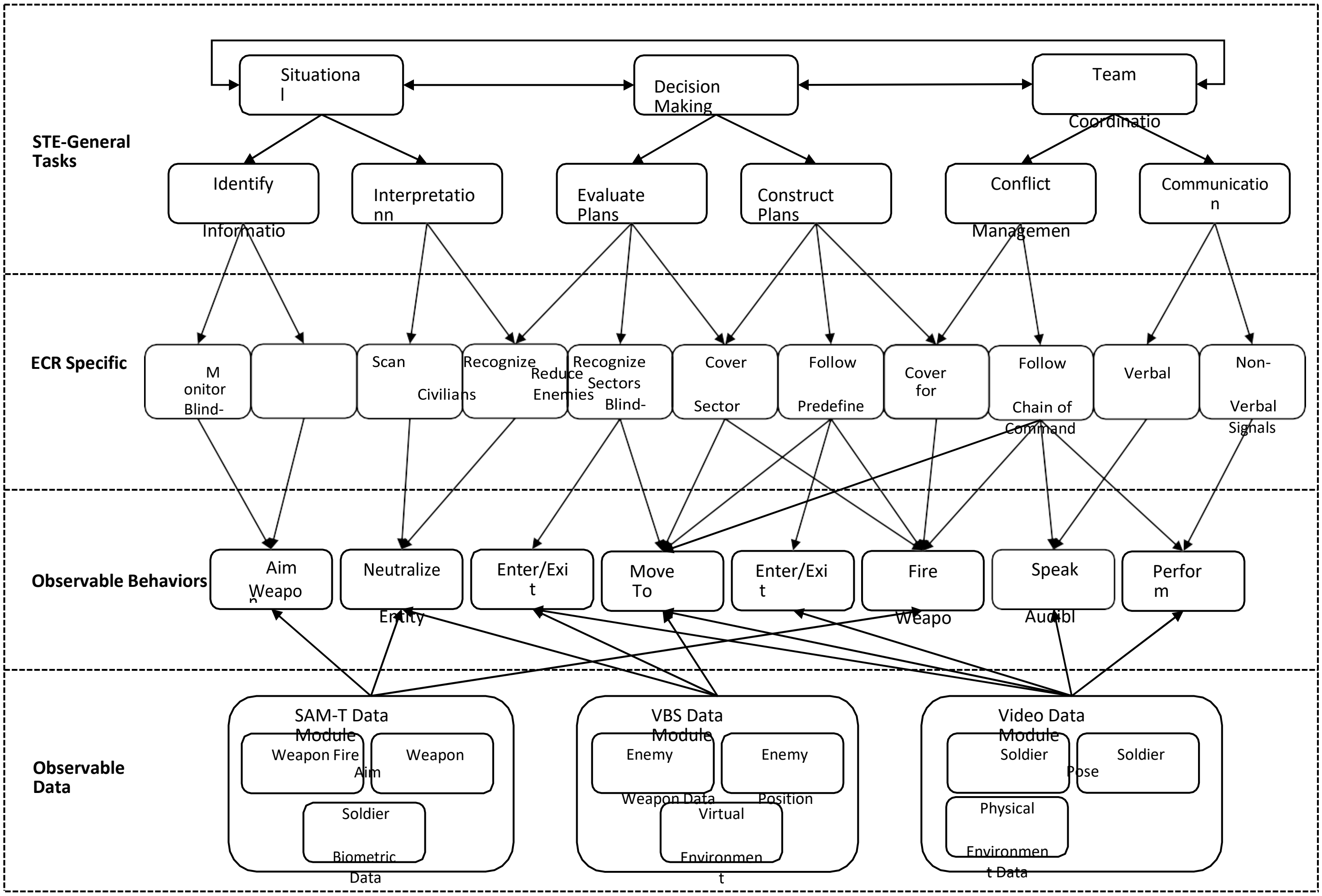


Figure 9: Multilevel task model with data fusion for the Enter and Clear Room domain

In previous applications, all of the observable low-level behaviors have been derived from a single data source, i.e., from the log files generated in the learning environment. However, our training environments for DBDs combine multi-modal data from multiple data streams. This data must be pre-processed, aligned, and aggregated before soldier skill levels can be inferred. To clearly outline our extended approach, we add a layer at the bottom of the task model, called the *observable data* layer to capture the set of data sources, their pre-processing modules, and the task-level information that can be derived from each of these sources. At this level, we include modules that capture SAM-T data events, VBS data events, and video data events.

Moving up to the next layer, which we call the *observable behaviors* layer, we capture how the information derived from the processing modules of the data layer are grouped into a common representational format. In our case, this creates a timeline of observable behavior events, such as move, aim, change posture, gesture, and fire. These observable behaviors are the lowest level tasks that soldiers complete correspond to the lowest layer of a traditional task model which does not require data fusion.

From the behavior layer, we move up to the *domain-specific skills* layer that represents the over- all tasks that soldiers must complete within the specific ECR domain. This layer most closely corresponds to the metrics described in Section 5, with many of the described metrics directly evaluating these domain-specific skills. The skills at this layer can be analyzed by processing the timeline of events from the behavior layer and aligning these timeline behaviors with the context of the evolving scenario. We examine both the data for each soldier in the squad individually to compute individual skills, as well as the squad in unison to compute team-related skills.

At the final layer, we have the *STE-General tasks*, which represent more abstract behaviors and tasks which could be common to many STE environments and domains. For example, this layer encodes concepts such as situational awareness and information gathering, decision making and plan construction, team communication and conflict management, etc. While all of these tasks are used in the ECR domain, they also generalize to other domains and environments. The tasks at this level correspond to the high-level metric categories outlined in section 5.

Timeline of Observable Behaviors

As previously discussed, one of the key problems we have to address in this domain is to fuse information from multi-modal data sources into a common representational format, to support the computation of performance metrics. As part of this fusion process, we build a timeline of observable behaviors, which combines the data from multiple sources into a common format. To build the timeline, we begin with a set of *action affordances*, which is a set of concepts representing things that the participants can do in the environment. In the context of DBDs, this would be concepts such as *move, signal, aim,* or *fire*.

Borrowing from the domain of AI planning, we follow the STRIPS action representation format, where each action is defined as a set of *parameters, preconditions,* and *effects* (Fikes & Nilsson, 1971). Parameters represent the abstract objects that are involved in the action. For example, the Move action would include three parameters: (1) the agent who is moving, (2) the starting location of the agent, and (3) the ending location (destination) of the agent. Preconditions represent the concepts, which must be true for the action to be executable. Continuing with the *Move* example, the agent must be at the starting location prior to taking the specific move action, which has that starting location as a parameter. Another example is a *CeaseFire* action, which has the precondition that the agent must currently be in a state of firing. Effects represent the concepts that change once an action is completed. For example, the *Move* action has two effects: the agent is at the destination and the agent is not at the starting location. Figure 10 shows an example of a few action affordance definitions for the ECR domain specified in PDDL.

```
( : action Move
    :parameters (? a – agent ?start– location ?end – location)
    : precondition ( and (AT ? a ?start ) )
    : effect ( and ( not (AT ? a ?start ) ) (AT ? a ? end ) )
)
( : action Aim
    :parameters (? a – agent ? currA – screenPos ?newA – screenPos )
    : precondition ( and (AIMING ? a ? currA ) )
    : effect ( and ( not (AIMING ? a ? currA ) ) (AIMING ? a ?newA) )
)
( : action Fire
    :parameters (? a – agent )
    : precondition ( and ( not ( FIRING ? a ) ) )
    : effect ( and ( FIRING ? a ) )
)
```

Figure 10: Example of PDDL action affordance definitions for the ECR domain

Given the action affordances specified in PDDL, the timeline is then created by grounding ac- tions parameters with the data collected from SAM-T and motion tracking and associating each grounded action with a timestamp. During the grounding, we discretize any continuous sensor data variables (e.g., position, aim, etc.). For example, the position data in the map frame is discretized onto a grid with sensitivity specified in the configuration file. This discretization solves issues related to the computational tractability of the analysis. Converting the raw sensor data into this STRIPS-style format allows for a common representational format of many disparate data sources. In our EAE, adding another data source only requires the authoring of one Python class to convert the data to the STRIPS-style format. This common representation allows for analysis and fusion across multiple data sources. In addition, the STRIPS format allows us to utilize the tools and techniques from AI planning for further analysis, though this is currently reserved for future work.

Example Domain-Specific Metric Calculations

In this section, we provide a few examples of how the skills at the *ECR-Specific Skills* layer, as well as their associated metrics described in section 5, are analyzed and computed.

The ECR domain can be divided into two primary phases: enter phase and clear phase. During the enter phase, soldiers are expected to rapidly move through the doorway to enter the room while scanning for enemy combatants in the room, maintaining sectors of fire, and covering for soldiers currently entering. In order to maintain these conditions, soldiers are generally required to enter the room following tracks toward room corners that are opposite of the direction of the previous soldier. For example, if the first soldier enters the room and moves toward the left wall, the second soldier should enter and move toward the right wall (see Figure 2). In our EAE, we evaluate this behavior and refer to this metric as *Entrance Vectors*. To perform the evaluation, we utilize the position data generated by the motion tracking, as well as the *door region*, representing the area around the entrance door as defined in the configuration file. We add two action affordances to the system representing when a soldier enters the door region, and when the soldier exits the door region. When evaluating the entrance vector metric, the soldiers are ordered based on the first time the enter doorway event occurs in their timeline, which represents the order the soldiers enter the room. Beginning with the first soldier and for each soldier who enters, we calculate a vector from their position at the enter doorway event to their position at the leave doorway event. By comparing the angle between the vectors for subsequent soldiers, we can determine whether each soldier entered in the opposite direction of the previous soldier. Entrance vectors is a team metric, meaning only one score is produced for the entire squad. The final score determined by the EAE for the entrance vectors metric is the percentage of soldiers who entered correctly.

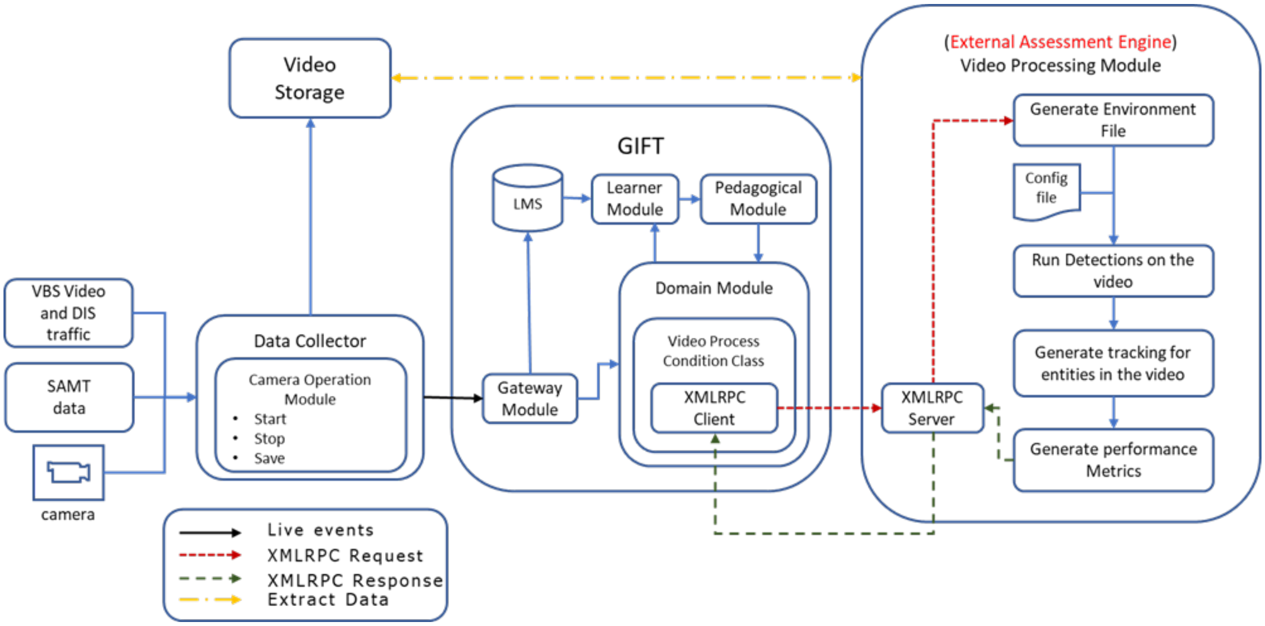


Figure 11: Proposed architecture for integration of GIFT and the external assessment engine in real time.

During the clear phase, soldiers are expected to move along the walls of the room while maintaining sectors of fire and neutralizing any enemy combatants. One element of particular importance in this phase is that soldiers maintain movement close-to and along the walls, as this minimizes the risk of enemy combatants firing in blind spots and maximizes the range of vision in the room. To evaluate this, we define the *Move Along Wall* metric. Contrary to entrance vectors, move along wall is an individual metric, meaning that a score is calculated for each soldier, rather than one for the entire team. For each soldier, we calculate the metric start time by the first appearance of the leave doorway event in the timeline and the metric end time by the last appearance of a move event in the timeline. Then, for each move event which occurs in between the start and end times, the distance from the soldier's position to the nearest wall is computed. If this distance is below a threshold, then the soldier is considered to be along the wall. The final score for each soldier for the move along wall metric is the percentage of time which the soldier is along the wall during the interval between the start and end times.

Conclusions and FutureWork

Real-time Architecture

The existing architecture focuses on a post analysis scenario where data is collected ahead of time. This architecture can be enhanced in future to work on a real time scenario where the data is collected from multiple sources and fed directly to GIFT and Video processing module. Assessment can be processed and generated at regular intervals in real time based on the events and the video recorded. Figure 11 shows a proposed architecture for a real time scenario.

Data collector will play a different role when integrating at real time with the EAE. The Data collector will collect data from multiple sources and pass data to GIFT to be interpreted and evaluated. The rest of the architecture follows the similar lines of the existing architecture except that the processing will be done in real time at set intervals rather than at the end of the entire data collection effort. Next, we describe a user data flow to show possible way to approach the real time architecture in future.

User Data Flow

The Data collector will make sure the camera is recording and data is being sent to GIFT along with SAM-T sensor data and VBS3 video and DIS data. The gateway module acts as an interface that collects data from the data collector and forwards it to the domain module. The domain module goes through the Domain knowledge file and triggers appropriate condition classes. Assuming the condition class for EAE is triggered at this point, the condition class will pass the execution to the video processing module assessment engine. The assessment engine will collect video downloaded from the camera, VBS scenario video data, SAMT sensor data and DIS packet data to run it through the 'process video' module. The process video module will generate a 2D map which can be used for after action review (AAR) and apply the motion tracking algorithm on the camera video. The output from the 'process video' module will be sent to the 'assess video' module which will align data from the camera feed and the VBS scenario video. The 'generate performance metrics' module will take the aligned data along with SAMT sensor data and DIS packet information to generate Assessment (Above/Below/At), Competence [0, 1], Confidence [0, 1] and Trend [-1, 1] metrics to be sent back to the domain module. The domain module will send the data to the pedagogical module through the learner module which will decide the appropriate strategy to be applied for feedback. This user data flow architecture is shown in Figure 12.

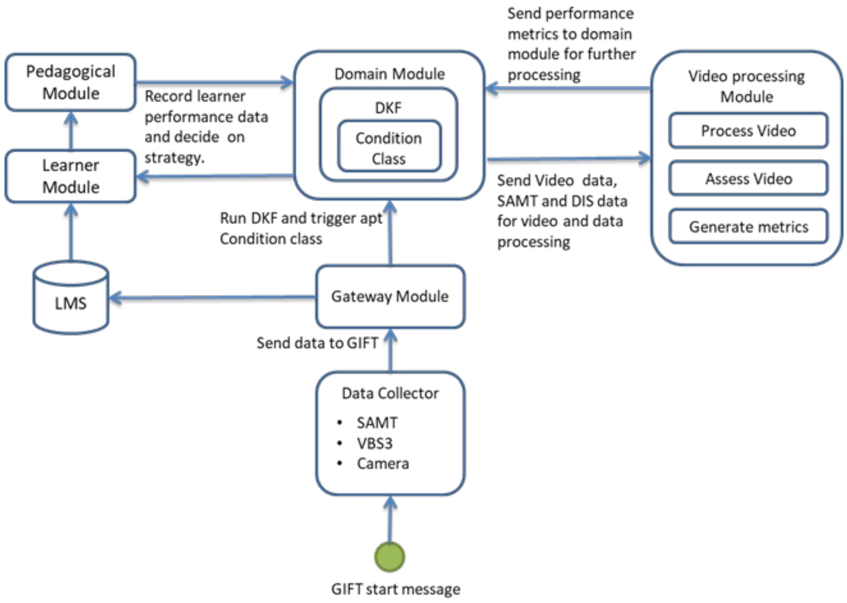

Send performance metrics to domain module for further processing
Pedagogical Module
Record learner performance data and decide on strategy.
Domain Module
DKF
Condition Class
Video processing Module
Process Video
Assess Video
Generate metrics
Learner Module
Send Video data, SAMT and DIS data for video and data processing
Run DKF and trigger apt Condition class
LMS
Gateway Module
Send data to GIFT
Data Collector
• SAMT
• VBS3
• Camera
GIFT start message

Figure 12: Architecture for the user data flow in the proposed real-time analysis scenario.

Domain Generality

The existing architecture presented in this paper focuses on analysis of only dismounted battle drills, and more specifically on the enter and clear a room exercise. However, we believe this architecture of fusion of multi-modal data using an external assessment engine in conjunction with GIFT can generalize to other DBDs and even to mounted battle drills. Future work will focus on this generalization for application to a variety of scenarios.

The task model presented in the previous sections provides the basis for the generalized architecture. At the bottom observable data layer, we maintain a series of modules to collect data from various sources. These modules are designed to be swapped in and out of the EAE dependent on which data sources are available for the given scenario. Above this, at the observable behavior layer, the architecture remains largely the same as well. This layer serves to combine each of the data modules in the lower layer into a common representational format. Specifically, we utilize a STRIPS-style action representation and a timeline of these defined actions across the course of the scenario. At the top of the task model, the architecture remains largely the same as well. The STE-General tasks layer is designed such that many of the tasks are relevant for any STE domain. For example, any team-based task will require analysis of the communication and conflict-management tasks. Because of the generality of this layer, it would require little modification for any STE domain.

Below this, at the domain-specific skills layer, the primary changes for domain generality of the system take place. At this layer, we still represent a variety of domain-specific skills, but instead of computing each skill as we do in the current system, in the proposed architecture, each skill can be enabled and disabled depending on the context of the scenario being analyzed. At a low level, these skills are enabled or disabled depending on what scenario is being run. For example, metrics related to vehicle operation would be disabled for DBDs but enabled for mounted battle drills. At a higher level, skills at this level of the model can be enabled or disabled dynamically, depending on the context of the specific scenario being run. For example in the ECR domain, a scenario instance where soldiers quickly neutralize enemy combatants may not need skills related to moving to cover, while a scenario that has enemies firing at squad members continuously might require these skills. At the time of analysis in the proposed architecture, the specific scenario would be analyzed to determine which tasks are required based on the evolution of scenario and the beliefs and intentions of the scenario participants. In this way, the proposed architecture will be far more adaptable to time-evolving scenarios in a variety of domains.

Acknowledgements

This research and development work has been partially supported by funding from US ARL grant # W911NF1820081 .

References

Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple online and realtime tracking. *2016 IEEE International Conference on Image Processing (ICIP)*, 3464-3468.

Biswas, G., Rajendran, R., Mohammed, N., Goldberg, B. S., Sottilare, R. A., Brawner, K., & Hoffman, M. (2020). Multilevel learner modeling in training environments for com- plex decision making. *IEEE Transactions on Learning Technologies*, *13* (1), 172-185. doi: 10.1109/TLT.2019.2923352

Fiaz, M., Mahmood, A., Javed, S., & Jung, S. K. (2019). Handcrafted and deep trackers: Recent visual object tracking approaches and trends. *ACM Computing Surveys (CSUR)*, *52* (2), 1–44.

Fikes, R. E., & Nilsson, N. J. (1971). Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence, 2*(3), 189-208.

Gant, T., Speidel, J., Tatum, D., & Zuelke, E. (2019). Squad advanced marksmanship trainer., 234-239.

Holmquist, J. P., & Goldberg, S. (2007). Dynamic situations: The soldier's situation awareness. Jocher, G., Stoken, A., Borovec, J., ChristopherSTAN, Changyu, L., Hogan, A., . . . yzchen (2020, July). *ultralytics/yolov5: v2.0.* Zenodo. Retrieved from https://github.com/ultralytics/ yolov5 doi: 10.5281/zenodo.3958273

Kinnebrew, J. S., Segedy, J. R., & Biswas, G. (2017, apr). Integrating model-driven and data- driven techniques for analyzing learning behaviors in open-ended learning environments. *IEEE Transactions on Learning Technologies*, *10*(02), 140-153. doi: 10.1109/TLT.2015.2513387

Pargett, M. (2019, March 25). *Squad tactics tested on new virtual marksmanship trainer.* Retrieved April 2, 2019, from https://www.army.mil/article/219231/squad tactics tested on newvirtualmarksmanshiptrainer _

Redmon, J., Divvala, S., Girshick, R. B., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788.

Scales, B. (2013). Virtual immersion training: bloodless battles for small-unit readiness. *The Magazine of the Association of the United States Army*, 24–27.

Schraagen, J. M., Chipman, S. F., & Shalin, V. L. (2000). *Cognitive task analysis*. Psychology Press.

Shao, S., Zhao, Z., Li, B., Xiao, T., Yu, G., Zhang, X., & Sun, J. (2018). Crowdhuman: A benchmark for detecting human in a crowd. *ArXiv*, *abs/1805.00123*.

Sinatra, A., Kim, J. W., Johnston, J., & Sottilare, R. (2018). Assessment of team performance in psychomotor domains. In *Design recommendations for intelligent tutoring systems: Volume 6.* US Army Research Laboratory.

Sottilare, R., Brawner, K., Goldberg, B. S., & Holden, H. (2012). *The generalized intelligent framework for tutoring (gift).* Retrieved from https://gifttutoring.org/projects/gift/ wiki/Overview. U.S. Department of the Army. (2011, June 10). *Army tactics, techniques and procedures – attp 3-06.11.*

ABOUT THE AUTHORs

***Gautam Biswas** is a Cornelius Vanderbilt Professor of Engineering, and a Professor of Computer science and Computer engineering in the EECS Department, and a senior research scientist at the Institute for Software Integrated Systems, Vanderbilt University. He conducts research in intelligent systems with primary interests in modeling and simulation, analysis of complex embedded systems, data mining, and computer-based learning environments (CBLEs) for STEM disciplines. He has also developed innovative educational data mining techniques for studying students learning behaviors and linking them to metacognitive strategies. His research has been supported by funding from ARL, NASA, NSF, DARPA, and the US Department of Education.He is a fellow of the IEEE and PHM societies.*

***Naveeduddin Mohammed** received the BE degree in Information technology from Osmania University and the MS degree in computer and information sciences from University of Colorado, Denver. He is currently working as a research engineer at the Institute for Software Integrated Systems, Vanderbilt University. His work focuses on designing, developing and maintaining frameworks for open-ended computer-based learning environments.*

***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) in Orlando, FL. He has been conducting research in the Modeling & Simulation community for the past nine years with a focus on adaptive learning in simulation-based environments and how to leverage Artificial Intelligence tools and methods to create personalized learning experiences. Currently, he is the LITE Lab's lead scientist on instructional management research within adaptive training environments and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT). Dr. Goldberg is a Ph.D. graduate from the University of Central Florida in the program of Modeling & Simulation.*

FLATT: A new real-time assessment engine powered by GIFT

Steven Harrison & Elyse Burmester
Dignitas Technologies

Introduction

Generalized Intelligent Framework for Tutoring (GIFT) (Overview - GIFT - GIFT Portal, 2021) is a complex system that is able to apply adaptations to a virtual training environment (VTE), but determining when to adapt and learning how to author via GIFT can be time consuming and may not always be intuitive. Trainers and researchers (operators) require a much simpler interface to capture and visualize information for learner states, and then act on that data to adapt the training in real-time. The Flexible and Live Adaptive Training Tool (FLATT) is a standalone web application that allows the operator to rapidly author rules before and during a live session that specify *when* and *how* to adapt a VTE.

While authoring can be done independently, the FLATT application will depend on an intelligent tutoring service (ITS) and VTE during a live session. FLATT is designed as a support tool to help adapt the VTE in real-time to collect useful metrics which will help in identifying and improving which adaptations are most effective to the trainee and to increase the effectiveness of training solutions. The FLATT application is ITS and VTE agnostic, so they can be replaced later if a better solution becomes available as long as they support the FLATT API.

RELEVANCY to GIFT

FLATT builds upon the benefits and advantages provided by GIFT as a third-party assessment engine for training and research applications. One of GIFT's primary advantages is that it can gather a large amount of data concurrently from various internal and external sources, bring them together, analyze all of it, and establish learner state information across cognitive, affective and performance. FLATT is able to tap into this wealth of information by communicating with GIFT through its domain knowledge file's (DKF) FLATT condition. The condition allows for bidirectional communication to external applications; this provides access points for external data streams to pass information to GIFT, but also allows GIFT to act as a service, sharing its capabilities and features. During a live session, GIFT can share this real-time data to FLATT, which is then able to determine if the data fulfills the authored rules' criteria. If so, the associated trainee state-driven customizations (TSCs) will be applied to the VTE, adapting it automatically.

FLATT provides a much simpler and straightforward interface to the user with step-by-step inputs to create rules based on raw trainee and/or environment data points, rather than relying on task or concept restrictions that are required for traditional GIFT DKF authoring. These rules are configured in a robust if/then format to account for any number of conditions and to create a unique situation in which the adaptations will be applied. The interface is designed to be intuitive enough that creating the rules should be as easy as if the operator was saying a sentence. "If A and B happens, but not C, then adapt the VTE by performing customizations 1 and 2". FLATT is also more flexible than GIFT in specifying when the adaptation should occur. By giving operators the ability to build their own custom "logic" for when the VTE customization should occur, they can tailor their adaptations for specific use cases or for general purposes. FLATT is also able to create multiple "logic flows" within a single rule to execute the same customization. These divergences can be at a low level such as checking two separate values for the same condition (e.g., checking if the trainee is at either of two different locations) or at such a high level that there are no common components (e.g., the operator can choose to adapt the VTE when the trainee's heart rate is steady *OR* stress level is low *OR* shots missed is 0). If any of these component conditions are met, FLATT will detect that the rule has evaluated to true and the VTE adaptation(s) will automatically be executed.

Research Phase 1 Decisions

The following sections outline the domain analysis performed during Phase 1 and our decisions made based on those findings. We looked into the various components that would need to be integrated with FLATT: ITSs, Sensors, VTEs, TSCs, graphical user interfaces (GUIs), and Rule Engines. The research was performed by analyzing our own products and the products of other companies, attending product demonstrations, and conducting internet searches for existing research. We also interviewed some of the companies over the phone and at I/ITSEC to get more detailed information on their products.

ITS

ITSs are to be used for capturing and delivering information that FLATT is not directly responsible for such as learner history, survey results, sensor data, conversations and natural language processing, and artificial intelligence and machine learning decisions. In choosing this use-case ITS, we were specifically looking to see if it would be relevant to FLATT as a research tool, if they are able to integrate with any VTEs or sensors, the application's licensing, and our familiarity with the software. Although FLATT is ITS agnostic, we chose GIFT as the primary ITS for use during development.

GIFT was chosen as the primary ITS because it provides three main capabilities: an ITS for adaptive learning, a platform for sensor data collection, and a platform that supports state of the art research in ITS concepts. As an intelligent tutoring framework, GIFT is unique in that it is open source and domain independent, includes a sensor framework, and is designed to integrate with external training applications

Sensors

Sensors are critical to FLATT and knowing when to adapt the VTE by providing additional information about the learner and the training environment. Using the output of the sensors, FLATT will be able to use them to understand the trainee's current state in order to trigger appropriate TSCs. We performed our research through online searches, analyzing sensors used by Dignitas programs, and by communicating with companies that specialize in collecting sensor data.

FLATT will be able to support an any number of sensors. During development, FLATT is focusing on integrating and using the Zephyr Bioharness (Zephyr) because it is readily available and provides a number of measurable sensor inputs including heart rate, breathing rate and waveform, and posture. GIFT has also used this sensor previously, so FLATT is able to benefit from that by re-using algorithms and calculations to handle the bioharnesses' raw sensor data. The sensors integrated with FLATT in the future will be determined by accessibility, cost, and the difficulty of interpreting the raw sensor data.

VTE & TSCs

The TSCs available are highly dependent on which VTE is being used. GIFT already has a large number of supported customizations with VBS3, so FLATT is able to utilize this existing relationship to support a large number of TSCs right off the bat. Some of these customizations include add, remove, and move entity, entity damage/health, breadcrumbs (follow path/waypoint), modify environment (fog, overcast, rain, etc.), and applying a custom script.

Part of our research was investigating the effectiveness of TSCs in relation to the trainee's state. We found two great published papers that support our work and show a direct correlation between adapting the scenario and the trainee's state

Physiological Based Adaptive Training (Thomas Schnell, n.d.)

The research done in this paper demonstrate the effectiveness of using workload, performance, and subject state to change the difficulty of the scenario by performing real-time adaptations. Their tests show as the difficult increased by adapting the scenario to reduce visibility, the trainee trended toward a higher workload and cognitive effort. This is very promising for FLATT because the operator can author rules to intentionally and automatically modify a scenario to maintain a higher workload and cognitive effort for the trainee.

Informing the Long-Term Learner Model: Motivating the Adult Learner (Phase 2) (Lauren Reinerman-Jones)

The other published work used motivation and motivator preferences to adapt an ITS to best suit the trainee. Their results showed that learners that are classified as Learning Driven, possessing higher intrinsic motivation tendencies, should be provided higher levels of autonomy in how they execute a task or training. Alternatively, a learner with lower intrinsic motivation tendencies, prone to high levels of stress and low competency, would require more frequent positive feedback, more opportunities to feel success, longer period of guidance, etc. Knowing the motivators for a learner will help direct the type of training environment that will be best suited for that individual (Figure 1). A FLATT operator can use this knowledge to create adaptations directed towards that trainee's motivational tendencies. Someone with low intrinsic motivation and effort would most likely benefit by increasing the number of TSCs pertaining to feedback messages; whereas the learner with a high intrinsic motivation would benefit from TSCs to make the scenario more competitive.

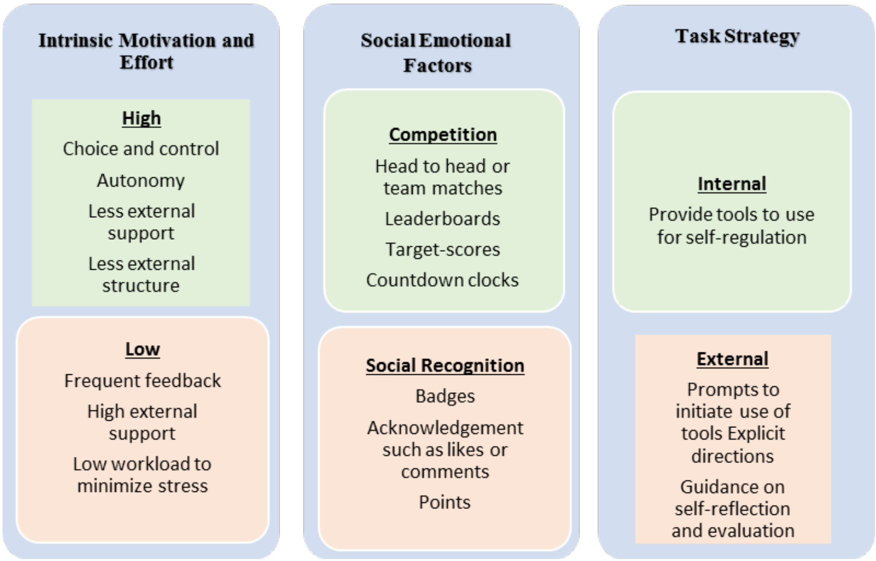

Intrinsic Motivation and Effort
High
Choice and control
Autonomy
Less external support
Less external structure
Low
Frequent feedback
High external support
Low workload to minimize stress
Social Emotional Factors
Competition
Head to head or team matches
Leaderboards
Target-scores
Countdown clocks
Social Recognition
Badges
Acknowledgement such as likes or comments
Points
Task Strategy
Internal
Provide tools to use for self-regulation
External
Prompts to initiate use of tools Explicit directions
Guidance on self-reflection and evaluation

Figure 3: Motivators and Suggested Training Adaptations

GUI

The GUI is extremely important since this is what the FLATT users will be interacting with. The main purpose for FLATT's user interface is to make it intuitive and easy-to-use while allowing the user to perform advanced tasks.

We chose ReteJS (ReteJS) because it allows for simple node-based authoring as seen in Figure 2. This "flow" is intuitive to the user so there is a minimal learning curve. It is a modular event-based architecture which makes it possible to add new functionality in the form of plugins. Due to these plugins, ReteJS is extremely flexible, allowing plugins to be connected that are best suited for designing the FLATT interface. Regardless of which domain is used to design the FLATT architecture, ReteJS is universal and is not bound to a specific domain, but only visualizes and processes the node editor.

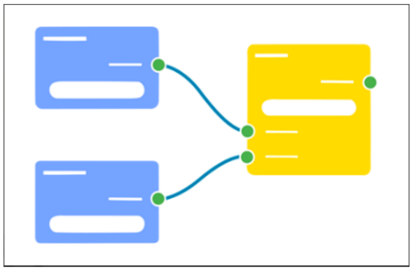
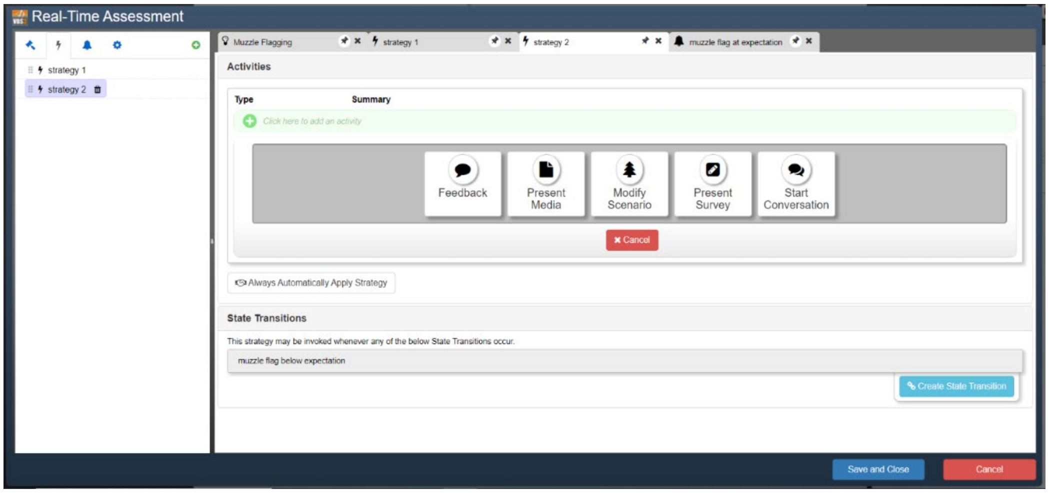


Figure 4: ReteJS visual scripting

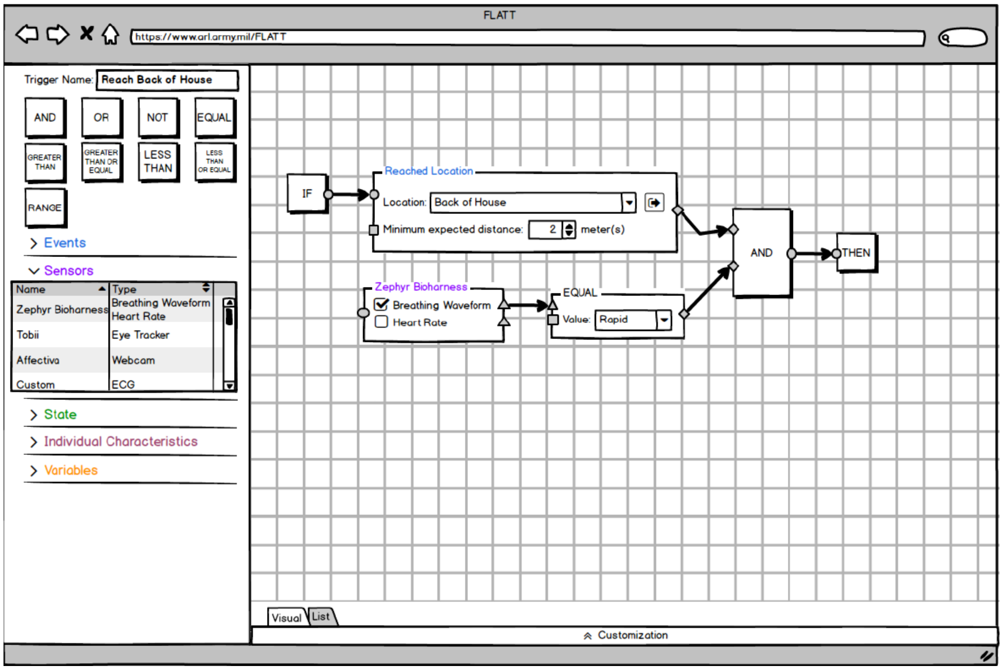
When designing a DKF in GIFT (Figure 3), the user currently has to navigate through several tabs in the authoring tool to configure the scenario adaptation/inject and state transition individually to create a "rule" – for example, if the user wants multiple adaptations to apply after one state transition, they would have to author each adaptation individually (one adaptation takes an average of 5-6 mouse clicks), then author the state transition in a different tab and select the adaptation set to apply (one state transition takes an average of 4-5 mouse clicks). In FLATT (Figure 4), the user has access to all VTE specific components that are available in one simple drag and drop interface (reducing the number of mouse clicks needed to an average of 3-4 total) so they can visualize the flow of events starting with the "triggers", ie. Trainee performance, sensor data, etc., and the resulting customization that will occur.



This strategy may be invoked whenever any of the below State Transitions occur.

muzzle flag below expectation

Figure 5: DKF Authoring Interface



Trigger Name: Reach Back of House

Name	Type
Zephyr Bioharness	Breathing Waveform Heart Rate
Tobii	Eye Tracker
Affectiva	Webcam
Custom	ECG

Location: Back of House

Minimum expected distance: 2 meter(s)

☑ Breathing Waveform

☐ Heart Rate

Value: Rapid

Figure 6: FLATT Authoring Interface

Rules Engine

The rules engine is the backbone to the entire FLATT application. IT will be used to determine when a FLATT trigger criteria has been fulfilled and then will execute the TSCs to adapt the VTE. There are many different third-party libraries specifically designed to handle this problem-space efficiently. FLATT is designed to be modular, so it will not be dependent on any one engine, so if a better library is found later on, it can be easily swapped and utilized within FLATT.

For initial development we chose to integrate the Easy Rules library (EasyRules). Easy Rules was chosen because it has fewer licensing restrictions, a low to middling footprint, and excellent API documentation. This engine is Java-based, which fits well with our Java-based program. Easy Rules handles dynamic changes to the rules in real-time, this allows us to modify the rules while an active session is ongoing. Other rules engines, including OpenL Tables, Rule Engine 2.0, Clara, JSON Rules, Node-Rules, and Nooles, were also researched but were ultimately eliminated as a compatible choice for FLATT due to limited functionality, difficulty of use, large footprint, and no longer being developed/maintained.

FLATT Architecture

When authoring in FLATT, the user will create a set of rules to specify *when* and *how* to adapt a VTE. Each rule is composed of a trigger and a customization (Figure 5). The rule trigger determines when a rule should be executed, whereas the customization specifies how to adapt the VTE. The trigger can be made of any number of conditions or inputs including sensor data, VTE events, individual trainee characteristics, trainee state, etc. The customization can execute any number of trainee state-driven customizations (TSCs) including VTE adaptations, state changes, direct messages to the trainee, etc.

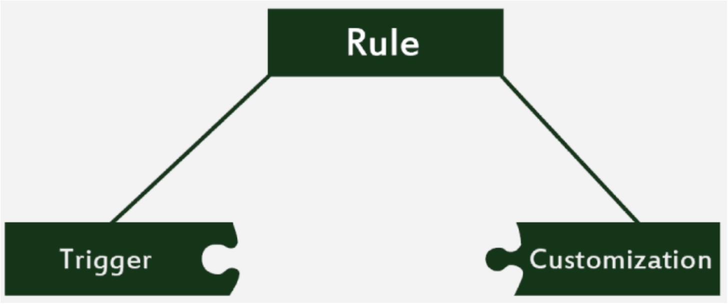


Figure 7: Rule Components

FLATT can maintain its simplistic design because it partners with the VTE and ITS in order to reuse the work already done in those applications. The VTE is responsible for physically running the scenario and can output useful data such as DIS and trainee game states. The ITS (GIFT) is the powerhouse behind the scenes for FLATT. By utilizing GIFT as a service, FLATT delegates the complicated and intensive task of needing to receive, analyze, and process the incoming data streams (Figure 6). GIFT is able to perform the heavy-lifting and can then send FLATT only the information that is relevant, such as assessed learner states, individual characteristic levels (e.g., high grit, low motivation), and processed sensor data. FLATT receives this consolidated dataset and determines if any of the authored rules have been satisfied by using an internal Rules Engine. Once a FLATT rule is triggered, FLATT can use GIFT as a service once again. Since GIFT already has an established communication pathway to the VTE, it manages sending the training adaptations to be applied. The TSCs are sent to GIFT, which then determines if the action can be managed by itself or needs to be communicated to the VTE. GIFT is able to handle communicating information to the trainee, such as presenting a survey, whereas the VTE is responsible for adaptations involving the game environment, such as creating a new in-game actor, highlighting hostiles, modifying the environment with rain or fog, and other supported adaptations.

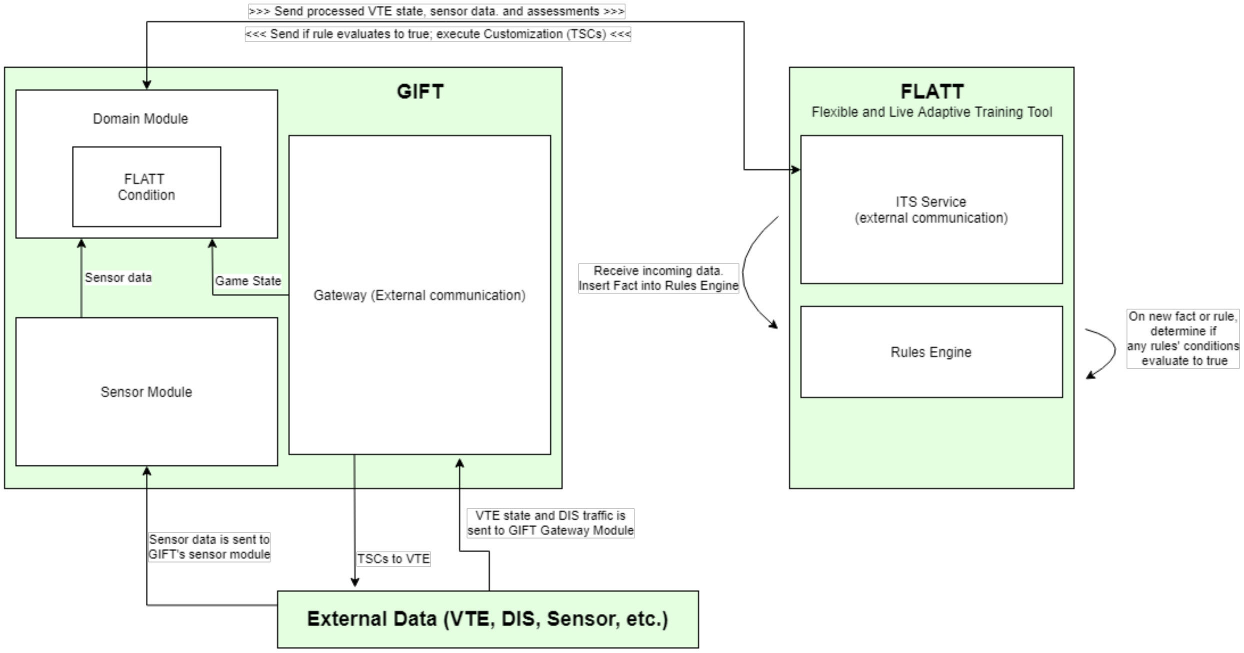

>>> Send processed VTE state, sensor data. and assessments >>>
<<< Send if rule evaluates to true; execute Customization (TSCs) <<<
GIFT
Domain Module
FLATT Condition
Sensor data
Game State
Gateway (External communication)
Sensor Module
FLATT
Flexible and Live Adaptive Training Tool
ITS Service (external communication)
Receive incoming data. Insert Fact into Rules Engine
Rules Engine
On new fact or rule, determine if any rules' conditions evaluate to true
Sensor data is sent to GIFT's sensor module
TSCs to VTE
VTE state and DIS traffic is sent to GIFT Gateway Module
External Data (VTE, DIS, Sensor, etc.)

Figure 8: FLATT-GIFT Communication

Use Case

Observing Trainees

FLATT will allow the operators to observe the state of the trainees and data in real-time during a session and in AAR. This will include sensor data, trainee state, and which FLATT rules were executed and when. This can be a "hands-off" approach to monitoring the trainee session.

Authoring Rules

An operator can use the FLATT user interface to author learning models, which express the trainee states of interest for an exercise, and any automated actions to take based upon the trainee state. Prior to a training exercise, the instructor can create if/then rules based on virtual environment data, trainee performance, or other sensor data (eye tracking, posture, etc.). This gives users the ability to rapidly create their own unique TSC triggers and adaptations without the need to hire an engineer to develop a new feature or wait for a new build for a specialized ITS implementation; this saves the users time and money. The FLATT authoring tool is designed to be as user friendly as possible, putting the power in the hands of the scenario designer to create any adaptations they'd need, on the fly. This allows new rules to be created, existing rules to be modified, subjective observations to be recorded, and issuance of virtual environment changes.

Executing TSCs Automatically via Authored Rule

The rules authored in the previous section will be applied in the scenario when their trigger conditions have been fulfilled. This will be done without the need for human intervention. FLATT will record when TSCs have been triggered. The operator will have the ability to manually trigger an existing TSC in real-time regardless of trainee state. If manually applied, FLATT will record the user that executed the TSC.

Providing Unscripted TSCs

During runtime, new rules can be created and existing ones modified. These new and updated rules will take effect immediately and will be executed automatically as soon as their conditions are met. The operator will also have the ability to manually execute a TSC in real-time regardless of trainee state without first needing to author a rule.

FLATT as a service for GIFT

FLATT is also able to provide additional functionality to external applications such as GIFT by providing its live session features as a service. Each feature will have its own service so GIFT can embed them individually into its own interfaces. This will be especially useful for GIFT's authoring and Game Master pages.

FLATT Authoring Service

The FLATT authoring tool can be used as a service within GIFT to enhance the course creator experience. The user could be authoring a GIFT course or DKF, and with the click of a button, navigate to the FLATT authoring tool to create rules to adapt the VTE during the session. The user can then return to GIFT to continue building the rest of the course. Using the FLATT authoring tool service, GIFT is able to simplify its own authoring process.

Sensor Service

The sensor service will contain a collection of interactive sensor graphs, which will display the sensor data received from the ITS in real-time during a session (Figure 7). GIFT's Game Master page can embed the sensor graphs into its own page. Seeing the visualized sensor data will help the operator to maintain a complete view of the trainee state within the exercise. Using this data, the operator can choose to adapt the exercise on the fly in order to maximize the trainee's experience. To reduce clutter, graphs can be minimized so only the relevant data is being displayed. The graphs can also be favorited which will pin the graphs to the top so the operator has easy access to the most relevant data in the moment. If additional sensor information is provided, such as battery life or signal strength, that information will also be displayed.

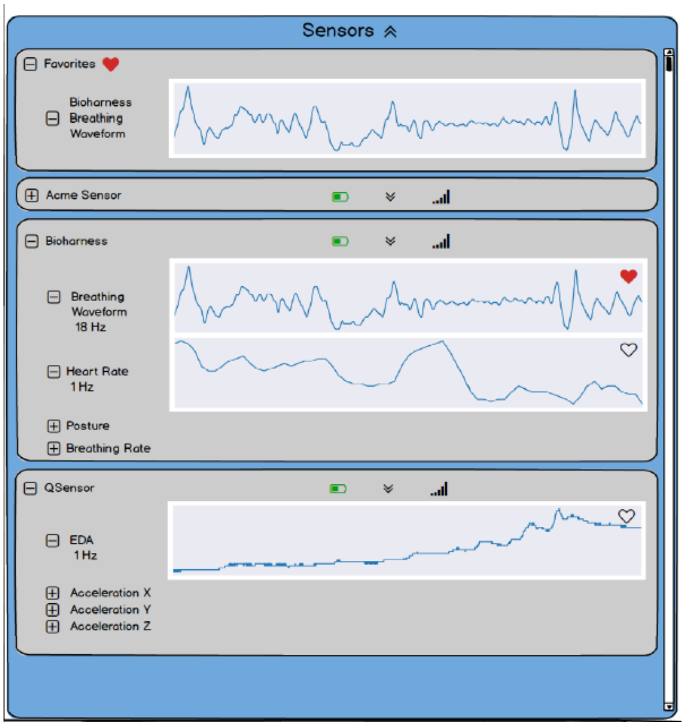


Figure 9: Sensor Graphs

Rule Tracking Service

The last feature will be interactive rule tracking (Figure 8), which is a real-time visual representation of the active FLATT rulesets being applied to the session and the status of each one. The operator can drill down into the rules to see exactly which conditions in the rule have been satisfied and which have not, allowing them to identify the conditions that are preventing the TSC from being applied. Triggered TSCs will be identified so the operator can evaluate what changes the VTE has undergone and how the trainees are responding to the change. There will also be a feature to allow the operator to create rules on the fly that will be added to the current ruleset in effect or to manually send VTE adaptations instantly.

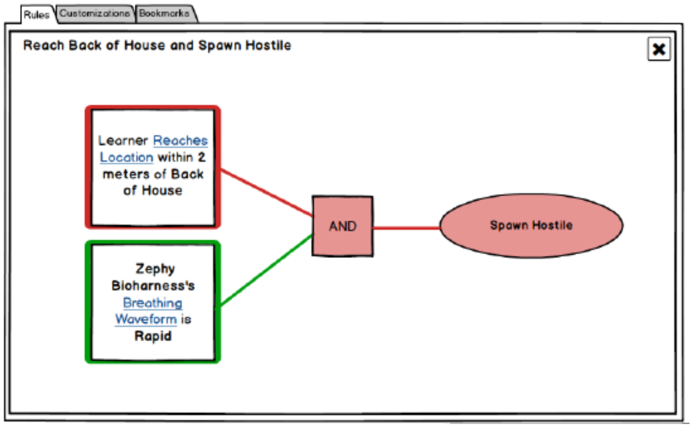

Rules
Customizations
Bookmarks
Reach Back of House and Spawn Hostile
Learner Reaches Location within 2 meters of Back of House
Zephy Bioharness's Breathing Waveform is Rapid
AND
Spawn Hostile

Figure 10: Rule Tracking

CONCLUSION

Through FLATT's simplistic authoring tool and flexible adaptation strategies, FLATT will be able to provide researchers with the data to develop new models of learners, and determine effective training applications that are appropriate for each learner state. Trainers can use this data to increase the effectiveness of training solutions by characterizing a learner's state and then adapting the training. FLATT enables the user to avoid wasting precious time and resources building a stovepipe ITS solution that may become obsolete after a few data collections. These resources can instead be focused on researching new scenario adaptation ideas, determining which adaptations are most effective, and improving the overall training experience for the learner.

References

(n.d.). Retrieved from https://gifttutoring.org/projects/gift/wiki/Overview

(n.d.). Retrieved from https://rete.js.org/#/

(n.d.). Retrieved from https://www.zephyranywhere.com/media/download/bioharness3-user-manual.pdf

(n.d.). Retrieved from https://github.com/j-easy/easy-rules

Reinerman-Jones, E. L. (2017). Informing the Long-Term Learner Model: Motivating the Adult Learner (Phase 2).

Schnell, C. J. (2018). Physiological Based Adaptive Training. In Proceedings of 7th Annual GIFT Users Symposium, Orlando, FL. Retrieved from https://www.gifttutoring.org/attachments/download/2136/10_Learner%20Modeling%20Paper_Schnell%20et%20al.pdf

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S Army DEVCOM – Soldier Center. The statements and opinions expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHORs

***Steven Harrison** is a software engineer at Dignitas Technologies and the technical lead for the FLATT project. He has over 10 years of professional software research and development experience. Steven has been a member of the GIFT development team since 2017 and is also the technical lead for the Squad Advanced Marksmanship Trainer (SAMT) effort that integrates GIFT intelligent tutoring with marksmanship training simulators, used on systems such as the Engagement Skills Trainer (EST). The Marksmanship effort supports research and human experimentation on providing ITS capabilities for training marksmanship, including research for capturing various sensor and device data streams. Steven has been responsible for ensuring the development of FLATT and SAMT.*

***Elyse Burmester** is a Test Engineer at Dignitas Technologies. Since 2017, she has contributed as a member of the GIFT development team in addition to directly supporting the Creative and Effective Training Technologies Branch at U.S. Army Combat Capabilities Development Command - Soldier Center (CCDC-SC) SFC Paul Ray Smith Simulation & Training Technology Center (STTC). Ms. Burmester has a bachelor degree in Political Science from Florida Gulf Coast University.*

Automated Assessment of Teamwork Competencies using Evidence-Centered Design-Based Natural Language Processing Approach

Randall Spain[1], Wookhee Min[1], Jason Saville[1], Keith Brawner[2], Bradford Mott[1], and James Lester[1]
North Carolina State University[1]
U.S. Army Combat Capabilities Development Command Soldier Center – STTC[2]

Introduction

Team communication offers a critical medium for investigating how teams collaborate, coordinate, and distribute information as members work towards a shared goal. Analyzing communication between team members can provide evidence for teamwork competencies, including information exchange, supporting behavior, initiating leadership, and communication (Marlow et al., 2018; Sottilare et al., 2018). Because many teamwork and team decision-making behaviors can be assessed by monitoring a team's verbal communication, a significant task for the adaptive instructional system (AIS) research community is to develop natural language recognition and processing capabilities that facilitate team performance measurement. In particular, being able to leverage natural language processing (NLP) capabilities to automatically categorize verbal behaviors into teamwork competencies would represent a significant step towards the development of AISs for teams that can partially or fully automate the assessment of teamwork skills and team performance.

Over the past year, the U.S. Army Combat Capabilities Development Command Soldier Center, Simulation and Training Technology Center (CCDC-Soldier Center, STTC) and North Carolina State University have been collaboratively developing a deep learning-driven NLP framework that aims to automatically analyze team communication data, parse it into classifications schemes, and provide summary statistics of critical team communication features that can be used to analyze and identify antecedents of team performance. The goal of this framework is to integrate NLP assessment capabilities into the Generalized Intelligent Framework for Tutoring (GIFT), an open-source software framework of tools, methods, and standards for developing AISs. To date, much of our research has focused on assessing the accuracy of automatic speech recognition capabilities and devising models that can automatically classify team communication utterances into speech act and team development labels (Min et al., 2021). A significant step towards realizing the potential of NLP-based assessment capabilities is being able to develop robust team communication evidence models that can drive accurate assessment of team performance and teamwork skills to provide adaptive support in AISs for teams.

In this paper, we investigate whether advances in NLP can be utilized to detect when teams have engaged in key communication patterns or behaviors that represent evidence of engaging in teamwork competencies. For accurate assessment of team performance and teamwork skills, we adopt an evidence-centered assessment design approach (Mislevy et al., 2003). We specifically focus on evidence models that process team communication data and estimate beliefs about the state of competency variables centered on advanced situation awareness (ASA), information exchange, backup behavior, and communication delivery. Evidence models generally consist of evidence rules and statistical models. Evidence rules produce observable, predictive features that effectively summarize trainees' performance from work products, while statistical models, often designed as Bayesian networks, account for estimating beliefs about competency variables given observations. This evidence-centered design (ECD) approach is similar to assessment approaches used by team science researchers, such as the behavioral observation scales and Targeted Acceptable Responses to Generated Events or Tasks (TARGETs) methodology (Folkes et al.,

1994) that link team competencies to team performance objectives and are used to measure evidence of teamwork skills (i.e., information sharing, providing backup, taking initiative, and communicating clearly and concisely).

We describe an initial attempt towards automating the assessment of teamwork competencies captured during a live training session from the Squad Overmatch Project (Johnston et al., 2019). To effectively deal with a limited size of the labeled dataset, our approach utilized n-gram linguistic feature-based evidence models that analyze team-level communication data and automatically detect evidence of teams engaging in teamwork behaviors. Additionally, we explore speech labeling categorizations that are related to performance of different competencies of interest. We describe the n-gram-based distinctive communication pattern analysis method that we investigated to facilitate the analysis and prediction of teamwork competencies and evidence states used to evaluate team performance. We conclude with a discussion of directions for future research and design recommendations for integrating NLP-based assessment approaches into GIFT to support the assessment of teamwork skills using team communication data.

BACKGROUND

Accurate assessment of teamwork and team performance is critical for meeting the Army's vision of providing automated coaching and feedback to squads and collective units in synthetic training environments. A common technique applied to the measure of teamwork in scenario and simulation-based training events is the TARGETs methodology, which is a behavioral observation-based approach that utilizes trained raters to measure whether team members engage in key teamwork behaviors—such as exchanging information, providing complete and accurate reports, offering backup and assistance, and initiating leadership—during specific events in a training scenario (Folkes et al., 1994). The TARGETs methodology is often used in conjunction with an event-based approach to training (EBAT) which translates team-based competencies into event-based training narratives (Fowlkes et al., 1998). Utilizing the TARGETs methodology, key behaviors are identified that represent teamwork competencies of interest, and the training scenario is then structured with opportunities to demonstrate these behaviors at key timepoints. These events and the associated team behaviors and constructs are translated into a TARGETs checklist that trained human raters use to indicate (Yes vs. No) whether a team engaged in a behavior link to a teamwork construct at a specific window of opportunity in the training scenario. In this manner, behavioral observers and raters can know exactly what they are looking for and when each behavior of interest should occur, allowing for more complete attention to be given to the rating process and further standardization to be utilized (Rosen et al., 2010).

The TARGETs methodology offers a theory-driven approach for devising measures of teamwork but can be limited in its application due to the burden placed on raters particularly during dynamic and face-paced training events which impact measurement reliability. Synthetic training environments offer unique opportunities to utilize theory-driven and data-driven approaches to facilitate obtrusive measurement of team performance (Orvis et al., 2013). Data-driven measurement could be automated or augment the measurement of key teamwork constructs, thus allowing human observers to focus on a limited and select set of competences important for team performance (Salas et al., 2017).

In this paper, we adopted an evidence-centered design approach to estimate beliefs about the state of team competency variables derived from team communication. Evidence-centered design (ECD) offers a methodological framework to assess learners' focal knowledge, skills, and abilities by analyzing data drawn from learners' interactions with a training and learning environment (Mislevy, Almond, & Lukas, 2003). An important step in ECD is evidence modelling focusing on where the skills and knowledge constructs 3 are identified from observed interactions with a broad range of task models. While the design of evidence rules and statistical models is often created through the collaborative work of the domain experts and assessment designers (Mislevy & Riconscente, 2011), more recent work has investigated data-driven approaches to automatically devise evidence models using machine learning techniques such as deep neural networks (Akram et al., 2018; Min et al., 2020), probabilistic graphical models (Georgiadis et al., 2019; Shute et al., 2021), and a hybrid method (Henderson et al., 2020). The inferred evidence on competency variables is then referenced by the competency model to perform a holistic assessment of each learner's proficiencies on focal knowledge, skills, and abilities.

The primary goal of the current work was to apply an ECD methodology to develop evidence models of team performance that utilize team communication data as the primary data stream. Automated assessment of team communication is a key requirement to the development of AIS for team tutoring, and advances in NLP and machine learning offer promising approaches for analyzing team discourse and predicting team communication behaviors (Min et al., 2021). A significant challenge of developing robust team performance assessment models using machine learning techniques is that they require large, labeled datasets, which can be difficult to obtain and labor-intensive to create. Utilizing small datasets can lead to poor generalizability. Instead of adopting machine learning techniques, our data-driven approach identifies n-gram features that are associated with high and low performing teams only, measures overlapping ratios of these two distinct groups in comparison to n-gram instances extracted from the test set, and classifies it to the label with a higher ratio. We use a leave-one-squad-out cross-validation to determine if the approach offers a generalized assessment technique for classifying team performance. We report its strengths and weaknesses based on transcripts of team communication and ratings of squad teamwork from the Squad Overmatch Project (Johnston et al., 2019) and discuss direction for future research to improve GIFT's team assessment capabilities.

DATASET AND METHODOLOGY

Within the framework of the Squad Overmatch project, a series of three scenarios were developed for US Army and US Marine Corps squads to each participate in (referred to as Missions 1, 2, and 3 . Each scenario was developed using the EBAT and TARGETs methodology to elicit opportunities for squads to complete key behaviors, with each behavior associated with both a specific event within the scenario and with a specific competency of interest. Teamwork competencies assessed within each scenario included ASA, information exchange, initiating leadership, supporting behavior, and communication (Johnson et al., 2019). Squads were randomly assigned to either receive teamwork, stress exposure, and tactical combat casualty care training using an integrated training approach (experimental condition) or receive standard tactical training in an outdoor facility (control condition).

The datasets used for this study consist of transcribed audio logs of six U.S. Army squads completing Mission 3, as well as the TARGETs ratings for each squad which identified each behavior as completed or not completed. Mission 3 contained a scripted set of training events in which each squad was tasked with entering a village from prior missions, contacting local key leaders, and gaining intelligence on local gang activity. As the scenario unfolded. squad members encountered critical events that required squad members to engage in coordinated actions, information exchange information, and supportive behavior. The scenario lasted approximately 45 minutes. Transcribed communication from the platoon leader, squad leader, team leaders, and several supporting team members were investigated. A total of 6,181 utterances across all squads were coded using a framework of 27 speech act labels and 18 team development labels, seven ASA labels, 12 event labels and 65 floor labels. Speech act labels represented the basic purpose of a given utterance, such as requesting information or stating an action being taken. Team development labels reflected how information passed within a squad (Saville et al., 2021). ASA labels reflect when team members passed information that described potential threats or threat behavior to other squad members. 4 Floor labels represent sub conversations that occurred between participants (inter and intra-squad communications) over the course of the training event, with each label containing a speaker and addressee. Episode labels reflect major sections of the training scenario.

We investigate a data-driven approach to predict each squad's multi-dimensional team performance with respect to TARGETs ratings on four competency variables (ASA, information exchange, communication, and supporting behaviors) using an ECD framework. Casting it as a multi-task, binary classification problem, we assigned high and low labels based on a median split of the six teams' average competency scores per competency variable (i.e., four different competency grades per squad). A guiding research question was: Can linguistic properties from team communication data be used as evidence to predict teamwork as measured by teamwork and team decision making items used in the Squad Overmatch Project? The primary reason to focus on linguistic features as our first attempt is this data-driven approach is readily applicable upon having a transcript using automatic speech recognition, as opposed to speech act, team development, and ASA labels that require manual annotations by domain experts or a robust NLP classification model. To address this, we extract linguistic features from each team members' utterances made in their communication during Mission 3 of the Squad Overmatch training exercise and predict four teamwork construct labels for a squad in the test set. Specifically, n-gram encoding is used to distill the linguistic features with n varying from 1 to 3, by which we extract a set of salient features observed in team communication while managing the feature set to a reasonable size that can be generalizable to an unseen dataset. Due to the limited dataset size (6 squads), we devised a predictor directly relying on n-gram features rather than using a machine learning technique that can cause an overfitting issue.

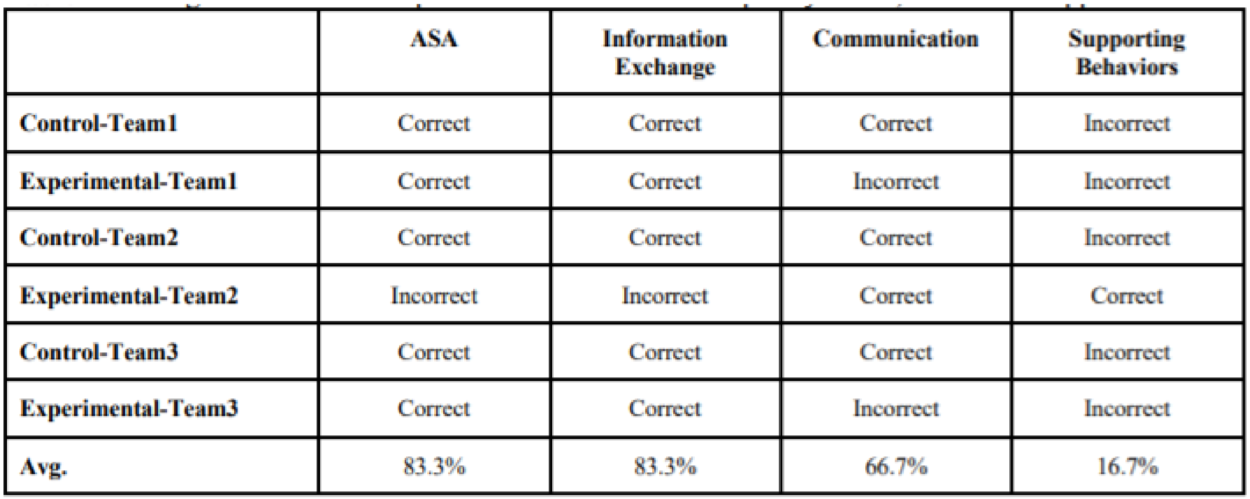
The steps proposed to create a team performance predictor are as follows. First, using the five squads' data only that serves as a training set in each fold in the leave-one-squad-out cross validation setting, we extract a set of n-gram features from the high-performing group (2 or 3 squads) and another set of n-gram features from the low-performing group (2 or 3 squads) per competency variable, as well as the frequency of each n-gram feature. Second, we sort each group's n-gram features based on their frequencies observed in each group's communication. In this way, we identify which words and word sequences were more frequently observed in high-performing and low-performing groups. Third, among the top-k frequent n-gram features per n (we set k to 75 in this work), we identify which n-gram features were commonly observed in both the groups, which n-gram features were observed in the high-performing group only, and which n-gram features were observed in the low-performing group only. Fourth, to identify distinctive patterns between the two groups, we only take each group-specific n-gram features (e.g., n-gram features that only appeared in the high-performing group). Since we exclude commonly observed, frequent n-gram features from both the groups, the number of each group-unique n-gram features remain the same. Finally, we extract n-gram features from the test set (one squad), compare between high-performing and low-performing groups which group has more n-gram features in common with the test set, and classify the test set to the label of the group that is shown to have more common n-gram features. As a variant of this approach, we also investigate an occurrence-based method, with which we extract n-gram features that are present in the highperforming group or low-performing group only instead of choosing top-k most frequent n-grams, while following the same procedure described above.

RESULTS

Data-Driven Indicators of Teamwork Competencies

Table 1 shows the predictive accuracy of the four competency variables using two data-driven approaches following the leave-one-squad-out cross-validation. The frequency-based, data-driven approach achieves 62.5% of predictive accuracy overall, where high-performing groups and low-performing groups on ASA and information exchange were the most correctly inferred with 83.3% (5 squads out of the 6 squads) 5 predictive accuracy, followed by communication delivery (66.7%; 4 squads out of the 6 squads) and supporting behaviors (16.7%; 1 squad out of the 6 squads). It should be noted that each squad was labeled with a median split per competency variable (i.e., high-performing squads and 3 low-performing squads per competency), and thus the class label of the test squad data to be predicted in each cross-validation split is a minor class. For this reason, the majority class baseline is not an effective solution for not being able to classify each squad instance into the minor class label (0% accuracy rate). The occurrence-based approach, which assessed uniquely occurring words and phrases in high and low performing groups, shows a similar predictive performance to the frequency-based approach, though its predictions of communication delivery (16.7%) are significantly inaccurate compared to the frequency-based classification (66.7%).

Table 1. Average cross-validation prediction results of the frequency-based, data-driven approach.



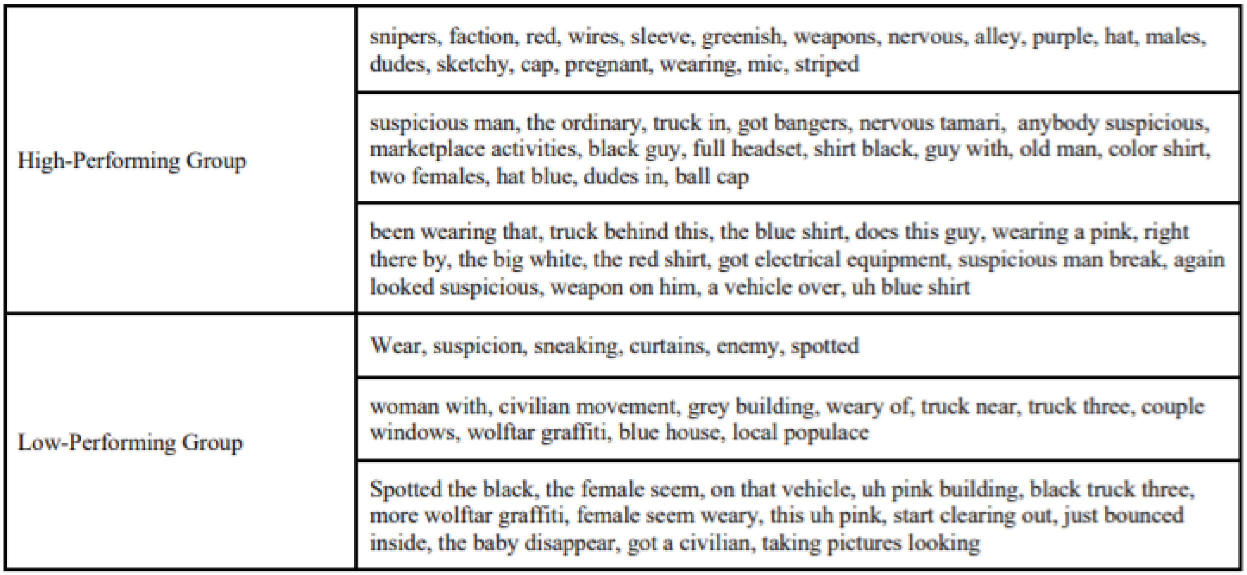
	ASA	Information Exchange	Communication	Supporting Behaviors
Control-Team1	Correct	Correct	Correct	Incorrect
Experimental-Team1	Correct	Correct	Incorrect	Incorrect
Control-Team2	Correct	Correct	Correct	Incorrect
Experimental-Team2	Incorrect	Incorrect	Correct	Correct
Control-Team3	Correct	Correct	Correct	Incorrect
Experimental-Team3	Correct	Correct	Incorrect	Incorrect
Avg.	83.3%	83.3%	66.7%	16.7%

While teamwork competencies such as ASA, information exchange, and communication delivery were inferred using team members' verbal communication content, our preliminary analyses suggest that assessing behavior-based competencies, such as supporting behaviors, which are often measured by monitoring teams' nonverbal backup behaviors, was not reliably achieved solely using the data driven linguistic features approach. The results suggest that other types of features (e.g., speech act, team development, ASA labels) and data (e.g., non-verbal behaviors) will be important to incorporate in our data-driven evidence models to improve the predictive performance.

Table 2 below shows commonly observed n-gram tokens for the high-performing group (3 squads) and the low-performing group (3 squads) for each competency variable (i.e., ASA, information exchange, communication delivery, supporting behaviors). These key words were derived from the occurrence-based approach and show unique phrases and words that distinguish between high and low performing squads.

An analysis of n-gram results for ASA revealed that compared to high-performing teams, low-performing teams generated fewer unique unigrams (229 versus 535), bigrams (1,805 versus 4,142), and trigrams (3,036 versus 6,760). While one primary factor that impacted the lower frequencies of unique n-grams for the low-performing group is that high performing teams had greater total utterances (Saville et al., 2021), a qualitative review of the n-gram results suggests that higher-performing teams exhibited greater detail in their terminology, with identified colors, locations, and articles of clothing more salient throughout each ngram list. It is important to note that the ASA TARGETs items captured how well squads identified and communicated atmospheric and behavioral deviations from a previously established baseline. Highperforming groups' communication indicated they shared and passed more detailed information about potential threats than low-performing groups. That is, the quality of the information they passed and the quantity was greater compared to low performing groups. Although the example n-grams included in the 6 table above demonstrate that low-performing teams do still describe threats and critical information, the salience of detail in n-grams of high-performing squads assists in highlighting an area of improvement for advanced situational awareness for low performing squads not wholly captured via the labeling and classification process.

Table 2. Examples of distinct n-gram features for ASA based on the occurrence-based, data-driven approach. These n-gram features reported were extracted from the dataset in the Squad Overmatch project.



High-Performing Group	snipers, faction, red, wires, sleeve, greenish, weapons, nervous, alley, purple, hat, males, dudes, sketchy, cap, pregnant, wearing, mic, striped
	suspicious man, the ordinary, truck in, got bangers, nervous tamari, anybody suspicious, marketplace activities, black guy, full headset, shirt black, guy with, old man, color shirt, two females, hat blue, dudes in, ball cap
	been wearing that, truck behind this, the blue shirt, does this guy, wearing a pink, right there by, the big white, the red shirt, got electrical equipment, suspicious man break, again looked suspicious, weapon on him, a vehicle over, uh blue shirt
Low-Performing Group	Wear, suspicion, sneaking, curtains, enemy, spotted
	woman with, civilian movement, grey building, weary of, truck near, truck three, couple windows, wolftar graffiti, blue house, local populace
	Spotted the black, the female seem, on that vehicle, uh pink building, black truck three, more wolftar graffiti, female seem weary, this uh pink, start clearing out, just bounced inside, the baby disappear, got a civilian, taking pictures looking

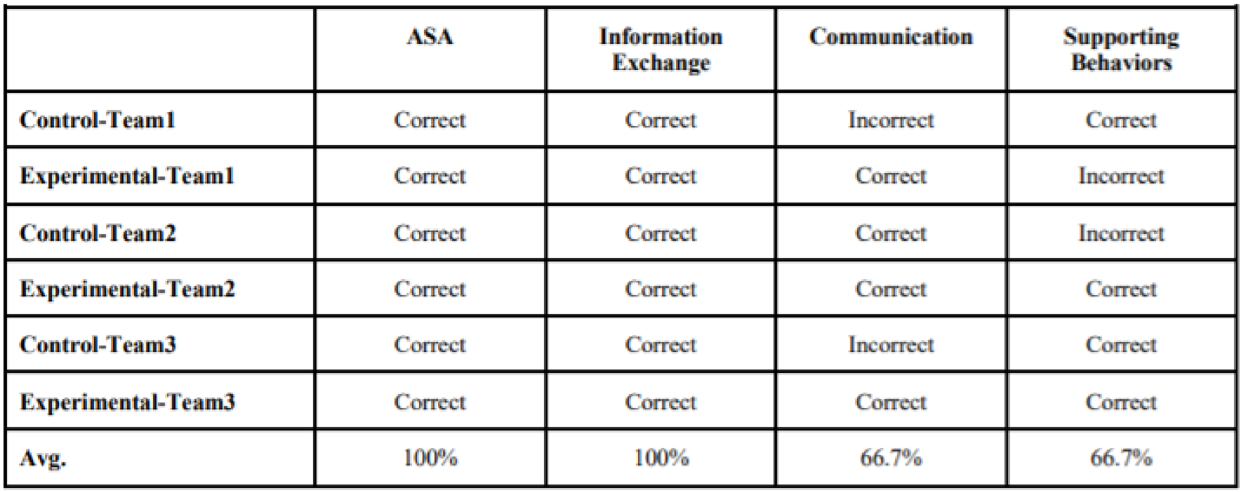
Rule-Driven Indicators of Teamwork Competencies

In addition to examining two data-driven approaches, we also created a set of rule-based indicators to explore how accurately we could replicate the classification of high and low performing squads from the TARGETs ratings provided by expert human raters. To facilitate this exploratory analysis, the team examined each TARGETs behavior and decided upon labels and patterns within the communication transcripts that would indicate completion of that behavior. For example, episode seven included an event described as “report intel about the high value target to the platoon leader,” with a TARGETs behavior of “a situation update is provided up the chain of command.” We determined that completion of this behavior could be identified via the total number of utterances of passing or providing information up the chain of command within episode 7 that included communications with the platoon leader. Thus, each rule effectively served as a filter that could be applied to the data to generate a frequency-based output for the number of team communication behaviors that matched the rule for each squad. This process was repeated for each TARGETs behavior that the team could identify speech indicators for. A total sum (referred to as an “indicator score”) was calculated for each competency based on these speech indicators for the competencies of situational awareness, information exchange, supporting behavior, and communication. We then split the data into high and low groups using a median split per competency and compared the classification results from the indicator driven predictions to the rankings of high and low-performing teams derived from TARGETs from the Squad Overmatch project.

Preliminary results show the indicator data matched the classification of high and low squads from the TARGETs checklist ratings at an agreement rate of 100% (6 squads out of 6 squads) for ASA and information exchange (Table 3). The classification results for supporting behaviors and communication using the rule-driven approach reached 66.7% and 66.7% accuracy respectively. Next, we computed 7 Pearson's correlations to statistically compare the rule-based indicator scores for each squad to the

TARGETs checklist competency scores to determine whether the generated indicator frameworks co-vary with true competency performance scores. A significant positive correlation was found between true scores and indicator scores of information exchange (r = .950, p = .004). Non-significant correlations were found between true scores and indicator scores of ASA (r = .695, p = .125), supporting behavior (r = .534, p = .275), or communication (r = .215, p = .683).

Table 3. Accuracy of rule-driven competency indicator scores in predicting high TARGETs performance.



	ASA	Information Exchange	Communication	Supporting Behaviors
Control-Team1	Correct	Correct	Incorrect	Correct
Experimental-Team1	Correct	Correct	Correct	Incorrect
Control-Team2	Correct	Correct	Correct	Incorrect
Experimental-Team2	Correct	Correct	Correct	Correct
Control-Team3	Correct	Correct	Incorrect	Correct
Experimental-Team3	Correct	Correct	Correct	Correct
Avg.	100%	100%	66.7%	66.7%

DISCUSSION

In this paper we describe an initial attempt towards automating the assessment of teamwork competencies captured during a live training session by analyzing team communication data. To effectively deal with the limited labeled dataset, our approach utilizes n-gram linguistic feature-based evidence models that analyze team-level communication data and automatically detect evidence of teams engaging in teamwork behaviors. Results suggest that using n-grams to analyze team utterances offers a promising technique for classifying squads into high and low-performing groups for teamwork competencies tied to information exchange and ASA, but that other types of features (e.g., speech act, team development, ASA labels) and data (e.g., non-verbal behaviors) will be important to incorporate in our data-driven evidence models to improve the predictive performance. Results also suggest that rule-driven indicator scores could offer a reasonable approach for predicting dichotomous group membership of high or low-performing squads for information exchange and ASA competencies, while exhibiting a significant improvement (66.7%) over the data-driven approach (16.7%) for predicting team performance on supporting behaviors.

Our rule-driven approach has several notable limitations. We did not generate rules for all of the TARGETs items because several behaviors did not lend themselves to analysis using the speech label data. The ruledriven analyses also assumed higher communication frequencies were indicative of more evidence of the teamwork competencies. While this assumption may be feasible for constructs such as information exchange and ASA which are reflected in team communication content and patterns, it will likely be less feasible for other teamwork constructs that are difficult to discern when analyzing speech acts. Furthermore, we did not evaluate the rule-driven approach's generalization performance. The outputs of the rule-based features need to be further evaluated to determine their reliability and validity. As a work in progress, we will continue to explore how advances in NLP and domain knowledge can be incorporated to facilitate team performance measurement in AISs for teams.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

Accurate and comprehensive assessment of team performance can be quite complex, often requiring multiple data collection strategies to reliably measure variables of interest. One advantage of the evidencecentered design we utilized is it offers a data-driven approach that automatically analyzes basic linguistic features to predict teamwork competencies during training scenarios. Data-driven approaches to team communication assess could ease the burden placed on instructors or observers to rate and provide evidence of team performance constructs during dynamic synthetic training events. Our preliminary results suggest that certain teamwork competencies lend themselves to being more accurately predicted by analyzing linguistic features and by using rule-driven approaches than others. In GIFT this can be helpful for identifying team variables that can be reliably and accurately assessed and predicted using team communication data and which competencies should be assessed using a human-in-the-loop approach with GIFT's GameMaster Interface.

In the future, it will be important to develop a robust team performance assessor with improved predictive performance by effectively integrating the data-driven approach with the domain knowledge-based, ruledriven approach in GIFT. This line of research can be greatly enhanced by gaining access to large and labeled datasets that can be used to train prediction models. In addition, to fully automate the evidencecentered design process, our assessment framework should be expanded to incorporate automatic speech recognition (e.g., Spain et al., 2020) and team communication analysis models (e.g., Min et al., 2021), while both quantitative and qualitative predictive performance of the assessment models should be measured thoroughly to guide adaptive training environments. While the n-gram approach demonstrates distinctive patterns between high and low-performing groups, distributed language representation methods (e.g., Devlin et al., 2019) could additionally provide semantic and contextual information underlying the team communication to develop reliable evidence models. In our future work, we will continue to explore how researchers and AIS developers can associate tasks and concepts in GIFT with speech features that can be utilized to generate evidence of teamwork and guide team assessment.

CONCLUSION

Text

References

Akram, B., Min, W., Wiebe, E., Mott, B., Boyer, K. E., & Lester, J. (2018). Improving stealth assessment in gamebased learning with LSTM-based analytics. In K. E. Boyer & M. Yudelson (Eds.), Proceedings of the 11th international conference on educational data mining (pp. 208 - 218).

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), NAACL-HLT 2019 (pp. 4171-4186). Association for Computational Linguistics.

Fowlkes, J. E., Dwyer, D. J., Oser, R. L., & Salas, E. (1998). Event-based approach to training (EBAT). The International Journal of Aviation Psychology, 8(3), 209-221.

Fowlkes, J. E., Lane, N. E., Salas, E., Franz, T., & Oser, R. (1994). Improving the measurement of team performance: The TARGETs methodology. Military Psychology, 6(1), 47–61. 9

Georgiadis, K., van Lankveld, G., Bahreini, K., & Westera, W. (2019). Learning analytics should analyze the learning: proposing a generic stealth assessment tool. In 2019 IEEE conference on games (CoG) (pp. 1-8). IEEE.

Henderson, N., Kumaran, V., Min, W., Mott, B., Wu, Z., Boulden, D., Lord, T., Reichsman, F., Dorsey, C., Wiebe, E., & Lester, J. (2020). Enhancing Student Competency Models for Game-Based Learning with a Hybrid Stealth Assessment Framework. In Proceedings of the 13th international conference on educational data mining (EDM 2020) (pp. 92-103). International Educational Data Mining Society.

Johnston, J. H., Phillips, H. L., Milham, L. M., Riddle, D. L., Townsend, L. N., DeCostanza, A. H., Patton, D. J., Cox, K. R., & Fitzhugh, S. M. (2019). A team training field research study: extending a theory of team development. Frontiers in Psychology, 10, 1480.

Marlow, S. L., Lacerenza, C. N., Paoletti, J., Burke, C. S., & Salas, E. (2018). Does team communication represent a one-size-fits-all approach: A meta-analysis of team communication and performance. Organizational Behavior and Human Decision Processes, 144, 145-170.

Min, W., Frankosky, M. H., Mott, B. W., Rowe, J. P., Smith, A., Wiebe, E., Boyer, K. E., & Lester, J. C. (2020). DeepStealth: Game-based learning stealth assessment with deep neural networks. IEEE Transactions on Learning Technologies, 13(2), 312-325.

Min, W., Spain, R., Saville, J. D., Mott, B., Brawner, K., Johnston, J., & Lester, J. (2021). Multidimensional team communication modeling for adaptive team training: A hybrid deep learning and graphical modeling framework. To appear in Artificial intelligence in education: 21st International Conference, AIED 2021. Springer International Publishing.

Mislevy, R. J., Almond, R. G., & Lukas, J. F. (2003). A brief introduction to evidence-centered design. ETS Research Report Series, 2003(1), 1-29.

Mislevy, R. J., & Riconscente, M. M. (2011). Evidence-centered assessment design. In S. Lane, M. R. Raymond, & T. M. Haladyna (Eds.), Handbook of test development (pp. 75-104). Routledge.

Orvis, K. L., DeCostanza, A. H., & Duchon, A. (2013). In The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC) (No. 1).

Rosen, M. A., Weaver, S. J., Lazzara, E. H., Salas, E., Wu, T., Silvestri, S., Schiebel, N, Almeida, S., & King, H. B. (2010). Tools for evaluating team performance in simulation-based training. Journal of Emergencies, Trauma and Shock, 3(4), 353-359.

Salas, E., Reyes, D. L., & Woods, A. L. (2017). The assessment of team performance: Observations and needs. In A. von Davier, M. Zhu, & P. Kyllonen (Eds.), Innovative assessment of collaboration (pp. 21-36). Springer.

Saville, J. D., Spain, R. D., Johnston, J., & Lester, J. (2021, July). Exploration of team communication behaviors from a live training event. To appear in Applied Human Factors and Ergonomics 2021. Springer International Publishing.

Shute, V., Rahimi, S., Smith, G., Ke, F., Almond, R., Dai, C. P., Kuba, R., Liu, Zhichun, Yang, X., & Sun, C. (2021). Maximizing learning without sacrificing the fun: Stealth assessment, adaptivity and learning supports in educational games. Journal of Computer Assisted Learning, 37(1), 127-141.

Sottilare, R. A., Burke, C. S., Salas, E., Sinatra, A. M., Johnston, J. H., & Gilbert, S. B. (2018). Designing adaptive instruction for teams: A meta-analysis. International Journal of Artificial Intelligence in Education, 28(2), 225-264.

Spain, R., Min, W., Saville, J. D., Mott, B., Brawner, K., Johnston, J., Goodwin, G., & Lester, J. (2020, May). Team communication analytics using automated speech recognition. In B. Goldberg (Ed.), Proceedings of the 8th Annual Generalized Intelligent Framework for Tutoring Users Symposium (GIFTSym8) (pp. 145).

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S. Army Research Laboratory under cooperative agreement W912CG-19-2-0001. The statements and opinions expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHOR

***Dr. Randall Spain** is a Research Psychologist in the Center for Educational Informatics at North Carolina State University where he uses principles, theories, and methods of applied psychology to design and evaluate the impact of advanced training technologies on learning and performance. Dr. Spain holds a PhD in Human Factors Psychology and serves on the editorial board for Military Psychology.*

***Dr. Wookhee Min** is a Research Scientist in the Center for Educational Informatics at North Carolina State University. He received his BS degree in Computer Science from Yonsei University, and MS and PhD in Computer Science from North Carolina State University. His research focuses on artificial intelligence centering on user modeling, educational data mining, multimodal learning analytics, and natural language processing, with an emphasis on game-based learning environments.*

***Mr. Jason Saville** is a fourth-year doctoral student of industrial and organizational psychology at North Carolina State University and Graduate Research Assistant in the Center for Educational Informatics. He received his BA degree in Psychology from North Carolina State in 2018 and also serves as a member of the 4D Lab research team, where his research focuses on the intersections of work, psychology, technology, and global development.*

***Dr. Keith Brawner** is a Senior Engineer for the U. S. Army Combat Capability Development Command Soldier Center at the Simulation and Training Technology Center (CCDC-SC-STTC), and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT). He has 13 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.*

***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 worked in the video game industry developing crossplatform middleware solutions used extensively in commercial games and training applications. His research focuses on game-based learning environments, intelligent tutoring systems, and computational models of interactive narrative. He received his PhD in Computer Science from North Carolina State University in 2006.*

***Dr. James Lester** is a Distinguished University 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 PhD 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)*

Enhancement of GIFT Enabled 3A Learning: New additions

Faruk Ahmed[1], Keith Shubeck[1], Xiangen Hu[1,2]
The University of Memphis[1], Central China Normal University[2]

Introduction

This work is an extension of (Ahmed et al., 2020). A modular and domain-independent framework designed for authoring intelligent tutoring systems (ITS) is Generalized Intelligent Framework for Tutoring (GIFT)(R. Sottilare et al., 2013; R. A. Sottilare et al., 2017). In this work we are going to integrate an easy-to-develop technology solution for the 3A (Content-Aware, Context-Aware, and Learner-Aware) learning environment. Although GIFT has a sensory module in addition to other common features of ITS authoring tools, the current online cloud implementation limits its utility due to limited functionalities of the browser. In earlier approaches (Ahmed et al., 2020), we used some features of html5-enabled emotion detection. In this paper, we are using third-party tools to further explore the potential of a 3A extension of GIFT. Among those tools the AWS Rekognition engine captures emotions. The Learning Record Store (LRS) with Expression API (xAPI) is used to blend human tutoring behind an ITS avatar. Proper analysis and utilization of these tools will help to understand and answer many research questions in learning science.

The 3A learning environment is a learner-aware, content-aware, and context aware environment. It can identify the emotions of learners. To connect those emotions with content and context we have added additional components, namely a blend of human tutoring using both text-chat and video-chat. The text-chat is built without a so-called server, and instead only uses a LRS. Although video-chat requires some server elements like Session Traversal of User Datagram Protocol [UDP] Through Network Address Translators [NATs] (STUN) server and SIGNALING server that manages the connections between devices. The backbone used here is a LRS for storing and retrieving records from client side on-demand. We also explored the possibility of a 3A learning environment in a collective environment (e.g., classroom).

With these components in hand, we are designing experiments to collect 3A learning data for further answering research questions.

EMOTION ANALYSIS

The prototype ITS gathers emotions by using the AWS Rekognition technology in combination with the modern browser's HTML5 media capability. Previous research suggests that analyzing facial expressions works best in a multimodal system (D'Mello & Graesser, 2010). In this prototype,the emotions are synchronized with a certain stimulus at a certain time point. For example, a student is asked to answer a statistical question and has ten tries available. The student tries to answer the question correctly and every time he answers that question and waits for feedback his emotion is captured. Finally, when students answer correctly or incorrectly, the emotion is captured again. This approach is interesting because the time immediately before and after a student receives feedback from the tutor may indicate a student's eagerness to learn the content.

Prior work with 3A only involved individualized emotion capture and analysis. We expanded on this work by exploring the potential of analyzing collective emotions in a classroom environment. The same technology is used in this case, but a statistical summary of the emotions of a group of students is analyzed here instead of individual emotions. For example, the percentage of students that are focused along with the percentage of students losing focus after a certain period of time. This could be an indicator that students are losing interest or getting lost in the content. In fact, this approach allows us to identify the times students begin to lose focus.

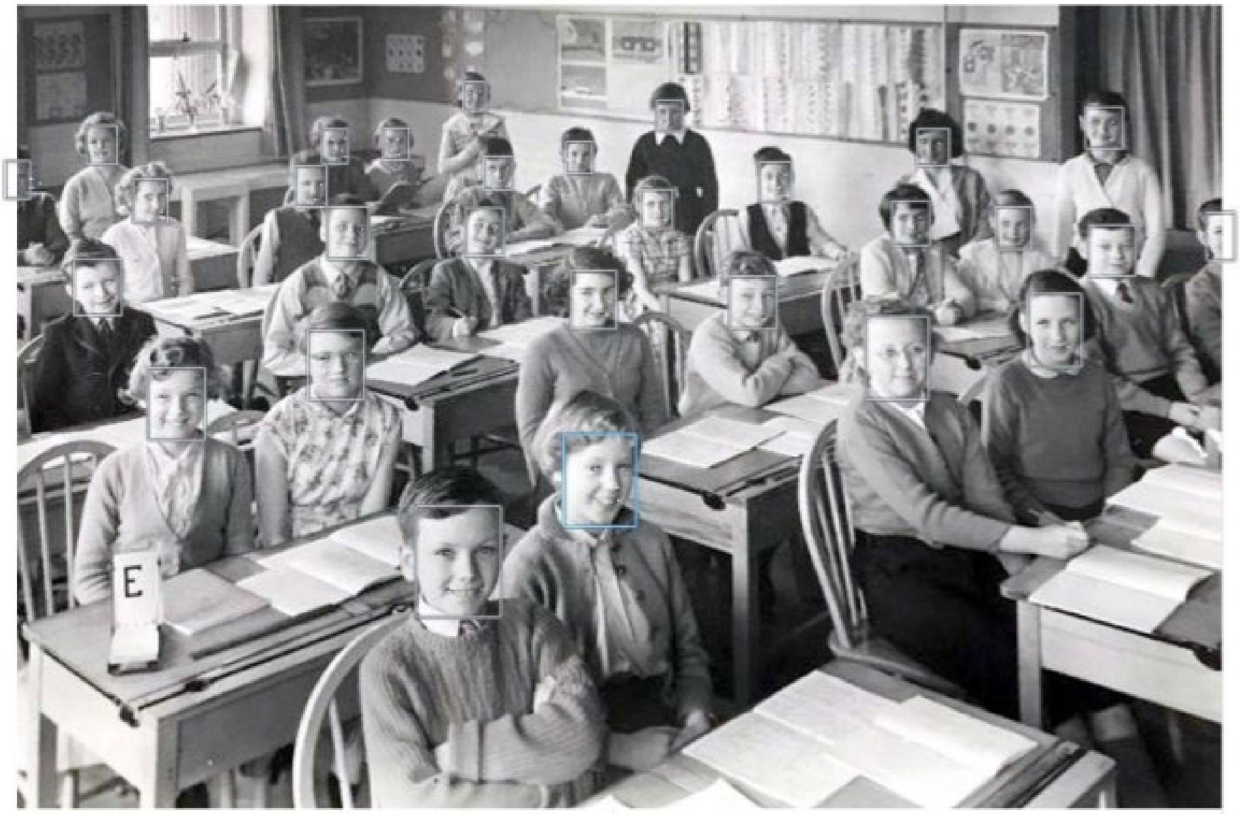
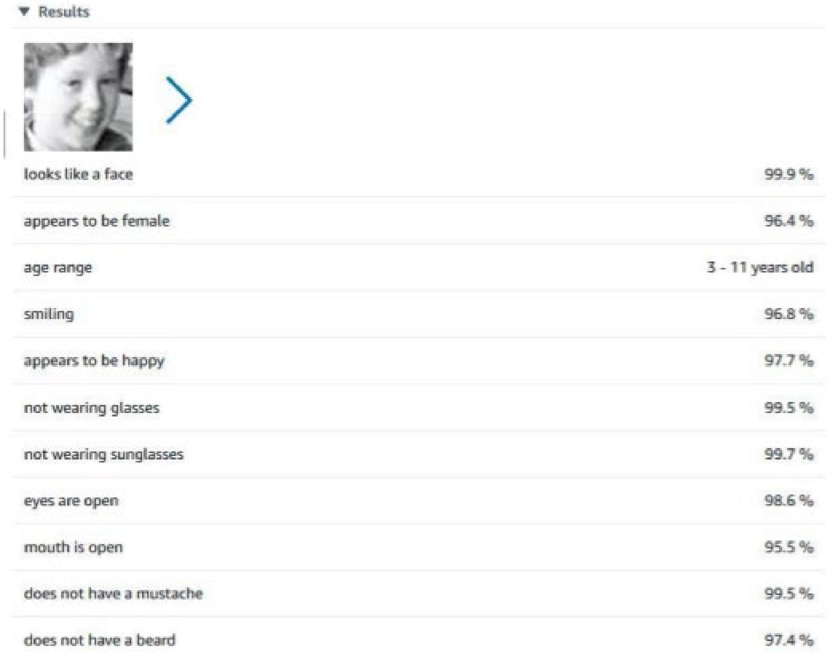


Figure 1 (a). Identified faces in a classroom including partially occluded ones. (Mick Cooper's Home Page, n.d.)



▼ Results

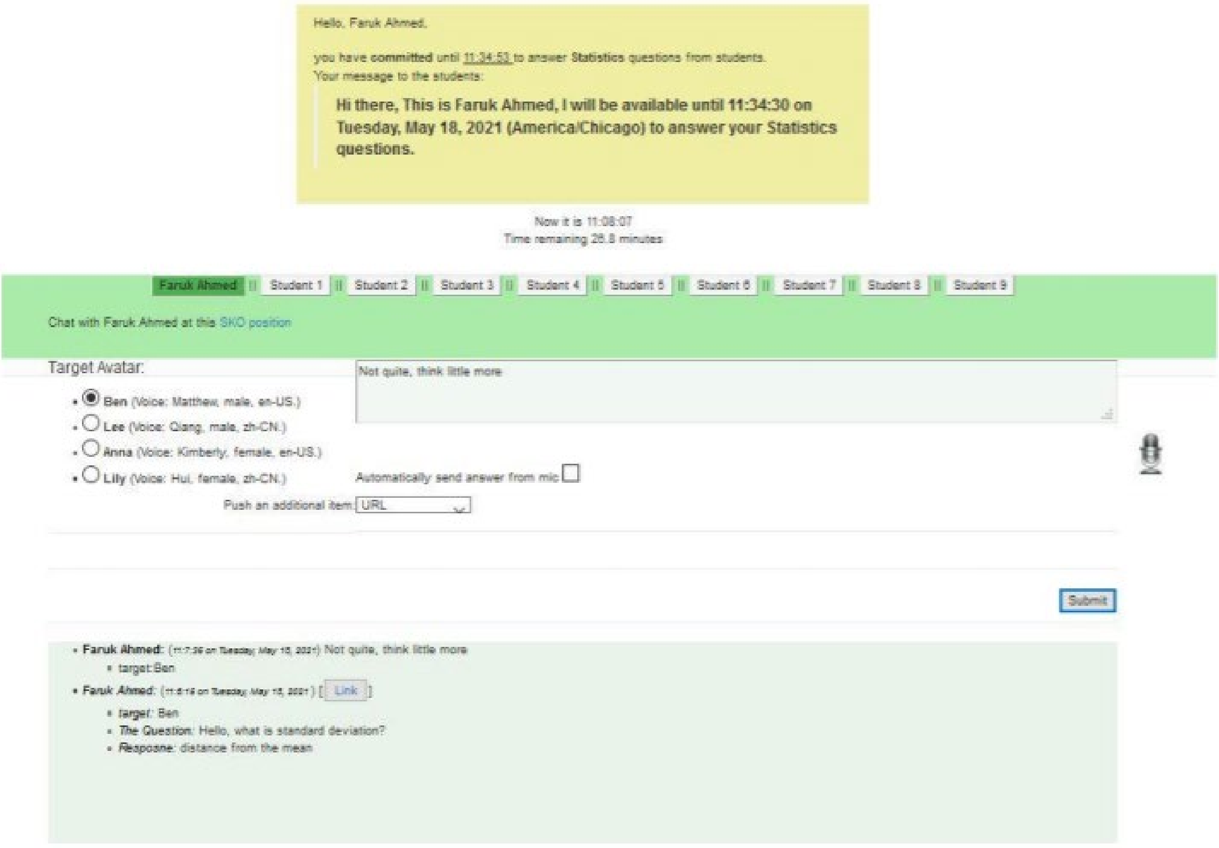
looks like a face	99.9 %
appears to be female	96.4 %
age range	3 - 11 years old
smiling	96.8 %
appears to be happy	97.7 %
not wearing glasses	99.5 %
not wearing sunglasses	99.7 %
eyes are open	98.6 %
mouth is open	95.5 %
does not have a mustache	99.5 %
does not have a beard	97.4 %

Figure 1 (b). Details of a single face

Figure 1 shows an example of a facial expression analysis in a classroom. The figure (a) contains all the faces including occluded ones (e.g., very left and very right on the edge of the picture). Figure (b) shows details of a single face which includes gender, age range, apparent emotion, if wearing glasses, if eyes are open, if mouth is open, if there is a mustache or beard on the face.

BLENDING HUMAN TUTOR WITH SERVERLESS CHAT

The ITS is blended with a human tutor along with an intelligent tutor to perform a Wizard of Oz type experiment. Behind an avatar there will be a human tutor available and they will interact on-demand. Usually students ask questions by typing or by speaking. Sometimes the questions are beyond the scope of an intelligent tutor, which requires human intervention. Moreover, the human tutor's intervention allows students to become more focused on the content. This happens because of their understanding that there is either a human or the tutor is intelligent enough to not be fooled.



Hello, Faruk Ahmed,

you have **committed** until 11:34:53 to answer **Statistics** questions from students.
Your message to the students:

> **Hi there, This is Faruk Ahmed, I will be available until 11:34:30 on Tuesday, May 18, 2021 (America/Chicago) to answer your Statistics questions.**

Now it is 11:08:07
Time remaining 26.8 minutes

Chat with Faruk Ahmed at this SKO position

Target Avatar:

- ● **Ben** (Voice: Matthew, male, en-US.)
- ○ **Lee** (Voice: Qiang, male, zh-CN.)
- ○ **Anna** (Voice: Kimberly, female, en-US.)
- ○ **Lily** (Voice: Hui, female, zh-CN.)

Not quite, think little more

Automatically send answer from mic □

Push an additional item: URL

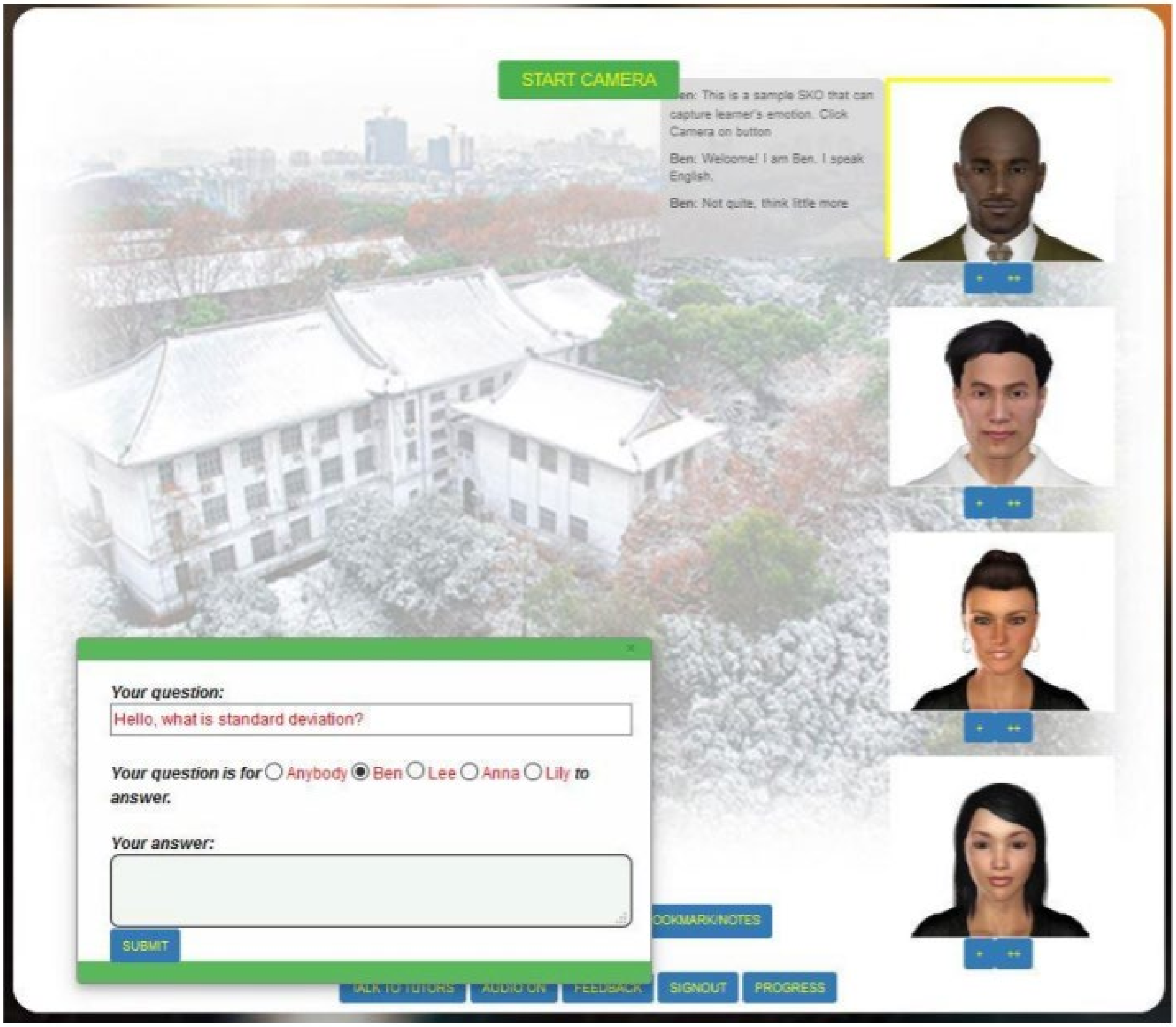
- **Faruk Ahmed**: (11:7:36 on Tuesday, May 18, 2021) Not quite, think little more
  - target:Ben
- ***Faruk Ahmed***: (11:8:16 on Tuesday, May 18, 2021) [Link]
  - ***target***: Ben
  - ***The Question***: Hello, what is standard deviation?
  - ***Response***: distance from the mean

Figure 2. Tutor interface

Figure 2 depicts the tutoring interface. In the tutor interface a tutor is able to choose a certain amount of time for a certain subject (e.g., statistics) to be available. Within that duration students are able to ask questions. The tutor's response to a student question is spoken by the avatar in a specific language; currently, there is no translation enabled.

A tutor can operate in two modes: "Answering questions" and "Monitoring students". In the answering mode, tutors answer questions, whereas in the monitoring mode, tutors can interrupt students. For example, a student may pause to better understand a topic, or reach an impasse, and then ask questions. At this point, a tutor in the student monitor mode can provide additional help or clues in the form of a url directed to relevant content or video.

The conversation with students and teachers can be stored in the LRS. Whenever a student replies, we will querythe LRS to find relevant answers from previous tutoring sessions. These relevant answers will be displayed to the human tutor which they can judge and create the next reply to a student question.Ideally tutors are experts on all topics within the given domain, but in reality tutors may have knowledge gaps that could be filled by accessing responses to questions from previous tutors.



en: This is a sample SKO that can capture learner's emotion. Click Camera on button

Ben: Welcome! I am Ben. I speak English.

Ben: Not quite, think little more

Your question:

Hello, what is standard deviation?

Your question is for ○ Anybody ● Ben ○ Lee ○ Anna ○ Lily ***to answer.***

Your answer:

Figure 3. Student interface

Figure 3 shows the student interface with the avatars as conversational agents. At any point students can choose to stop and ask questions by clicking the "Talk to Tutor" button. Students also can request to see additional questions from a tutor for extra practice. Their responses to these questions further assist the tutor in detecting misconceptions and ultimately improves their ability to help the students. Usually a computer tutor follows a script and provides feedback based on a set of rules, but a blended human tutor participates according to students' needs to provide even more targeted feedback and questions. A knowledge base on a variety of domains can be established by using the interventions provided by the tutor and conversations between students and human tutor. Mining that knowledge base would provide ITSs the ability for true self improvement. For example, the computer tutor's script could be automatically adjusted based on previous interventions or responses that were more successful in improving learning.

The human tutor is also reachable through video chat. When students want to get help from a tutor and request for a video chat, the avatar is replaced by the video feed of the human tutor. This provides an additional way a student can interact live with a tutor on demand.

BLENDING HUMAN TUTOR WITH VIDEO

The human tutor is also reachable through video chat. Once students want to get help from a tutor and request for a video, the avatar is replaced by the video feed of the tutor. This way a student can interact with a tutor on demand.

LEARNING RECORD STORE AS A BACKBONE

The best part of blending an ITS with a human tutor is that the LRS with xAPI performs the heavy lifting of synchronizing chat requests and responses. There is no real server involved. Each AutoTutor terminal is a client and each tutoring interface is a client. These clients post questions and availability in a LRS. Based on the demand, the client pulls answers from the LRS when available. Pushing records in the LRS and pulling filtered information serves as a lambda function for the chat facility.

The LRS plays a role of central data storage (e.g., a public notice board). Any student or tutor (e.g., as a client) can post a notice which is visible to other students and human tutors and they can act accordingly. Figure 4 shows a block diagram of this serverless chat engine built on LRS.

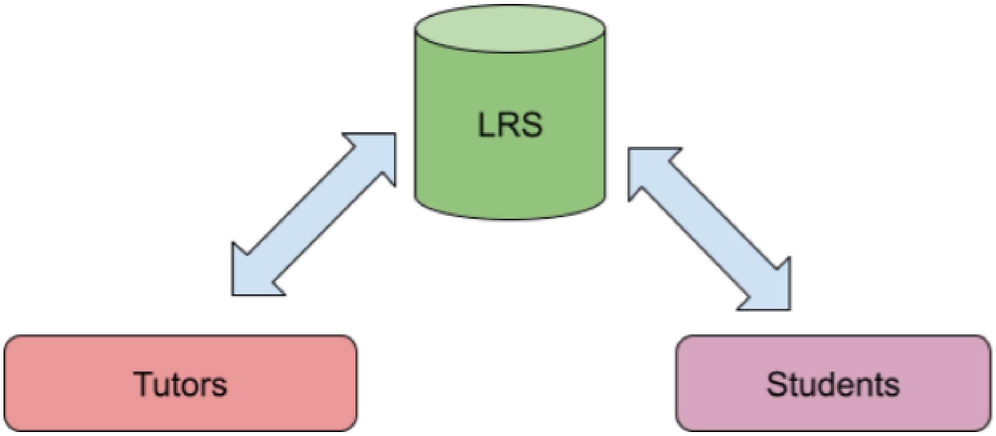


Figure 4. Block diagram of serverless chat engine

WIZARD OF OZ EXPERIMENT

The chat environment is extended to use speech synthesis. In this prototype, a student asks a question, tutors reply with text, but that text is played back via the avatar. The avatar is selected by the tutor. Students do not know that the answers are coming from a human tutor. Moreover, the tutor can see the emotion without looking at the video or face of the student; instead tutors see the facial expression score. This integration can be used to conduct wizard of oz experiments to understand behaviors of human tutors and computer tutors. Additionally, the tutors can reply in multiple languages. We are considering using a translation tool so students and tutors will be able to converse in a multilingual environment.

IMPROVED FEATURES OF GIFT

Earlier work of ours showed the proof of concept of emotion recognition in individualized and collective environments. The current work made it possible to integrate a human tutor with a computer tutor. While this idea is intuitive conceptually, it was not easy technologically. We are able to create a serverless implementation which can be considered as a simple add-on for GIFT. With this technological solution, we envision the following improvements can be added to GIFT.

- Although there is an opinion to regulate AI that interprets human emotions (Crawford, 2021), individual and collective metrics of emotion helps improve both the learning and teaching experience. This is the core concept of adaptive instructional systems (AISs).

- GIFT is being used by learning scientists every day. The inclusion of a blended human tutor facilitates the development of courses. Additionally, it allows for wizard of oz experiments which would give access to learning data to answer numerous learning science and human-computer interaction research questions. Finally, using the LRS reduces the complexity of implementation.

The main idea is to combine the power of GIFT intelligence and intelligence from AI. We explored the possibility of rectifying inaccurate or off-target responses. Blended human tutors will necessarily modify responses generated by machines and eventually machines will learn from humans. This is a method for supervised training of machines over time.

CONCLUSION

The potential of individualized and collective 3A data is huge. Analysis of these data can answer important research questions. For example, what is the optimum delay between a student asking a question and the tutor reply so that a learner is focused?How can we be certain that we are providing high quality answers? How can we efficiently monitor the learning progress of a learner? How many students can a tutor interact with comfortably and efficiently?

References

Ahmed, F., Shubeck, K., & Hu, X. (2020). Towards A Gift Enabled 3a Learning Environment. Proceedings Of The 8th Annual Generalized Intelligent Framework For Tutoring (Gift) Users Symposium (Giftsym8), 87.

Crawford, k. (2021). Time to regulate AI that interprets human emotions. Nature, 592(7853), 167.

D'mello, S. K., & Graesser, A. (2010). Multimodal Semi-Automated Affect Detection From Conversational Cues, Gross Body Language, And Facial Features. User Modeling And User-Adapted Interaction, 20(2), 147–187.

Mick Cooper's home page. (n.d.). Retrieved may 20, 2021, from http://www.michaelcooper.org.uk/

Sottilare, R. A., Brawner, K. W., Sinatra, A. M., & Johnston, J. H. (2017). An Updated Concept For A Generalized Intelligent Framework For Tutoring (Gift). Gifttutoring. Org. Https://Www.Gifttutoring.Org/Attachments/Download/2076/Updated%20concept%20for%20the%20generalized%20intelligent%20framework%20for%20tutoring_9%20may%202017.Pdf

Sottilare, R., Graesser, A., Hu, X., & Holden, H. (2013). Design Recommendations For Intelligent Tutoring Systems: Volume 1: Learner Modeling. Army Research Laboratory.

ABOUT THE AUTHOR

Dr. Faruk Ahmed *is instructor in the Department of Electrical and Computer Engineering at The University of Memphis (UofM). Dr. Ahmed received his MS in electrical engineering from UofM and Ph.D. in engineering from UofM. His primary research areas include Assistive Technology Development, Human Computer Interaction (HCI), Machine Learning, Computer Vision, and Adaptive Instructional Systems (AIS).*

Keith Shubeck *is doctoral candidate in cognitive psychology at the University of Memphis. He received a Certificate of Cognitive Science and a M.S. in psychology in December 2015. He works with his major professor, Dr. Xiangen Hu, in the Advanced Distributed Learning Partnership Lab at the Institute for Intelligent Systems. His dissertation work compares interactive and vicarious tutoring frameworks in a conversation-based intelligent tutoring system for critical thinking and scientific reasoning.*

Dr. Xiangen Hu *is a professor in the Department of Psychology, Department of Electrical and Computer Engineering and Computer Science Department at The University of Memphis (UofM) and senior researcher at the Institute for Intelligent Systems (IIS) at the UofM and is professor and Dean of the School of Psychology at Central China Normal University (CCNU). Dr. Hu received his MS in applied mathematics from Huazhong University of Science and Technology, MA in social sciences and Ph.D. in Cognitive Sciences from the University of California, Irvine. Dr. Hu is the Director of Advanced Distributed Learning (ADL) Partnership Laboratory at the UofM, and is a senior researcher in the Chinese Ministry of Education's Key Laboratory of Adolescent Cyberpsychology and Behavior.*

Talking the Guns: Implementing a Live-Fire Exercise in a Virtual Environment with GIFT

Gregory A. Goodwin[1] and Elyse Burmester[2]
U.S. Army Combat Capabilities Development Command - Soldier Center[1], Dignitas Technologies[2]

Introduction

In 2019 the Program Executive Office (PEO) Soldier sponsored an effort to examine novel measures of squad performance. The lessons learned from this effort were to feed into a squad performance model (SPM) The design of the SPM included three main categories of measures: demographics/background, active measures, and passive measures. The demographic data includes standard items like gender, education, and age as well as records of trainingThe active measures consist of measures of performance (e.g., shooting, movement, communication) taken in real-time during training and operational events while the passive measures would be physiological/biological measures (e.g., heart rate, blood chemistry, etc.). The data contained in the SPM is to be used in conjunction with various models to diagnose/understand and predict performance in both the training and operational environments. The SPM is a sort of competency framework for infantry squads, but as originally described, it doesn't contain any team level metrics. The measures are primarily related to the individual team members.

One of the authors (GG) was able to participate in the squad exercises that PEO Soldier was sponsoring. In these exercises, infantry squads from active duty units conducted a live fire squad attack battle drill (BD 02a). The focus of our research was on measuring the performance of the support-by-fire team. We specifically measured voice communication using audio recorders, and shooting performance using both the hits recorded by the targets and location of misses and hits (LOMAH) sensors. The LOMAH sensors triangulate the shockwave created by supersonic rounds as they pass over the sensor to provide a two dimensional plot of the location of each round as it passes through a plane above the sensor (Goodwin, Teo, Anglin, Schreck, 2020).

By analyzing the verbal communication of the SBF team, we were able to identify elements of leadership, information exchange, and supporting behavior. Measures of lethality were derived from the effectiveness with which the team engaged the targets with live fire. These measures included the team's ability to distribute its fire effectively over time and across its field of fire as well as its ability to hit /kill targets.

Although this was an exploratory study, it was hypothesized, based on past research and theory, that better team processes would be associated with greater lethality. Though the novel measures of teamwork and lethality worked, we found limited evidence of a relationship between teamwork and lethality primarily because there wasn't much variance in the the skill level of the squads that participated (Goodwin, Teo, Anglin, Schreck, 2020).

Battle Drill 2a

The squad attack battle drill may seem simple on paper, but it is in fact a very complex team task that requires considerable practice to master. To understand how this battle drill is executed, it is necessary to understand the organization of an infantry squad. The nine person Army infantry squad is divided into two, four person fire teams (A and B) plus a squad leader (SL), usually an E6 Staff Sergeant. Each fire team is composed of a team leader (TL), usually an E5 Sergeant, an M249 squad automatic weapon (SAW) gunner, a grenadier armed with an M320 grenade launcher (GRD), and a rifleman (RIF). The M249 is the most casualty producing weapon and it is usually assigned to the most senior member of the fire team after the TL, typically an E4 corporal or specialist.

In the squad attack battle drill, the squad has made contact with and is receiving fire from an enemy position with no more than 3 individuals (Headquarters, Dept. of the Army, 2016, ATP 3-21.8, pp. J-8 – J-10). The fire team that first makes contact (e.g., A Team) will take covering positions and return fire (see point 1 in figure 1). This fire team's role is to lay down suppressive fire to pin the enemy down, forcing them to remain under cover. With the enemy position suppressed, the SL will then maneuver with the other fire team (B Team) to a flanking position about 90º to the enemy's left or right relative to the A Team direction of fire (see point 2 in figure 1). The SL decides whether to flank left or right based on terrain.

Once in position, the SL will signal to the A Team leader to shift fire from the enemy position and away from the assaulting team (see dashed arrow near point 1 in figure 1). This is followed by a lift fire (cease fire) command just before the B Team begins the assault (see point 3 in figure 1). After lift fire, the B Team then assaults through the enemy position, followed by the A Team. After securing the perimeter and searching/assessing casualties, the drill is over.

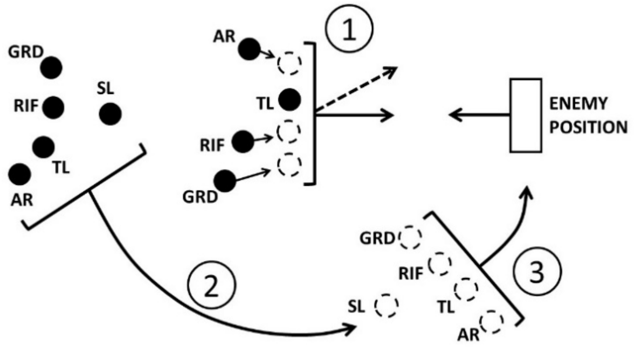


Figure 11. Depiction of the components of BD2A that were part of this data collection. This data collection began at point (1) when the SBF team made contact with the enemy. It continued through point (2) the flanking maneuver of the B Team, and ended at point (3) when the B Team began its assault on the enemy position.

The focus of the live-fire data collection was the performance of the A Team from the moment of contact until the lift fire command was given. This is a period of intense teamwork. Team members must call out contacts and magazine changes and malfunctions. These key pieces of information provide ongoing team situational awareness of who is able to engage targets. Team members use this information to adjust their individual rates of fire insuring all targets are engaged and that a sufficient volume of fire is maintained, a process called *talking the guns*. The TL monitors rates of fire and commands the team to adjust fire as needed. Finally, team members echo key commands from the TL to insure the command is heard all the way down the firing line over the intense sound of live fire.

Translating live training into simulation training.

In this paper we describe how we have translated these battle drill 2a SBF team role and measures developed for live training into a simulation based training exercise using GIFT. Before describing what we did, it is important to understand that going from live to simulation is not a simple apples to apples translation and therefore some creative license was needed as we authored the virtual training exercise.

The limitations of using virtual training for dismounted infantry have been documented (see Knerr, 2007 review). For example, locomotion and tactile interactions with objects in the environment are severely limited and therefore can't be trained. However, cognitive factors like planning, situational awareness, decision-making, and communication can all be trained effectively in a virtual simulator. Virtual simulators can also represent any terrain, time of day, and conditions needed for training.

While it is important for Soldiers to train with live ammunition, live fire training is inherently risky and in order to reduce risks, many controls are in place to prevent rounds from going into populated areas around the training range and to prevent fratricide. These controls unfortunately limit the realism of the training. In this regard, virtual training is more realistic than is possible with live-fire since it has very little risk of accident or injury (Knerr, 2007).

Finally virtual simulators offer units an opportunity to train at a much lower cost. Far less equipment is needed, wear and tear on equipment and ranges is minimized or non-existent and units can typically get many more repititions on the training event per hour in virtual than in live training (Knerr, 2007).

Because virtual simulators are best at training cognitive skills, as will be described in the next section, for this implementation of this SBF task in the Virtual Battlespace (VBS3) virtual simulator, we we focused almost completely on training and assessing cognitive and communication skills.

Methods

In the methods section, we describe our approach to creating the scenario used to train and assess a fire team engaged in a support by fire task. As noted above, we are focused on the role of the support by fire team (A Team) from the time that they make contact with the enemy until the assaulting team (B Team) begins their assault and the A Team is given a lift fire command.

The training scenario was developed to be executed in VBS3. To build this scenario, we developed some new condition classes to automatically assess whether the team is focusing their fire withing the correct sector of fire, to determine whether they are engaging the targets, whether they are maintaining the proper volume of fire, and whether they are pointing their weapons in a safe direction (i.e., not pointing at friendly forces).

These condition classes are aligned with concepts and tasks within the Domain Knowledge File (DKF). The organization of the DKF is summarized in Table 1.

Table 1. Domain Knowledge File Organization.

Task	Concept	Condition Class	team/ individual
React to Direct Fire			
START: OPFOR Engages	Dissiminate 3 D's	*Observed Assessment*	Alpha Leader
STOP: Lift Fire Command	Echo 3 D's	*Observed Assessment*	Team
	Seek cover	*Observed Assessment*	Individual
Provide Support by Fire			
START: Squad Leader Order	Minimize BLUFOR casualities	Health	**Individual
STOP: Shift fire command	Suppress OPFOR with well aimed fire	Engage Targets	Team
	Maintain proper volume of fire	Fireteam Rate of Fire	**Individual
	Stay within assigned sector of fire	Assigned Sector	Team/Indiv
	Prevent weapon safety violation	Muzzle Flagging	**Individual
	Follow rules of engagement	Rules of Engagement	Team/Indiv
	Continue to supress enemy by fire	*Observed Assessment*	Team
	Communicate reloads during fire	*Observed Assessment*	Individual
	Communicate weapon up	*Observed Assessment*	Individual
Response to Weapon Malfunction			
START: Wpn. mlf. start	Maintain proper volume of fire	Fireteam Rate of Fire	**Individual
STOP: Wpn mlf. stop	Malfunction Communicated	*Observed Assessment*	Team/Indiv
	Communicate reloads during fire	*Observed Assessment*	Individual
	Communicate weapon up	*Observed Assessment*	Individual
	Leadership and Directive	*Observed Assessment*	Individual
Shift Fire			
START: B team in position	Confirm BTL shift fire signal	*Observed Assessment*	Individual
STOP: Lift fire command	Instruct team to shift fire	*Observed Assessment*	Individual
	Instruct team to lift fire	*Observed Assessment*	Individual
	Adhere to shift fire commands	*Observed Assessment*	Team/Indiv
	Prevent weapon safety violation	Muzzle Flagging	**Individual
	Stay within assigned sector of fire	Assigned Sector	Team/Indiv
	Maintain proper volume of fire	Fireteam Rate of Fire	**Individual
	Communicate reloads during fire	*Observed Assessment*	Individual
	Communicate weapon malfunctions	*Observed Assessment*	Individual
	Communicate weapon up	*Observed Assessment*	Individual
Lift Fire			
START: Lift fire command	Stay within assigned sector of fire	Assigned Sector	Team/Indiv
STOP: 5 sec timer.	Maintain proper volume of fire	Fireteam Rate of Fire	**Individual
	Adhere to lift fire commands	*Observed Assessment*	Team

** = individual measures roll up into team measures

Before constructing the DKF, it was important to consider how we wanted to structure the tasks and concepts to be assessed. For example, should we use the existing Army training and evaluation outline report for the squad attack battle drill (Headquarters, Dept. Of the Army, 2019)? In this document there are 12 individual tasks and one collective task associated with this battle drill. Though it may seem to be a reasonable way to organize tasks/concepts to be assessed, the problem with the Army's doctrinal breakdown of tasks is that some tend to be at a somewhat higher level than what we want to assess in GIFT. This inconsistency makes it difficult to directly map some standard Army tasks to the DKF.

Another important consideration is that each task needs a discrete start and stop trigger that can be derived from the simulation. Tasks can run concurrently and they can repeat but for the observers' sake, you don't want them to be trying to assess too many tasks at once.

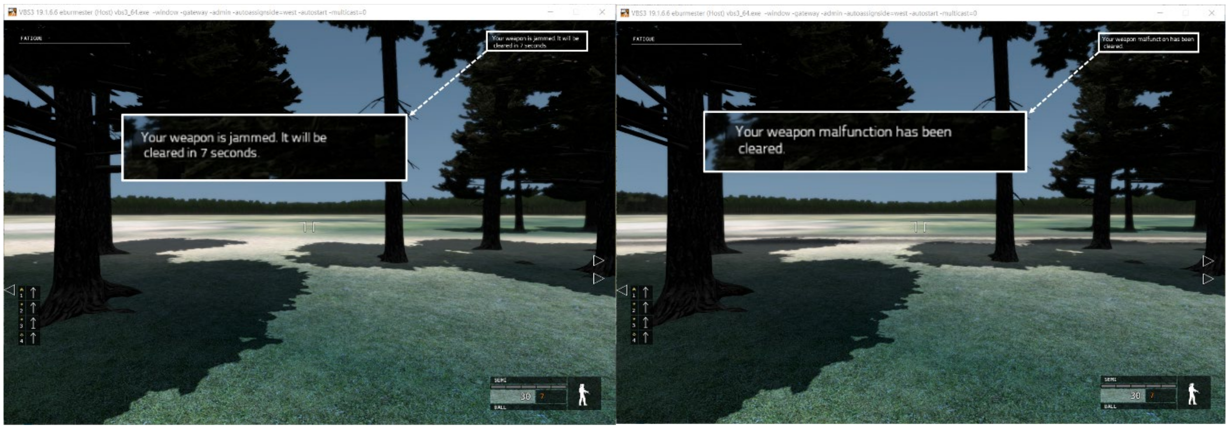
As shown in table 1, we decided to break the DKF into five main tasks, each with a set of concepts. Four of the tasks (react to contact, provide support by fire, shift fire, and lift fire) represent different phases of the exercise. The other task (response to weapon malfunction) occurred repeatedly throughout the exercise and was triggered whenever GIFT injected a weapon malfunction event. Because the tasks aligned with different phases of the exercise, the same concepts appear under multiple tasks because those concepts or subtasks were executed throughout the exercise.

Communications

As noted above, communication is a key performance measure in Battle Drill 2A. Team members need to communicate target information as well as weapon status to each other so that they can collectively suppress the enemy. Ideally, these assessments would be automated, but because GIFT does not have a natural language processing capability yet, we have to rely on human experts to provide these assessments and accurately mark the team for all relevant spoken commands/alerts. This human in the loop observes the trainees in the virtual environment while utilizing the Game Master tool to input their assessments as they happen in real-time or during a replay of the recorded exercise. The good news for human observers is that these communications are the only performance measures they need to make.

Weapon malfunctions

Simulating weapon malfunction in the VBS3 virtual environment posed a challenge in that while VBS3 has scripts available to disable weapon functions such as toggling the safety and weapon discharge, it does not provide the necessary actions for the virtual character to clear weapon malfunctions. To simulate the clearing of a malfunction, a script was developed that resets any disabled functions on the weapon after a preset time delay. In the scenario, the user is presented with a text box informing them that their weapon has malfunctioned and then another text box informs them when their weapon has been cleared. Although this is an admittedly artificial way to simulate a malfunction, it provides all the cues needed for trainees to call out malfunctions as well as indicating when they are back in action. It also has the advantage of allowing GIFT to control the duration of each weapon malfunction (e.g., the M249 takes longer to clear than the M4) and therefore more accurately measure the team members reaction time and communication skills.



VBS3 19.1.6.6 eburmester (Host) vbs3_64.exe -window -gateway -admin -autoassignside=west -autostart -multicast=0

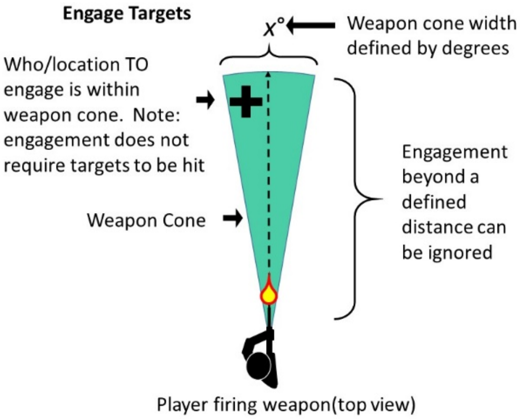
Your weapon is jammed. It will be cleared in 7 seconds.

Your weapon malfunction has been cleared.

Figure 2. Weapon Jam inject into the Battle Drill training environment

Target engagement

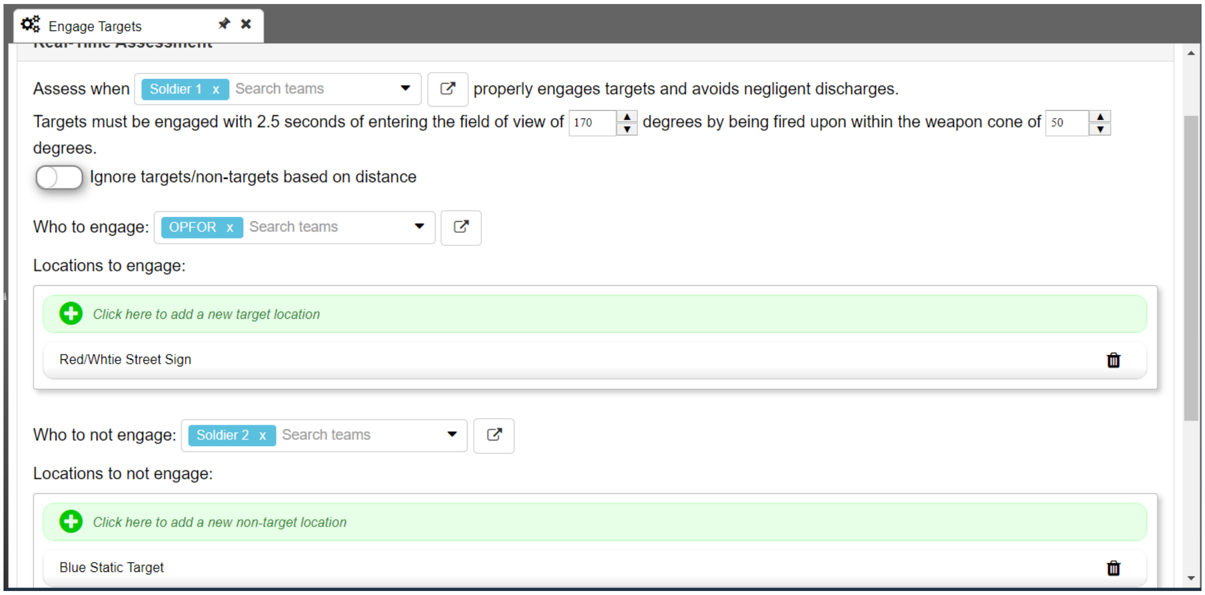
Target engagement is assessed using a condition class the determines whether or not the individual fires their weapon while aiming at a specific target. The player is aiming at a target when that target falls within an arc that is centered on the muzzle of the player's weapon. The size of the arc is configurable and engagement beyond a certain distance can be ignored. This condition class consideres only azimuth offsets and not elevation offsets when determining whether the player is engaging a target. Note that the player does not need to hit the target to successfully engage the target. A target can be an entity or a location. .



Engage Targets
x°
Weapon cone width defined by degrees
Who/location TO engage is within weapon cone. Note: engagement does not require targets to be hit
Engagement beyond a defined distance can be ignored
Weapon Cone
Player firing weapon(top view)

Figure 3. Illustration of the engage target condition class.

The condition class also allows for the configuration of a field of view so that the player is not penalized for failing to engage a target that is outside of his/her field of view. The size of the field of view can be configured in the condition class configuration screen (see figure 4). The player has 2.5 seconds to engage a target once it is in their field of view.



Engage Targets

Real-Time Assessment

Assess when Soldier 1 x Search teams ▾ properly engages targets and avoids negligent discharges.

Targets must be engaged with 2.5 seconds of entering the field of view of 170 degrees by being fired upon within the weapon cone of 50 degrees.

○ Ignore targets/non-targets based on distance

Who to engage: OPFOR x Search teams ▾

Locations to engage:

Click here to add a new target location

Red/Whtie Street Sign

Who to not engage: Soldier 2 x Search teams ▾

Locations to not engage:

Click here to add a new non-target location

Blue Static Target

Figure 4. Configuration screen for the engage targets condition class.

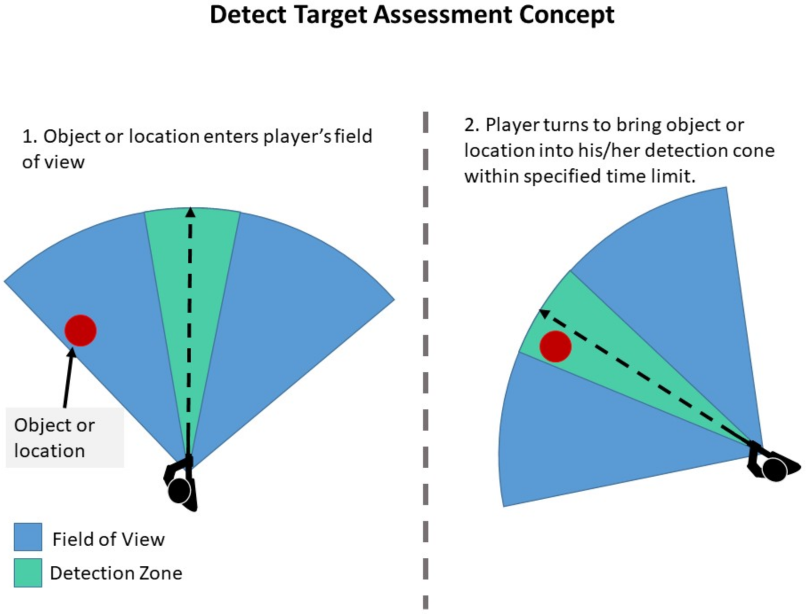

Detect Target Assessment Concept
1. Object or location enters player's field of view
2. Player turns to bring object or location into his/her detection cone within specified time limit.
Object or location
🟦 Field of View
🟩 Detection Zone

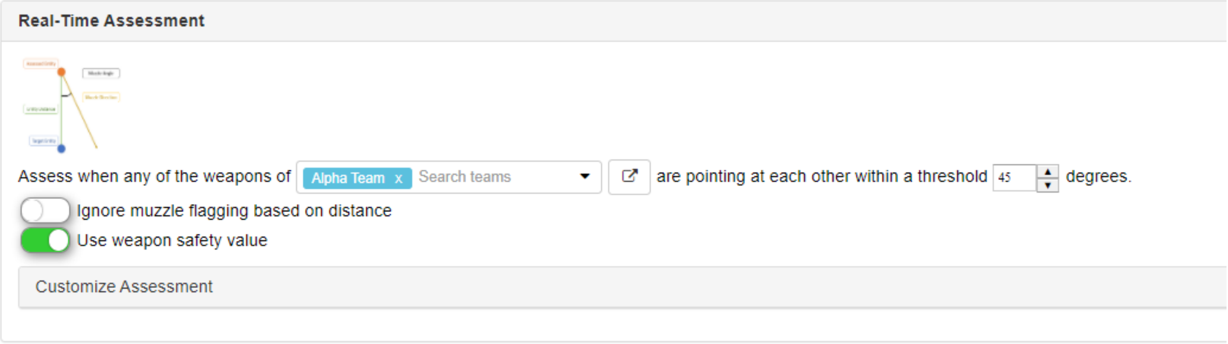
Figure 5. Illustration of how target engagement is assessed after the target is within the player's field of view.

The condition class also allows for the identification of entitites that should not be enaged, for example, team members. Engaging other prohibited friendly or civilian targets is known as a negligent discharge.

These individual assessments can be aggregated at the team level such that if all team members engages the target successfully, the team is assessed as at expectation. If any team member fails to engage successfully or has a negligent discharge violation, the team is assessed at below expectation.

Muzzle flagging

If a player aims his/her gun at another friendly player, this triggers an error known as muzzle flagging. As with target engagement, the player does not need to point directly at the friendly entity to trigger this error. As with the engage target condition class, it is possible to configure the width of a arc around the player's muzzle that defines the flagging zone. The configuration screen for this condition class is shown in figure 6. If any individual on the team violates the muzzle flagging rule, the team is assessed at below expectation. Otherwise, the team is assessed at expectation.



Real-Time Assessment

Assess when any of the weapons of Alpha Team x Search teams are pointing at each other within a threshold 45 degrees.

○ Ignore muzzle flagging based on distance

● Use weapon safety value

Customize Assessment

Figure 6. The muzzle flagging configuration screen.

Sectors of fire

In a scenario that requires a support by fire team, it is imperative that both the support by fire and assaulting team members orient themselves in the appropriate direction at all times to avoid fratricide and to maintain full coverage on the enemy. Measuring a team's ability to maintain the appropriate sectors of fire will be automatically assessed in GIFT by checking whether or not the correct team or individual is oriented in the correct direction when engaging a target.

The sector of fire condition class uses a pre-defined point in the virtual environment to define the center of the sector of fire for a player in a given location. The width of the sector is configurable. Aiming outside of that sector triggers a violation of the sector of fire condition class but brief (less than 2 sec) excursions can be allowed.

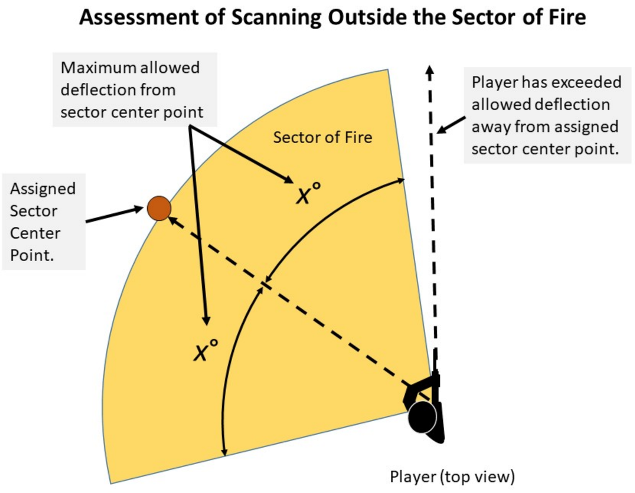
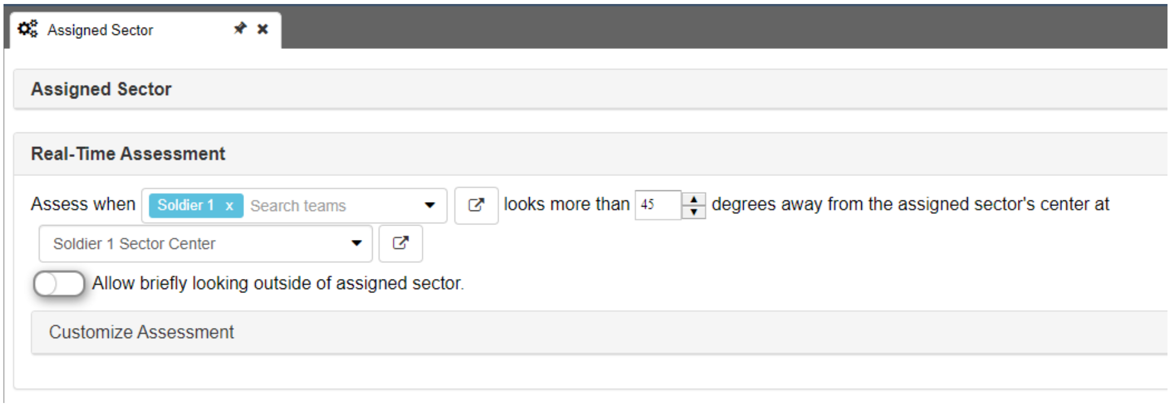

Assessment of Scanning Outside the Sector of Fire
Maximum allowed deflection from sector center point
Player has exceeded allowed deflection away from assigned sector center point.
Sector of Fire
Assigned Sector Center Point.
X°
X°
Player (top view)

Figure 7. Illustration of the sector of fire condition class.

The configuration screen for the assigned sector condition class is shown in figure 8. When assessing a team using this condition class, the team is assessed at expectation when aall team members aim within their assigned sector. The team is assessed below expectation if any team member aims outside of the assigned sector of fire.



Assigned Sector

Real-Time Assessment

Assess when Soldier 1 x Search teams looks more than 45 degrees away from the assigned sector's center at Soldier 1 Sector Center

○ Allow briefly looking outside of assigned sector.

Figure 8. The assigned sector configuration screen.

Talking the guns ratio

As described above, the support by fire team must maintain a consistent overall rate of fire at the enemy location in order to suppress them. When suppressed, the enemy is under cover and can’t return fire or maneuver effectively. To maintain a constant rate of fire, the team members engage in a process known as *talking the guns*. When team members talk their guns, the SAW gunner shoots a burst and then pauses before shooting another burst. This keeps the barrel of the SAWgun from overheating. During those pauses, the remaining team members will increase their rate of fire to fill those gaps. If the SAW gun is down for an extended period of time, for example to reload or clear a malfunction, the M4’s should pick up their rate of fire until the SAW is able to resume shooting.

If done properly, fire teams should maintain a consistent rate of fire throughout the drill such that there are very few pauses in shooting. To develop a quantitative metric, we created what we call the TG ratio.

This ratio looks at the total percent of drill time that rounds are being expended. The TG ratio is calculated using the formula below.

“Talking the Guns” (or TG ratio) = $1 \; - (\frac{Total\ "dead\ space"\ time}{Total\ drill\ time})$

Figure 9 illustrates data from two different fire teams that participated in a live fire training event. The high TG ratio (TG ratio = 0.426) shows a team that effectively maintained a consistent rate of fire. You can see that the team members firing their M4’s increased their rates of fire when the SAW gunner was dealing with malfunctions and reloads. The low TG ratio team (TG ratio = 0.138) shows a team that did not effectively fill in gaps when the SAW gun was dealing with reloads and malfunctions.

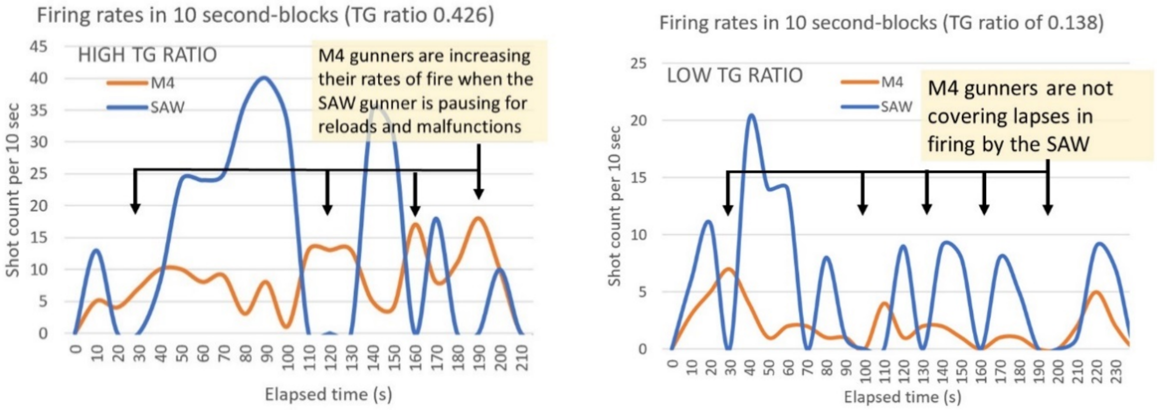
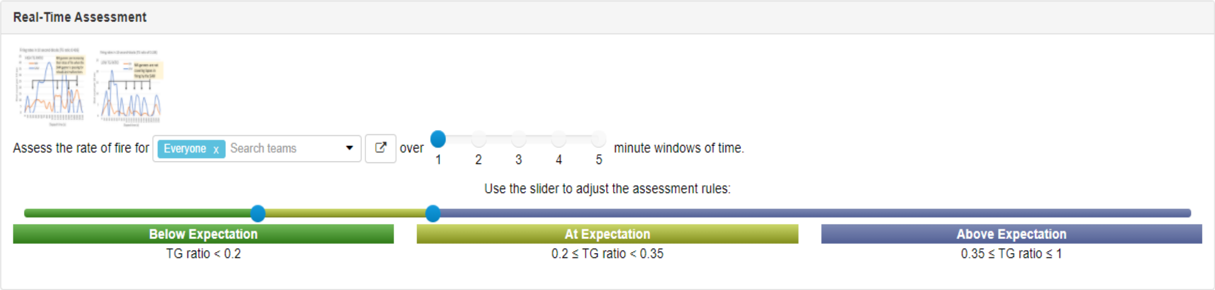


Figure 9. Talking the Guns Condition Class graphical definition

In figure 10 you can see how this condition class can be configured.



Real-Time Assessment

Assess the rate of fire for Everyone x Search teams over 1 2 3 4 5 minute windows of time.

Use the slider to adjust the assessment rules:

Below Expectation
TG ratio < 0.2

At Expectation
0.2 ≤ TG ratio < 0.35

Above Expectation
0.35 ≤ TG ratio ≤ 1

Figure 10. TG Ration configuration screen

Structure of the DKF

Tasks/Concepts/Conditions refresher

GIFT has a number of automated performance measure options, or Performance Nodes, that are available to use with most training applications. A concept is the lowest level performance node and is associated with a java class that contains logic to assess the learner’s actions in the domain. A concept is assessed via conditions. The concept/condition hierarchy supports infinite nesting by specifying the “concepts” choice for a concept instead of “conditions”.

Parent tasks have a lifecycle that must be defined by Start and End triggers, found in the Task Life Span panel when a parent task is selected from the list. Use these triggers to structure the flow of your DKF – the concepts listed under each parent task will only remain active for the duration of the parent tasks' lifecycle. The Start and End triggers are defined using either: 1) the learners location in the simulated environment, 2) the learners completion of a task or concept, 3) the learners performance on a concept, 4) when a Learner Action is selected, or 5) when a strategy is applied.

Team structure

This section defines the hierarchy of teams and team members within a scenario. Both teams and team members can be referenced in various parts of the DKF, such as strategies and conditions. This is beneficial when assessing multiple teams at once, as well as assigning assessments or strategies to a specific team or team member to separate responsibilities.

Strategies

GIFT has a number of adaptation options that can be automatically applied by the system when a learner state is identified based on the defined parameters. These are called "Strategies" in the DKF architecture. Strategies are sent from the Pedagogical module and implemented by the Domain module based on changes in learner state.

Results

A location was chosen for this scenario that has a relatively open area near a wood-line (see figure 11). In the scenario, the assaulting squad approaches the open area where it encounters the enemy in the field and begins to engage. The human players in the scenario are only in the support by fire element and AI squadmates carry out the assault. This location is found within Fort Stewart in the VBS3 terrain database.

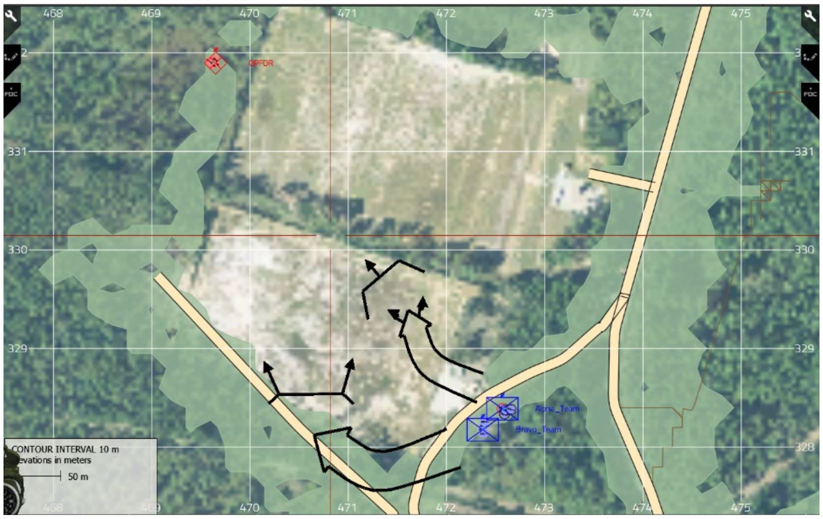


Figure 11. Map view of the scenario location in VBS3

While the scenario is playing out, the human observer sees the view below displayed in the game master interface (see figure 12). On the left is an overhead map/satellite image with the locations of players and AI forces superimposed. On the right are the tasks and associated concepts to be assessed.

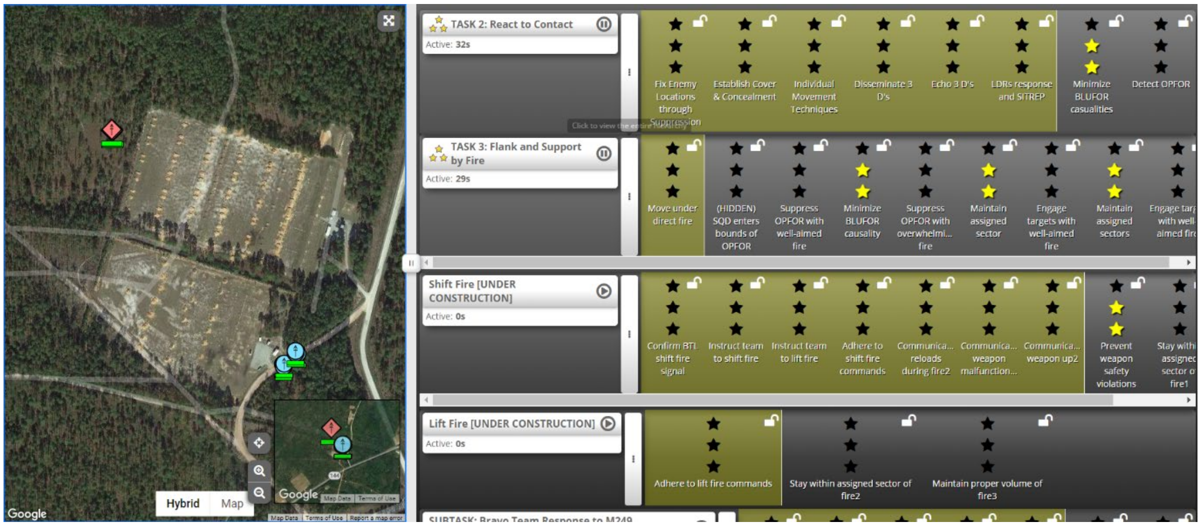

TASK 2: React to Contact
Active: 32s
Fix Enemy Locations through Suppression
Establish Cover & Concealment
Individual Movement Techniques
Disseminate 3 D's
Echo 3 D's
LDRs response and SITREP
Minimize BLUFOR casualties
Detect OPFOR
TASK 3: Flank and Support by Fire
Active: 29s
Move under direct fire
(HIDDEN) SQD enters bounds of OPFOR
Suppress OPFOR with well-aimed fire
Minimize BLUFOR causality
Suppress OPFOR with overwhelmi... fire
Maintain assigned sector
Engage targets with well-aimed fire
Maintain assigned sectors
Shift Fire [UNDER CONSTRUCTION]
Active: 0s
Confirm BTL shift fire signal
Instruct team to shift fire
Instruct team to lift fire
Adhere to shift fire commands
Communica... reloads during fire2
Communica... weapon malfunction...
Communica... weapon up2
Prevent weapon safety violations
Lift Fire [UNDER CONSTRUCTION]
Active: 0s
Adhere to lift fire commands
Stay within assigned sector of fire2
Maintain proper volume of fire3
SUBTASK: Bravo Team Response to M249
Hybrid
Map
Google

Figure 12. Game master display

Communications

As described above, the communications are assessed by the human observer. Communications include disseminating the 3 D's when reacting to enemy contact, announcing weapon malfunctions and magazine changes, and repeating leader orders. In Figure 13 below, you can see some of these assessments represented in Game Master. Notice these assessments, for example "Disseminate 3 D's", are inside a tan/gold background – this is intended to draw the Observers attention to assessments that require manual entry.

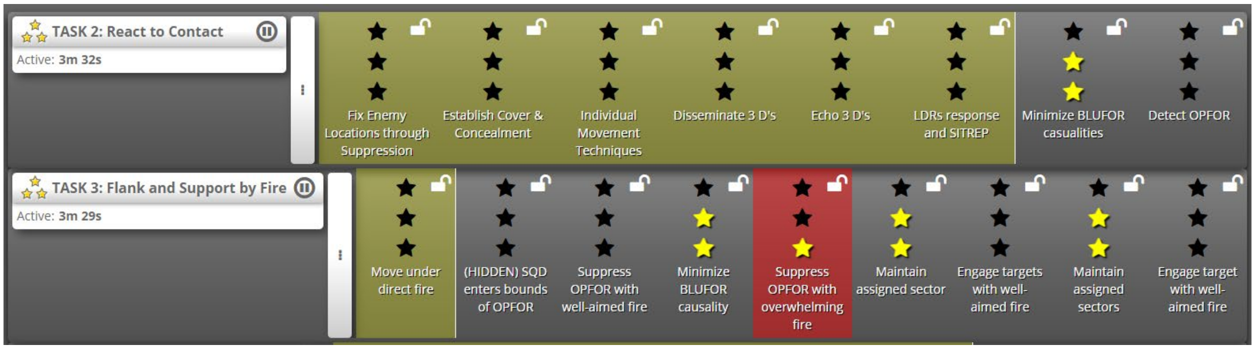
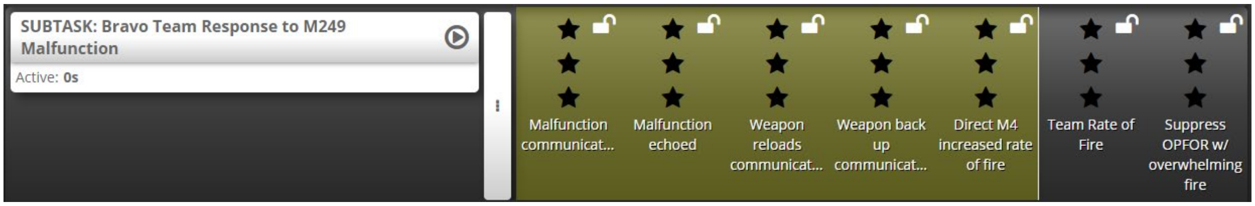

TASK 2: React to Contact
Active: 3m 32s
Fix Enemy Locations through Suppression
Establish Cover & Concealment
Individual Movement Techniques
Disseminate 3 D's
Echo 3 D's
LDRs response and SITREP
Minimize BLUFOR casualties
Detect OPFOR
TASK 3: Flank and Support by Fire
Active: 3m 29s
Move under direct fire
(HIDDEN) SQD enters bounds of OPFOR
Suppress OPFOR with well-aimed fire
Minimize BLUFOR causality
Suppress OPFOR with overwhelming fire
Maintain assigned sector
Engage targets with well-aimed fire
Maintain assigned sectors
Engage target with well-aimed fire

Figure 13. Game master view of communications assessed by a human observer.

Weapon Malfunctions

The weapon malfunction events in this scenario effectively assessed the squads ability to react and communicate their weapon status to their team in a timely manner. While the malfunction itself was applied automatically by GIFT, the human in the loop had to manually enter the assessments into the system based on the dialog they heard from the squad. Figure 14 shows the assessments associated with a weapon malfunction as it was displayed in Game Master.



SUBTASK: Bravo Team Response to M249 Malfunction

Active: 0s

Malfunction communicat...
Malfunction echoed
Weapon reloads communicat...
Weapon back up communicat...
Direct M4 increased rate of fire
Team Rate of Fire
Suppress OPFOR w/ overwhelming fire

Figure 14. Game master view of communication concepts to be assessed by a human obeserver.

Discussion

This scenario has been developed as a simple training tool for small units executing team training. It also has led to the development of new condition classes, one of which was derived from data collected at a live-fire training event. The team training is focused on cognitive skills and can be delivered over a set of networked laptop computers. This training scenario also serves as an example of how team training assessments can be implemented in GIFT using a combination of both automated and human observer based measures.

As noted in the introduction, the focus of this training is on cognitive procedural team tasks. By delivering the training in a virtual environment, it is possible for trainees to do a large number of repetitions of the task in a short time period. If simulated weapons were available though it would be possible to train some of the motor tasks associated with this scenario, but that would increase the cost of delivering the training and therefore would limit the availability/throughput of the training.

Ideally this type of training solution would exist in multiple modes so that it could be used as both a cognitive skills trainer and a cognitive plus motor skills trainer depending on the facilities available. The Army does use some marksmanship simulators like the Engagement Skills Trainer (EST) or the Squad Automated Marksmanship Trainer (SAM-T) that could be used to train support by fire scenarios that would include training on motor skills such as shooting, reloading, and clearing weapon malfunctions.

There are refinements and improvements to some of the measures that could be implemented. For example, we do not currently measure or report the shooting accuracy of individual team members. This decision was done in part because aiming and shooting is accomplished with the mouse and keyboard so such measures would probably not relate to the trainees' actual marksmanship skills. Nevertheless, it is possible that such feedback might increase the motivation of soldiers to use the training and if the scenario were to incorporate simulated weapons, these measures could be a more valid reflection of marksmanship skills.

Another area for improvement is in the behavior of the computer controlled enemy combatants. Currently their behavior is very simple. They will engage the players when they players approach them. What we would ideally like is for the enemy combatants to react to the suppressive fire. The rounds fired from M4 and M249 rifles are supersonic and so as those rounds pass by, you hear an unmistakable snap caused by the shockwave. Most people react by immediately dropping and taking cover and will stay under cover as long as they hear rounds passing by. It shouldn't be difficult to have computer controlled enemy combatants react realistically to that kind of suppressive fire.

The training value of this would be that if a support by fire unit had long pauses in shooting because they were not coordinating their fire effectively, then the enemy would have opportunities to come out from cover and start to return fire. This would provide immediate feedback to the trainees in that they might suffer casualties themselves and could even be suppressed themselves by the enemy. This kind of feedback would help to reinforce the importance of communication and coordination to make sure that they collectively maintained a consistent rate of fire

Finally, this scenario could be improved by expanding it to include the whole squad and the whole squad attack battle drill. This would add even more value as it would allow the whole squad to train and learn to coordinate their actions throughout the battle drill. Expanding the scenario in this way would mean making a much more complex DKF that assessed both fire teams as well as the entire squad.

References

Goodwin, G.A., Anglin, K., Teo, G., Schreck, J. (2020). Teamwork and lethality in a support by fire team. (paper 20309). Interservice Industry Training and Simulation Conference. Orlando., Fl.

Headquarters, Department of the Army (2016). ATP 3-21.8 Infantry Platoon and Squad. https://armypubs.army.mil/epubs/DR_pubs/DR_a/pdf/web/ARN13842_ATP%203-21x8%20FINAL%20WEB%20INCL%20C1.pdf

Headquarters, Department of the Army (2019). Training and Evaluation Report, Conduct and Attack - Squad

Knerr, B. W. (2007). Immersive simulation training for the dismounted soldier. (Study Report 2007-010). Alexandria, VA.: U.S. Army Research Institute for the Behavioral and Social Sciences.

ABOUT THE AUTHORs

Gregory Goodwin *is a senior research scientist and acting Team Leader at the U.S. Army Combat Capabilities Development Command – Soldier Center (CCDC-SC) at the Simulation and Training Technology Center (STTC) in Orlando, Florida. After completing his Ph.D. at the State University of New York at Binghamton in 1994, Dr. Goodwin completed post doctoral work at Columbia and Duke Universities and was an Assistant professor at Skidmore college. In 2005, Dr. Goodwin began working for the Army where he has been examining ways to leverage technologies to reduce the cost and improve the effectiveness of training.*

Elyse Burmester *is a Test Engineer at Dignitas Technologies. Since 2017, she has contributed as a member of the GIFT development team in addition to directly supporting the Creative and Effective Training Technologies Branch at U.S. Army Combat Capabilities Development Command - Soldier Center (CCDC-SC) SFC Paul Ray Smith Simulation & Training Technology Center (STTC). Ms. Burmester has a bachelor degree in Political Science from Florida Gulf Coast University.*

Theme IV: Collective and Team Tutoring

Designing a Distributed Trainer Using GIFT for Team Tutoring in Command Level Decision Making and Coordination

Randy Jensen[1], Bart Presnell[1], Jeanine DeFalco[2], and Gregory Goodwin[2]
Stottler Henke Associates, Inc.[1], U.S. Army Combat Capabilities Development Center (CCDC)[2]

Introduction

The increasing use of distributed collaboration environments for collective activities has changed the landscape for how teams must perform, and consequently how they train. Although the demand for technologies supporting remote collaboration has dramatically accelerated due to recent pandemic conditions, the good news is that the trajectory of prior research has produced models, methods and tools for development to address these growing needs. This paper discusses design factors in building a distributed team tutor to provide adaptive training in both teamwork and taskwork, in the context of scenario-based exercises requiring synchronous collaborative team performance. These design considerations are explored using an example application under development, which focuses on training team decision-making and coordination for Army command staff collaborating on the analysis and development of military courses of action (COAs). Although there are noteworthy team performance modeling considerations specific to this domain, the focus of this paper is primarily on the structural design of the distributed team trainer and how it integrates with existing tools. While this paper reports on early stage work and plans for building functionality with the Generalized Intelligent Framework for Tutoring (GIFT), the purpose is to share design concepts with the GIFT community. Ultimately this design may share common features with training needs for other domains.

One way to breakdown the functional elements in a distributed team trainer is to group them into three categories: (i) session and profile management, (ii) the operational, decision-making and communication environment, and (iii) instructional modeling, assessment, and feedback. Session and profile management refers to session status tracking, profile records for individual participants, and team and sub-team composition. The operational environment is where material is presented and exercises are performed, so it includes simulation interfaces specialized to the domain. In our example application this refers to a planned collaborative exercise interface with tactical maps, overlay tools, unit hierarchies, and other tools specific to the war gaming process. The environment also includes collaborative communication tools like chat rooms or other mechanisms to support interactions between remote participants, which may be either general purpose or specialized to the exercises. Instructional modeling, assessment, and feedback are all essential for providing a learning experience where teams receive direct tailored feedback on their performance working together. GIFT offers a reusable framework for building intelligent tutors, which provides models and practices to help with this category of training system elements (Sottilare et al., 2012). Several recent projects have implemented approaches configuring the GIFT Domain Module for team training, either by aggregating individual performance factors for their indications at the team level (Gilbert et al., 2018), or by evaluating performance of collective teams (McCormack et al., 2019). This paper discusses the design of a distributed team trainer integrated with GIFT intelligent tutoring infrastructure, to be deployed in a browser-based setting.

TEAM TRAINING DOMAIN

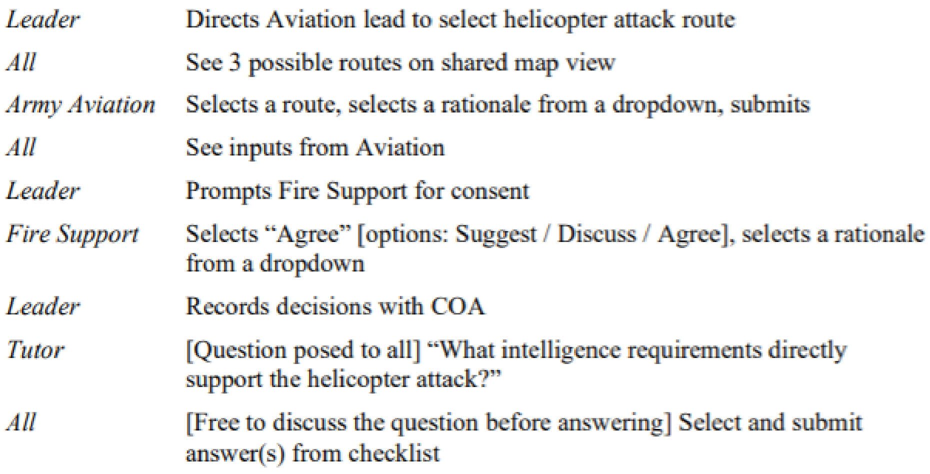
The application used as an example throughout this paper is a distributed team trainer under development in an effort for the U.S. Army Combat Capabilities Development Command. The effort is called Reusable Automated Assessment and Feedback for Teams (RAAFT), and the objective is to develop automated assessment mechanisms for team training with the goal of reusability in several forms – across scenarios, across platforms, and potentially across domains. Many constructs of teamwork inherently apply to different operational and training settings, so there is the prospect to develop consistent reusable methods grounded in instructional science. For the RAAFT effort the approach to reusability is to start with prototype development in a specific team training domain and then generalize assessment mechanisms from the initial application for reuse.

The team domain selected for the initial prototype under development is command staff war gaming conducted at the division level. War gaming is a clearly delineated step in the Army's Military Decision Making Process (MDMP), involving a deliberate group analysis of one or more courses of action. And yet, although MDMP processes are defined, there is also an art to effective war gaming, which specifically includes factors related to the human dynamics of teamwork. The literature in dimensions of teamwork has provided a number of effective general purpose breakdowns, often with common themes (Johnston et al., 1998; Kozlowski et al., 2015; Marlow et al., 2018; Salas et al., 2005; Sottilare et al., 2018). The model of teamwork to be used for the trainer includes six dimensions relevant to war gaming: Leadership, Supporting Behaviors, Information Exchange, Communication Quality, Team Cognition, Team Orientation. These six dimensions are the underpinning for team assessment in the distributed trainer, which is designed to use both declarative and observational measures. The initial prototype will primarily use declarative measures, which involve directive questions posed to all participants at predefined steps in the war gaming process. There are several categories of questions, to probe different elements of team-level war gaming effectiveness, such as shared mental models, understanding of roles, and team orientation. The observational measures to be developed subsequently will monitor the decisions, actions, and behaviors of participants during war gaming, which can be treated as teamwork indicators for automated assessment purposes. Given this order for planned system development, this paper focuses primarily on the first part – design of the initial declarative assessment measures.

Planned Training Experience

The use case envisions a training setting where all participants are in different physical locations, so primary interactions are in the browser-based environment, potentially augmented with videoconferencing or other live communications channels. However, co-located exercises are also allowed; it is not required that participants be remote. So co-located teams may also use verbal communications or other means. From a training perspective, the primary requirement in this regard is that official decisions and inputs are expressed in the environment, to make data available for assessment.

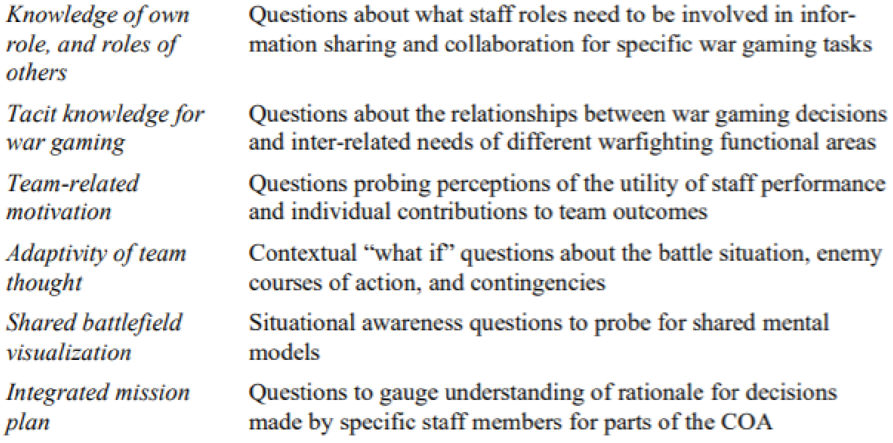
To illustrate, the following is an example vignette for the kinds of interactions envisioned in an exercise scenario as designed for the trainer. Participants include a Leader and five other staff members representing different warfighting functional areas such as Intelligence, Maneuver, Aviation, Fire Support, and Logistics. There is a status board showing the list of participants with symbology to indicate their status on current activities, such as any decision inputs in progress. Participants all see a shared tactical map view, and all have context-sensitive toolboxes based on their roles and the COA events currently being considered. The Leader has unique tools for controlling the flow through events in the COA. There is also a shared interaction panel for chat messages, tutor messages, and records of decisions from the war gaming process. In the example sequence below, midstream in the analysis of a COA, the team is considering a step that involves a planned helicopter attack.



Leader	Directs Aviation lead to select helicopter attack route
All	See 3 possible routes on shared map view
Army Aviation	Selects a route, selects a rationale from a dropdown, submits
All	See inputs from Aviation
Leader	Prompts Fire Support for consent
Fire Support	Selects "Agree" [options: Suggest / Discuss / Agree], selects a rationale from a dropdown
Leader	Records decisions with COA
Tutor	[Question posed to all] "What intelligence requirements directly support the helicopter attack?"
All	[Free to discuss the question before answering] Select and submit answer(s) from checklist

In this vignette, the Tutor question posed to all is a declarative assessment relating to shared mental models and an understanding of team roles. Throughout the exercise, the questions directed to participants are tailored so that the net combined effect is that they cover a cross-section of teamwork dimensions, while each question is also contextualized to scenario events. The observational assessments in this sequence would relate to team process concerns, such as information exchange between different roles (Aviation and Fire Support) before deciding on a route, the potential for supporting behavior in selecting the route, and the role of the Leader in facilitating this teamwork. The sequence above depicts a nominally effective flow of events, but the intention is to allow for possible teamwork errors. For example, an observational assessment would detect a situation where the Leader concludes the team review of the helicopter attack event, without Fire Support having given input on the route selected by Aviation.

The declarative assessment mechanism is a focus for initial development, and is planned to make use of GIFT survey assessment functionality. The declarative assessment questions are adapted from a set of assessments described by Cianciolo and Sanders (2006) in a conceptual framework for determining war gaming effectiveness. Questions fall into the following categories, specifically oriented toward teamwork in the war gaming process.



Knowledge of own role, and roles of others	Questions about what staff roles need to be involved in information sharing and collaboration for specific war gaming tasks
Tacit knowledge for war gaming	Questions about the relationships between war gaming decisions and inter-related needs of different warfighting functional areas
Team-related motivation	Questions probing perceptions of the utility of staff performance and individual contributions to team outcomes
Adaptivity of team thought	Contextual "what if" questions about the battle situation, enemy courses of action, and contingencies
Shared battlefield visualization	Situational awareness questions to probe for shared mental models
Integrated mission plan	Questions to gauge understanding of rationale for decisions made by specific staff members for parts of the COA

The categories are not necessarily exclusive, so some questions may be formulated to relate to more than one category. Some categories entail questions that are more context-sensitive than others. For example, questions relating to the integrated mission plan are intended to measure a factor of Team Cognition, by looking at the degree to which all team members understand the rationale for staff decision inputs in the war gaming process. Using the example in the earlier vignette, a question might be used to verify that there is understanding throughout the team for the Army Aviation lead's rationale for picking a certain route for the helicopter attack. In this case, both the question and the expected answer are contextualized to the decision input from Aviation. In more general terms, questions of this nature are configured based on the inputs from different staff members for different COA events. So one challenge in this regard is to deliver directive questions that are contextualized to the exercise flow by design and representation. Another challenge is to design the reasoning that takes the aggregate collection of answers from participants, and draw conclusions about teamwork, in a model that uses GIFT structures for team member roles and hierarchy. These goals are discussed further with the structural design below.

STRUCTURAL DESIGN

Since this paper discusses early stage work, the focus is primarily on design factors and plans for supporting the envisioned functionality in the distributed trainer and its integration with GIFT. Especially in a distributed architecture, there are questions about which "side of the fence" each element of the trainer functionality will be implemented on. That is, how features may be supported by existing utilities in GIFT, or supported with adaptations or extensions, or implemented as capabilities in the trainer environment external to GIFT. The overall design of the distributed trainer is broken down by the three functional areas discussed earlier: (i) session and profile management, (ii) the operational, decision-making and communication environment, and (iii) instructional modeling, assessment, and feedback. These functional distinctions are made for convenience in describing elements in a server-based architecture for synchronous exercises involving teams of remote participants. However, while these boundaries relate to distributed training for the sake of discussion, they are not intended to carry any special weight as a unique alternative organizational scheme compared to other designs. At the highest level, the training system design assumes there is a RAAFT server, a GIFT server, and browser-based clients for participants. Much of the complexity lies in defining the data flow between these components. This section lays out design thoughts for building this interoperability.

Session and Profile Management

GIFT provides existing utilities that help with session and profile management at the server, which can be used for outer loop coordination of distributed team members. Since the focus for initial implementation is on the training functionality within team exercises, the main purpose of outer loop functionality in this architecture is to establish the exercise coordination necessitated in the distributed setting. The initial design leaves out outer loop questions relating to the choice of courses or exercises. Inner loop session and profile management functionality is shared by parallel RAAFT and GIFT modules.

The planned design is that users first go through the GIFT Lobby as the precursor to an exercise, where they are given the opportunity to create a new session or join an existing session. By policy, the person designated the leader is responsible for creating a session. Once users are part of a session, they are prompted to select an unoccupied role.

Roles are defined using the team hierarchy organization in GIFT. For the war gaming application, the hierarchy of roles is flat. That is, the team composition is a simple collection of roles without nested subteams. So the exercises are defined with one Team level, and TeamMember nodes for each of the participant roles, in this case corresponding to the different warfighting functional areas.

The creator of the session is responsible for starting an exercise, which leads to two triggers. First, the RAAFT server is informed of a new session, and the playerIds and roles associated with each participant. Also the client user interface for each participant is started and informed of the session it is joining. After the client starts up, it connects to the RAAFT server supplying the playerId and session id. The RAAFT server relays updates to the connected clients, which display participant status and also show any expected connections that are still missing.

Inner loop session management related to the tracking of user state information within an exercise is shared between modules on the RAAFT and GIFT servers. Both systems maintain session information. GIFT maintains user session information to use with the Tutor Module and assessment. RAAFT maintains a parallel session profile with detailed records of what actions and decisions have been made in the exercise. These records can be used to restart an incomplete session or replay a scenario. This detailed session information is maintained in a JSON database, which is consistent with the planned method for communication data flow between the RAAFT server and client. Reusing the same structure for session management and persistence saves significant effort from using a table based approach.

There is a related area of interest involving the question of how to handle a persistent team model, as a team counterpart to an individual learner model. Although the current effort is initially focused mainly on inner loop, within-exercise team training, there will likely be a need for future work to define practices for preserving models of team competencies and performance, as teams are directed through a course involving multiple exercises, or even multiple courses. Teamwork factors such as Team Orientation may be difficult to represent and persist at the team level, when considering the possibility of changes in team composition (who's participating in each event) or variations in experience levels among different participants. Ideas for building support for a Team Model as part of GIFT have begun to be explored by Gilbert et al (2018) and others.

Operational, Decision-Making and Communication Environment

The main elements of the environment for distributed team exercises are the front-end client-side user interface and the back-end RAAFT server which manages data flow both to the front-end and to the GIFT server.

Client User Interface Framework

The design of the client side interface where participants interact with the trainer mainly involves the selection of a user interface framework that works readily for the needs of a distributed trainer. We discuss two frameworks – Unity and Angular – evaluated on four criteria: applicability to domain, ease of development, GIFT integration, and flexibility. Some of the benefits of each are enumerated, along with the initial selection of Angular as the framework to pursue based on its overall applicability to the domain, as well as its flexibility in being able to add additional collaborative tools and visualizations.

Unity is a widely used game development platform, originally intended for simplifying the process of developing 3D games. The application has expanded to support 2D games, as well as a large variety of applications for different industries. Unity has an editor for creating screens, and scripting is performed in C#. Unity applications are authored in the editor, and then can be exported to run on a variety of platforms, including as a WebGL application. Unity was considered for two reasons. First, it is a well established tool that carries significant resources and advantages for quickly developing a sophisticated application. Second, Unity WebGL has already been embedded within GIFT.

The second package we considered was Angular. This is one of the standard frameworks for building a web application. Angular unlike Unity was not intended for games, but is instead intended for general purpose applications. Angular is a JavaScript based framework, which allows it to be connected to a number of JavaScript based packages to support a variety of displays and visualizations. Angular was considered for two reasons. First, like Unity, Angular provides a number of tools and resources for rapid development.

Second, Angular provides access to a number useful display packages such as map displays and timeline displays. We compared the two frameworks based on four criteria: applicability to domain, ease of development, GIFT integration, and flexibility. For applicability to domain, we judged Angular better. The proposed application is more focused on graphical displays, and standard interface controls such as buttons and menus, rather than fast moving objects, which lends itself more to Angular than Unity. For ease of development, we felt both were comparable in terms of development support. For integration, we gave a slight edge to Unity, as there is already support for Unity WebGL being embedded within GIFT. However, much of this framework could be reworked slightly to support an Angular client. We consider Angular to offer more flexibility. Both platforms have significant flexibility, however Angular seems to offer more possible extensions or services relevant to this application.

The aim is that the in-exercise experience is managed by Angular, which has support for all elements of the user interface design, including the shared tactical map view, role-specific toolboxes, the status board showing participants, and the shared interaction panel. The initial plan is to use the existing GIFT Tutoring UI for tutor interventions and directive survey questions. This may be more tightly integrated with the Angular UI over time, if this leads to a more cohesive experience.

There are two other types of information that will be updated to the user regularly and make sense to consider placing in the same UI element. First, the design treats the team leader as a special role in each war gaming exercise, not only as a participant but also as an exercise controller. Thus leaders are given process cues to help in their role facilitating team execution of the war gaming process. These cues are different from assessment feedback, and other participants have no firsthand view of the system cues provided to leaders. We intend to explore from both a user perspective and an implementation perspective whether to deliver process cues to leaders in the same user interface element as tutor interactions and feedback. In a similar vein, another element of the initial user interface design is a visual timeline to depict the flow of the war gaming process and decisions made along the way.

RAAFT Server

The RAAFT server is built on Node.js to simplify connecting between the front-end client and the backend, as both systems can be built with Typescript. The RAAFT server and the GIFT Cloud server are connected via the GIFT Gateway Module and a RAAFT Interop Plugin. The connection between the two servers is based on REST protocols.

When an exercise starts, the GIFT server manages the lobby allowing participants to join the session and select their roles. The Interop Plugin informs the RAAFT server to start a new exercise session and the players-role assignments, which triggers the client sessions. Once the Angular client sessions start, they only communicate with the RAAFT server. The primary communication is a REST interface that passes player actions, and then receives updates from the server for state changes. An auxiliary connection based on websockets is needed to support real-time chat communication. The RAAFT Server will communicate actions available to each player. Updating controls based on player suggestions and decisions, as well as updates based on the state of the war-gaming process.

The Angular client could be configured to communicate with GIFT via JavaScript methods from the GIFT Tutor code, but the current design does not use this functionality for two reasons. First, the aim is to allow the client to function outside of GIFT if needed. Second, the RAAFT server is intended to com- municate teamwide actions and states to GIFT, rather than having individual communications from each client. However, tradeoffs for this design decision will continue to be explored.

Assessment Methods

For the functional area relating to instructional modeling, assessment, and feedback, our current focus is on assessment methods, specifically for measures of teamwork in distributed war gaming exercises. The initial assessment measures to be implemented will be those associated with the declarative questions that are injected during the war gaming process. The current design for these assessments is discussed in more detail below, in particular to address the question of how some questions will be contextualized to the exercise. The observational assessments will be more complex to implement, as they involve measures that monitor team members' behaviors. For example, complexity is added by the possibility of supporting behaviors, where observable actions can inherently be carried out by different individuals rather than a particular expected person, and still be considered effective teamwork. Although the design details for the observational assessments are not addressed in this paper, there is promising work that may assist with this added complexity. Folsom-Kovarik & Sinatra (2020) describe an approach extending GIFT to associate both Roles and Responsibilities with TeamMembers, to allow for more complex team relationships. For both the declarative and observational assessments, the RAAFT design anticipates that the mappings between team roles and logic to be defined in the GIFT Domain Module are straightforward, partly because of the flat hierarchy of roles for war gaming. This may make it possible to limit the number of Domain Knowledge Files (DKFs) that need to be created, so that there is one for each participant, and one for the team as a whole, but not the combinatoric expansion of DKFs for every n-wise subset of participants. For a team of 6, this means 7 DKFs rather than 63.

Declarative Assessments

One area of complexity in designing declarative assessments involves the generation and evaluation of questions that are contextualized to the decision inputs from participants during the exercise. Referring back to the helicopter route example, consider a question posed to all staff members to determine if they all understand the Aviation lead's rationale in selecting a particular route. The correct answer depends on the inputs previously given by the Aviation lead, and cannot be scripted in advance.

The initial version of the RAAFT server is designed to maintain a data structure with scenario information that includes decision elements involved in the analysis of each COA. Information about the decisions to be made, the roles involved, and the possible rationale factors are all defined in scenario data. So the contextualization is mostly a matter of referencing these items. This scripted format allows participants to gain practice in communication and teamwork with a facilitated war gaming process. This structure does require manual authoring initially, but it also allows us to provide GIFT with updates that are about generic teamwork actions and states that are separated from the domain specifics of the scenario.

The questions to be used as declarative assessments are designed with a representation to capture several features for effective use during an exercise:

- Metadata to identify how questions fit to the context of COA scenario events / conditions
- Categorization in terms of teamwork dimensions, so that the net effect of questions posed throughout the exercise is to have adequate coverage of the range of dimensions
- Representation of the nature of expected answers (e.g., multiple choice, checklist, ranking, etc.), to configure how the questions are presented and scored

GIFT provides different survey instruments, which may apply for different teamwork measures. Multiple choice survey instruments are suited for many questions, relating to adaptivity of thought, shared mental models, and integrated mission plans. Checklists apply for questions relating to the understanding of team roles and Information Exchange, for example to identify roles needing information in certain steps of the

COA. Questions expecting answers in the form of a ranking can be used for assessments in the team-related motivation category. These relate to teamwork factors like Leadership or Team Orientation, where the goal is to measure perceptions of the importance or effectiveness of certain team processes. In some cases the goal is to synchronize questions with information to be displayed in the client. For instance, situational awareness questions may query participants about key terrain associated with events in the COA. The simplest approach is to label terrain and provide multiple choices for answers referring to the labeled terrain features by name. A future approach might allow participants to answer by directly clicking on the map. Some questions are assessed in terms of correctness, while others related to team processes are assessed more in terms of consistency within the team. For example, if the Aviation lead chooses a certain route for certain reasons, then a question aiming to assess a shared team understanding of the rationale is looking less for an absolute justification for the route selection, and more for consistency reflecting team members' understanding of the underlying intention.

When a declarative question will be injected, the RAAFT server selects a question template from a library of survey instruments, each containing its own representation of information needed. Based on the current exercise state, configuration information is sent as a survey request through the Gateway Module, to the GIFT Tutor Module and presented. Participant inputs are then captured and processed for what they indicate about individual and team measures.

There are two additional design needs to be considered for the declarative assessments and the use of GIFT surveys. First, in the cases where assessment is a function of consistency within a team, there is a need that scoring should not be completed until all members of the team complete the survey. This is an open area to explore whether this can be accommodated with the standard GIFT survey, which is intended to either just collect information on a single user, or is scored on the basis of correct or incorrect answers. The second feature to consider involves runtime synchronization of the tutor UI during team-wide surveys. GIFT has the concept of teams, and has functionality to coordinate the launch of an external training application. However, we are exploring the best way to design support for synchronizing the pausing of the external training application while all team members take the survey, and then maintaining the paused state until all members of the team complete the survey.

One approach to these needs is to handle both within the RAAFT server and the Interop Plugin. Teamwide scoring involves three steps. First, the surveys themselves are marked as non-scored surveys, to stop GIFT from trying to score the results, so that results are just stored in the User Management System. Second, the Interop Plugin is configured to receive a message carrying the TUTOR_SURVEY_QUESTION_RESPONSE. This provides a means to monitor as players respond to each question. The responses can then be forwarded to the RAAFT server which collates responses from the team as a whole. Once all members of the team have completed a question, the RAAFT server calculates the appropriate metrics for teamwork scores, which are sent to GIFT as a state update to the Domain Module. The synchronization of pauses during surveys is handled in a similar manner. GIFT sends messages to the RAAFT server, which are relayed to Angular clients, to pause and unpause each client as they start and complete a survey. The messages to pause the client are sent immediately, but the unpause messages are reserved until all team members in a session are prepared to continue (or some other state is reached, such as a timeout or leader override). Alternative approaches could involve changes within the standard GIFT modules to support this functionality. Although the approach on the RAAFT side may be more manageable as an initial implementation, there may be value in the future to expanding GIFT survey functionality in several ways to support more team related situations like these.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

The discussion in this paper shares the RAAFT project design thoughts for using GIFT to develop a distributed team trainer for war gaming, while acknowledging that in its early stage of development there are few specific lessons-learned to offer to the community. Future work will lead to more concrete findings resulting from implementation. The intention is to highlight areas where there appears to be existing support within GIFT for needed functionality or a need for new extensions, and also to welcome feedback. The following are some elements and observations from the current design work.

- The distributed training architecture involves a GIFT server, a RAAFT server to manage the training experience, and client user interfaces to be developed with the Angular framework. The servers communicate using REST protocols, and the RAAFT server communicates directly with the clients. Any user-specific client information going to or from GIFT is relayed through the RAAFT server. Session and profile management functions are shared across GIFT and the RAAFT server. Team member roles are defined in GIFT, and the GIFT Lobby is used to coordinate preparation for an exercise.
- There are two kinds of teamwork assessment in planned exercises. Declarative assessments involve questions posed to all participants at specific times during war gaming, using the GIFT surveys. Observational assessments monitor war gaming process behaviors to identify instances of good or bad teamwork. Both are organized for traceability to generalized teamwork dimensions.
- The representation for the declarative assessment questions to be generated includes parametric information for teamwork categories, contextualization to war gaming processes, and the nature of expected answers. These parameters and especially exercise context are prepared at the RAAFT server to create survey requests sent to GIFT.
- Question to be explored: how can the GIFT Tutor Module survey mechanisms support synchronized inputs from a team? For example, when team responses are to be assessed for consistency with each other, is there a best practice for suspending client-side actions and suspending scoring until all responses are collected?
- Question to be explored: for system process cues to leaders facilitating exercise control, is it most effective to use the same mechanisms as tutor feedback, or a different unique leader interface?

The assessment design is intended to have applications for other team trainers. For example, a wide range of team training applications both for distributed or co-located settings may have needs for declarative assessment mechanisms that use GIFT survey functionality but add contextualization to exercise events. Future work defining the observational assessment methods for monitoring team actions in the exercises is also intended to ultimately apply to other team trainers, as the focus is on assessment at the team decision level, abstracted from the platform.

ACKNOWLEDGMENTS

The research reported in this document was performed in connection with contract number W912CG-20-C-0014 with the U.S. Army. The views presented in this paper are those of the authors and should not be interpreted as presenting official positions, either expressed or implied, of the U.S. Government.

References

Cianciolo, A. T., & Sanders, W. R. (2006). *Wargaming Effectiveness: Its Conceptualization and Assessment* (Technical Report No. 1178; p. 128). U.S. Army Research Institute for the Behavioral and Social Sciences.

Folsom-Kovarik, J. T., Sinatra, A. M. (2020). Automating Assessment and Feedback for Teamwork to Operationalize Team Functional Resilience. In *Proceedings of the 8th Annual GIFT Users Symposium* (p. 126). US Army Combat Capabilities Development Command–Soldier Center.

Gilbert, S. B., Slavina, A., Dorneich, M. C., Sinatra, A. M., Bonner, D., Johnston, J., Holub J., MacAllister A., & Winer, E. (2018). Creating a team tutor using GIFT. *International Journal of Artificial Intelligence in Education*, 28(2), 286-313.

Johnston, J. H., Cannon-bowers, J. A., & Salas, E. (1998). Tactical Decision Making Under Stress (TADMUS): Mapping a program of research to a real world incident—The USS Vincennes.

Kozlowski, S. W., Grand, J. A., Baard, S. K., & Pearce, M. (2015). Teams, teamwork, and team effectiveness: Implications for human systems integration.

Marlow, S. L., Lacerenza, C. N., Paoletti, J., Burke, S., & Salas, E. (2018). Does team communication represent a one-size-fits-all approach?: A meta-analysis of team communication and performance. *Organizational Behavior and Human Decision Processes*, 144, 145–170. https://doi.org/10.1016/j.obhdp.2017.08.001

McCormack, R., Kilcullen, T., Sinatra, A. M., Case, A., & Howard, D. (2019). Teamwork Training Architecture, Scenarios, and Measures in GIFT. In *Proceedings of the 7th Annual GIFT Users Symposium* (p. 131). US Army Combat Capabilities Development Command–Soldier Center.

Salas, E., Sims, D. E., & Burke, C. S. (2005). Is there a "big five" in teamwork? Small Group Research, 36(5), 555–599.

Sottilare, R., Brawner, K., Goldberg, B. & Holden, H. (2012). The Generalized Intelligent Framework for Tutoring (GIFT). US Army Research Laboratory.

Sottilare, R. A., Shawn Burke, C., Salas, E., Sinatra, A. M., Johnston, J. H., & Gilbert, S. B. (2018). Designing Adaptive Instruction for Teams: A Meta-Analysis. *International Journal of Artificial Intelligence in Education*, 28(2), 225–264. https://doi.org/10.1007/s40593-017-0146-z

ABOUT THE AUTHORS

***Randy Jensen** has been a project manager for over 20 years at Stottler Henke Associates, Inc., in San Mateo, California, where he has led projects to develop intelligent tutoring systems for domains involving complex decision-making. Examples include trainers for air attack planning, unmanned vehicle command and control, small unit tactics, combined arms team exercises, information systems troubleshooting, satellite scheduling, and a current effort for division level command group war gaming. Randy has a B.S. with honors from Stanford University.*

***Bart Presnell** is a software developer at Stottler Henke Associates Inc. in San Mateo, California. He has over 15 years experience developing game-based training systems, including projects to train teamwork in medical scenarios, healthy eating and exercise, small unit tactics, and decision making in air combat rescues. Previously Bart worked in video game development at Electronic Arts. Bart has a B.S from Brown University and M.S from the Georgia Institute of Technology.*

***Jeanine A. DeFalco, PhD**, is Research Psychologist (Adaptive Training) with the Army Futures Command, CCDCSTTC, Orlando. Current research includes developing pedagogical models for the Generalized Intelligent Framework for Tutoring (GIFT) to accelerate expert problem-solving in critical care medical education, and supporting ethical decision making in soldier tactical and leadership training. She is currently the chair for IEEE P2247.4, Recommended Practice for Ethically Aligned Design of Artificial Intelligence (AI) in Adaptive Instructional Systems. Dr. DeFalco received her PhD in Psychology from Columbia University (2017), specializing in Human Development/Cognitive Studies in Education with a concentration in Intelligent Technologies.*

***Gregory Goodwin** is a senior research scientist with the U. S. Army Combat Capability Development Command Soldier Center at the Simulation and Training Technology Center (CCDC-SC-STTC), in Orlando, Florida. For the last decade, he has worked for the Army researching ways to improve training methods and technologies. He holds a PhD in Psychology from Binghamton University and an M.A. in Psychology from Wake Forest University.*

Automated Coaching in Synthetic Training Environments: Developing an Adaptive Team Feedback Framework

Randall Spain[1], Jonathan Rowe[1], Benjamin Goldberg[3], Robert Pokorny[2], Bradford Mott[1], and James Lester[1]
North Carolina State University[1]
Intelligent Automation, Inc.[2]
U.S. Army Combat Capabilities Development Command Soldier Center - STTC[3]

Introduction

As team performance becomes more important to organizational success, providing guided feedback and coaching to teams to improve their effectiveness has become increasingly important. Team science researchers have made substantial progress towards developing effective training strategies to improve team effectiveness, but these efforts have not been fully realized in adaptive instructional systems (AISs) for teams. The goals of AISs are to optimize learning and performance, improve retention of knowledge and skills, and support skills transfer to new tasks and domains. These goals are achieved by automatically tailoring instruction and scaffolding to fit the needs and skills of learners based on assessments of knowledge, skills, and abilities. Devising computational models that provide instructional support effectively—determining when to present feedback, what type of feedback to deliver, and how it should be realized—is a critical challenge, particularly for designing AISs to support team training.

The presence of multiple team members in a synthetic training exercise–compared to an individual learner–introduces new opportunities for providing feedback and coaching to learners and new challenges for testing team-based pedagogical models (Sottilare et al., 2018). For example, feedback could be delivered to an individual team member or the entire team. It could be delivered immediately after an error occurs (i.e., immediate feedback), during a lull between training events (partially delayed), or at the end of a training event (i.e., delayed feedback). The feedback could include minimal outcome feedback ("Your response was incorrect!") or detailed process feedback that provides clear, explicit instructions on how to correct errors or perform the task more appropriately ("You need to issue an alert prior to engaging the threat"). The level of cognitive engagement required of the learner upon receiving the feedback message can also vary. A feedback message that tells team members what they have done wrong and how to correct their actions could be modeled as a passive feedback message. In other instances, asking crews to engage in a more constructive activity, such as self-critiquing an exercise, may lead to deeper learning and better training transfer. Furthermore, the intent of the feedback message may vary. Human coaching often mixes motivational comments with corrective comments to help keep learners engaged and to develop a learners efficacy for learning or mastering a task. These design choices for feedback shape the volume and complexity of feedback that team members receive as well as the level of cognitive engagement and reflection time required of learners. In addition to considering how and when feedback should be delivered to team members, the team training literature suggests that a team's development phase may influence the type of feedback and coaching team members should receive. Understanding these phases of team development as well as the competencies and skills that team members must acquire to develop into an expert team is critical for developing AISs that can automatically and continuously improve when and how teams receive feedback.

There is growing evidence that machine learning techniques, such as reinforcement learning (RL), can provide effective data-driven approaches for modeling pedagogical coaching and feedback in AISs. However, a critical step towards leveraging machine learning techniques for providing automated coaching and feedback in AISs is creating a generalizable framework that AIS developers can utilize to determine 2 when, what type, and how to provide feedback to support team performance and learning during collective training tasks in synthetic training environments. In this paper, we begin to outline a general feedback framework that aims to meet this need, which is being developed for integration with the Generalized Intelligent Framework for Tutoring (GIFT), an open-source software framework of tools, methods, and standards for developing AISs. The framework delineates features that may impact how to provide feedback to teams during a synthetic training exercise, while accounting for factors that may impact the effectiveness of team feedback. The framework aims to assist AIS designers in developing pedagogical models that can automatically delineate what type of feedback to provide to teams as they advance from team formation towards becoming a high-performing team.

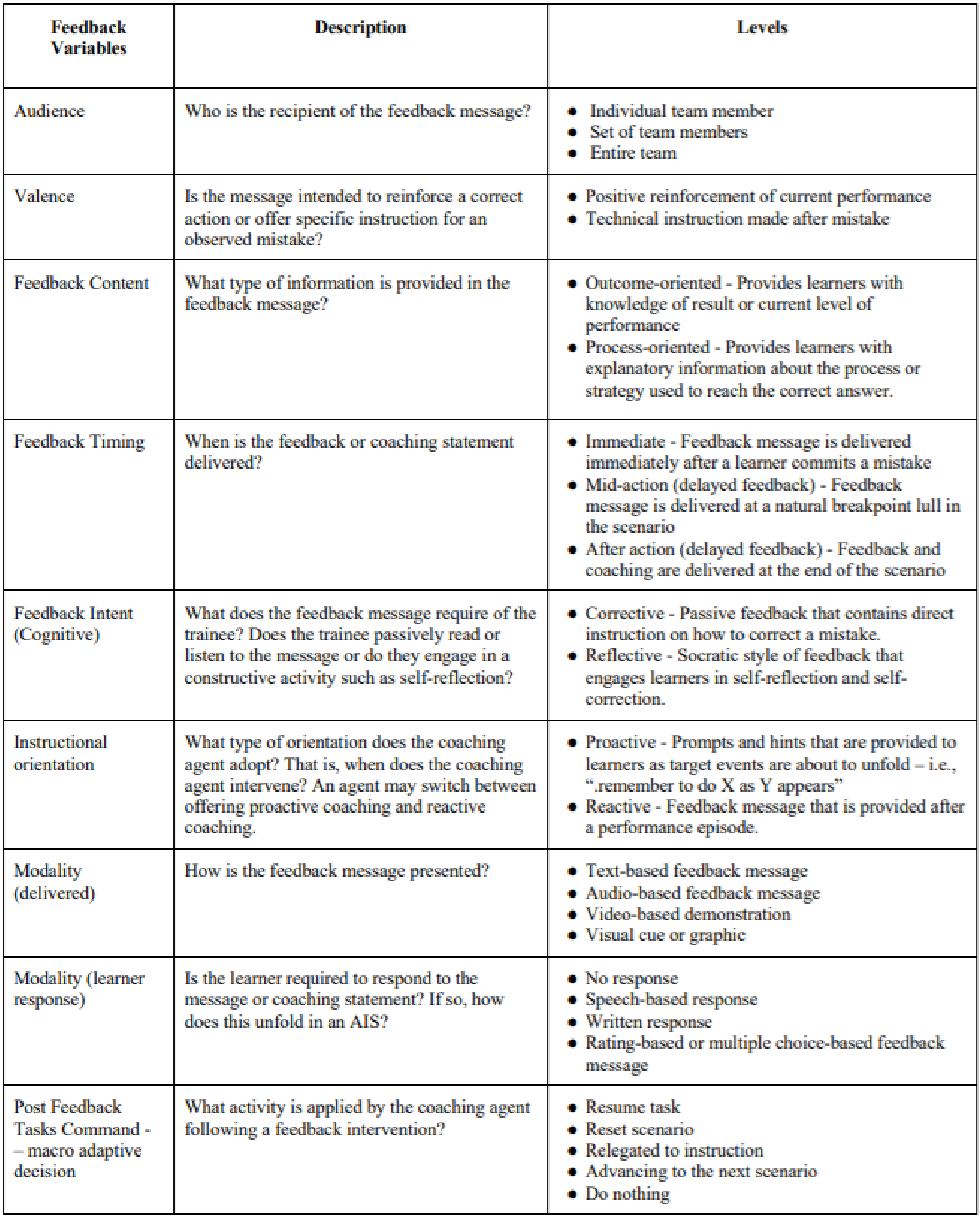
TEAMCOACH FRAMEWORK FOR ADAPTIVE TEAM FEEDBACK IN SYNTHETIC TRAINING ENVIRONMENTS

Establishing a feedback framework is an important first step towards providing automated coaching and feedback in an AIS. There have been several notable efforts to address this need. Goldberg et al., (2018) gathered pedagogical insights from the sports psychology literature to relate what is known about feedback in education and training settings and how it might enrich the automated delivery of feedback through AISs to support team training. The authors present a typology that organizes coaching statements based upon when to provide feedback, who to provide the message to, the type of information contained in the message, and what should occur following the feedback message. Goldberg's (2019) Team Interaction Feedback Taxonomy for AISs extends this typology and presents a high-level taxonomy of feedback and adaptation types for application in team-based AISs. The taxonomy establishes pedagogical decisions at three levels of interaction: (feedback direct toward an individual, team lead, or team), two levels of valence (positive and negative feedback statements), and two levels of timing (feedback delivered in realtime and after-action review). It also establishes whether the target of the feedback message is directed at the individual or teamcompetency level. The taxonomy is designed to serve as a foundation of pedagogical activities to build policies that can structure and scaffold the delivery of feedback based on runtime monitoring and trainee information.

The TeamCoach Framework for Adaptive Team Feedback in Synthetic Training Environments shown in Table 1 extends previous research to provide a set of feedback strategies and tactics that could be implemented in an AIS for teams. The framework is devised towards explicit feedback which refers to information provided to a learner from an external source such as a tutor, coach, or AIS, that intends to inform a student or learner how he or she performed. This is in contrast to implicit feedback which refers to information inherent in a scenario, task, or environment that may be used to determine the accuracy of a student's decision or input (Narciss, 2008). In the following sections, we describe the feedback variables outlined in the table.

Audience. In team-based AISs, feedback messages can be delivered to an individual team member, a subset of the team, or the entire team. The decision regarding who to direct the message towards depends on several factors including whether the learning objective targets individual or team-level skills and the desired outcome of the message. Salas (2015) notes that individual-level feedback should be delivered individually to learners rather than in a group to avoid embarrassment, which would reduce learners' experience of psychological safety and could impact trainee motivation. Conversely, feedback that aims to improve team-level KSAs should be delivered to the team.

Table 1. TeamCoach Framework for Adaptive Team Feedback Framework in Synthetic Training Environments



Feedback Variables	Description	Levels
Audience	Who is the recipient of the feedback message?	• Individual team member • Set of team members • Entire team
Valence	Is the message intended to reinforce a correct action or offer specific instruction for an observed mistake?	• Positive reinforcement of current performance • Technical instruction made after mistake
Feedback Content	What type of information is provided in the feedback message?	• Outcome-oriented - Provides learners with knowledge of result or current level of performance • Process-oriented - Provides learners with explanatory information about the process or strategy used to reach the correct answer.
Feedback Timing	When is the feedback or coaching statement delivered?	• Immediate - Feedback message is delivered immediately after a learner commits a mistake • Mid-action (delayed feedback) - Feedback message is delivered at a natural breakpoint lull in the scenario • After action (delayed feedback) - Feedback and coaching are delivered at the end of the scenario
Feedback Intent (Cognitive)	What does the feedback message require of the trainee? Does the trainee passively read or listen to the message or do they engage in a constructive activity such as self-reflection?	• Corrective - Passive feedback that contains direct instruction on how to correct a mistake. • Reflective - Socratic style of feedback that engages learners in self-reflection and self-correction.
Instructional orientation	What type of orientation does the coaching agent adopt? That is, when does the coaching agent intervene? An agent may switch between offering proactive coaching and reactive coaching.	• Proactive - Prompts and hints that are provided to learners as target events are about to unfold – i.e., ".remember to do X as Y appears" • Reactive - Feedback message that is provided after a performance episode.
Modality (delivered)	How is the feedback message presented?	• Text-based feedback message • Audio-based feedback message • Video-based demonstration • Visual cue or graphic
Modality (learner response)	Is the learner required to respond to the message or coaching statement? If so, how does this unfold in an AIS?	• No response • Speech-based response • Written response • Rating-based or multiple choice-based feedback message
Post Feedback Tasks Command - – macro adaptive decision	What activity is applied by the coaching agent following a feedback intervention?	• Resume task • Reset scenario • Relegated to instruction • Advancing to the next scenario • Do nothing

Feedback Valence. Valence refers to whether the message is intended to reinforce a correct action or offer specific instruction or remediation for an observed mistake. This variable is adopted from Goldberg's Team Interaction Feedback Taxonomy for AISs and is based on previous research examining coaching behaviors in the sports psychology literature. Specifically, Smith et al. (1977) devised a coaching behavior assessment system that classifies coaching behaviors into negative or positive responses. Examples of positive responses include providing trainees with positive statements following good performance, providing encouragement after a player commits a mistake (a positive, consoling reaction after a mistake), and providing technical instruction after a mistake. Negative reactions include providing punitive feedback after observing a mistake (i.e., calling out mistakes in front of others) and providing punitive technical instruction following a mistake. These forms of feedback are reflected in our taxonomy.

Feedback Content. The TeamCoach framework identifies two types of feedback content that are frequently discussed in the feedback and team-training literature: outcome and process-oriented feedback (Johnston et al., 2017). Outcome-oriented feedback provides learners with information about their current level of performance. It alerts trainees to performance problems and helps them to focus their attention on subsequent performance. In this manner, outcome feedback has both directive (i.e., you need to correct this mistake in the future) and motivational properties (I need to close the gap between current performance and my desired level of performance). As noted by Cannon-Bowers and Salas (1997), outcome feedback only informs a trainee that their performance must change, it does not provide guidance on how to change it or what needs to be changed. Process-oriented feedback aims to address this by providing learners with explanatory information about the processes or strategy needed to reach a desired goal or outcome (CannonBowers & Salas, 1997; Johnson et al., 2017). Its purpose is to provide the learner with information that can be used to close the gap between his or her current level of understanding or performance and the level required for successful performance. Process-oriented feedback can function to highlight errors, replace incorrect actions with appropriate responses, and reinforce the response so individuals will be more likely to act appropriately in the future. In general, empirical evidence suggests that process-oriented feedback is superior to outcome-oriented feedback for helping learners develop a deeper understanding of instructional material. The benefits of process-oriented feedback are evident in near transfer tasks and tests of knowledge retention (Billings, 2012; Johnson et al., 2017).

Feedback Timing. The TeamCoach framework distinguishes between three levels of feedback timing. Immediate feedback refers to guidance given directly following a mistake. Mid-action feedback is a form of delayed feedback in which feedback and coaching are provided at some break-point or lull in the training scenario but before another scenario or task begins. After action review is feedback that is delivered after the training event has concluded. There are a number of factors to consider in deciding whether to present learners with feedback immediately after an error or after delay. Empirical evidence suggests that immediate feedback seems to be more beneficial during the acquisition phase of learning and for promoting declarative and procedural knowledge and problem solving (Anderson et al., 1995; Shute 2008), but delayed feedback may be better for promoting skill transfer and retention (Astwood, 2009; Goodman & Wood, 2009). Immediate feedback has also been linked to higher learner motivation and attitudes (Wilson et al., 2009), but these studies have mainly investigated feedback timing in individual training settings.

Few studies have systematically evaluated feedback timing policies to support team training in synthetic training environments. One notable exception is a study conducted by Astwood (2009) who examined the impact of immediate and delayed feedback on team performance during a simulation-based training exercise. Results showed that training performance was not impacted by feedback timing, but that teams who received delayed feedback outperformed teams who received immediate feedback on a delayed retention test. More recently, Grande et al. (2016) found support for providing immediate feedback to enhance team knowledge, and more specifically for fostering better information sharing and exchange among team members, during a team training exercise that simulates a naval crisis relief operations and decision-making task. Results showed that teams that received feedback and coaching targeted at improving

5 information exchange were more efficient at generating collectively held knowledge compared to teams that did not receive feedback during training.

So far, we have treated the discussion of immediate versus delayed feedback as if training designers are forced to use one approach exclusively, but evidence suggests that using both immediate and delayed forms of feedback can offer superior learning outcomes compared to using either approach alone (Sanders, 2005).

Feedback Intent (Cognitive). Feedback intent refers to whether the message or coaching statement is meant to be corrective ("...you need to do X...") or reflective ("...can you identify what your team did incorrectly at this point in the scenario? What caused this breakdown?). This distinction is rooted in the human tutoring literature which show tutors can adopt either a Socratic style of tutoring which relies on questioning or a direct tutoring style which relies on telling learners what they did incorrectly (Lepper et al., 1997). In an AIS, the differences between these two styles may have implications on the level of cognitive engagement required of the trainee in an AIS. A reflective statement that directs learners to constructively self-reflect or to engage in dialogue with other team members to critique their collective performance may foster deeper more meaningful learning experiences than a corrective coaching statement that learners passively observe (Chi, 2009).

Instructional Orientation. Instructional orientation refers to whether coaching and feedback are provided reactively after the trainee has committed a mistake or proactively, during the task, but prior to an upcoming event in order to prevent a mistake. Proactive coaching statements are akin to feedforward statements that provide guidance to a learner before he or she attempts a task (Hendry White & Herbert, 2016). By definition, feedforward is timely and future-oriented and provided to the trainee in anticipation of the upcoming task or event. It includes giving student tips or guidance about what to do or what not to do based on previous performance (Webb & Moallem, 2016). Human coaches frequently switch between proactive and reactive coaching strategies. They provide tips and instruction to a learner—based on what they know about the learner's strengths and weakness—during training as a critical event approaches (e.g., "alright, remember to include a sensing term in your fire command"), and they provide feedback on performance once the event has unfolded. Freeman et al. (2005) utilized both proactive and reactive coaching in an adaptive team tutor designed to train air weapons officers with communication protocols and skills for communicating with fighter pilots. The coaching agent presented trainees with proactive messages leading up to an event or delayed feedback after the event had occurred if the timeliness did not allow for the proactive coaching to be provided. The coach's instructional model chooses between the different formats for feedback, escalating from feedforward statements of rules and warnings to direction, to explicit examples ("Say this now...") based upon the timeliness of when the statements could be delivered.

Feedback Modality (Delivery). Feedback modality reflects whether the feedback statement is presented as spoken words, as text, or as both in the training environment. The modality principle of multimedia learning states that presenting textual information in an auditory modality when concurrent visuals are present (i.e., pictures and diagrams) results in better learning than presenting textual information visually (Mayer, 2009). From a cognitive load perspective, providing feedback as spoken text is theorized to allow the learner to utilize visual and verbal information processing channels, which frees additional attentional resources for engaging in generative processing of the material. Providing visual feedback can, conversely, overload visual processes and can cause task interference and visual overload, leaving fewer resources available for the learner to engage in the cognitive processes necessary to engage in generative processing. The benefits of applying the modality principle to feedback in simulation-based training environments have been demonstrated in several studies (Fiorella et al., 2012; O'Neil et al. 2010).

Feedback Modality (Response). The feedback modality response variable addresses the questions of whether a learner is required to respond to a feedback or coaching statement. If so, how does this unfold in an AIS? Typically, during a feedback intervention, no overt or direct response from the trainee is registered

6 by the training system. However, in scenarios in which a team or individual is working with a synthetic coach, understanding how a train interaction with the coach becomes an important variable to model. The taxonomy contains four response approaches: no response, speech-based response, written response, and rating or multiple-choice based response.

Post Feedback Task Command (macro-adaptive decision). Similar to Goldberg et al. (2018), our taxonomy also includes a variable to account for the action the tutor should take upon providing feedback. These actions include resuming the task, resetting the scenario after delivering the message, ending the scenario so that trainees can engage in remediation or individual training, or advancing to the next activity or training exercise.

ADAPTIVE TEAM FEEDBACK IN SYNTHETIC TRAINING ENVIRONMENTS WITH GIFT

The TeamCoach framework outlines a broad range of variables that shape how, when, and what types of feedback are delivered to individuals and teams during team training in AISs. The design of adaptive feedback can have a significant impact on the effectiveness of team training activities (Shute, 2008). Reviews of the literature indicate there is high variability in the impact of feedback on learning, suggesting there may be no single best way to design feedback (Johnston et al, 2017; Narciss, 2008; Shute, 2008). Furthermore, there are many open questions about how different contextual factors, such as task characteristics, learner characteristics, team characteristics, and situational factors mediate the effects of feedback on learning. This points to the importance of developing computational models that can adapt feedback based upon the behavior and performance of individual learners and teams, as well as the context in which training is situated. Furthermore, it highlights the promise of data-driven approaches for developing adaptive feedback models by directly observing what works and what does not for a particular task, learner population, and training environment.

Developing computational models of adaptive team feedback raises important research and development challenges. For example, the U.S. Army's Synthetic Training Environment (STE) aims to provide effective multi-echelon training and mission rehearsal capabilities across a range of training domains and environments. This calls for the creation of generalized methods and resources to support the development and evaluation of adaptive team feedback models that facilitate reuse across different tasks and echelons, while supporting significant adaptivity to ensure effectiveness across a range of training contexts. There are several key requirements that computational models of adaptive feedback must meet to support training capabilities such as STE. First, adaptive feedback models must be capable of monitoring a broad range of factors to determine what types of feedback to deliver under different conditions. Second, feedback models must be explicit about the optimization criteria that are used to drive decisions about feedback variables and how feedback's impact should be evaluated. Third, adaptive feedback models must be compatible with generalized methods, models, and tools that can be used across a broad range of domains and training environments. Finally, adaptive feedback models must be computationally efficient in order meet the runtime performance requirements of simulation-based team training.

In support of these requirements, GIFT provides a suite of software tools and standards that can be used to create models of adaptive team feedback in AISs. By providing a reusable framework for adaptive training, GIFT can support the design, development, and investigation of AI-based feedback models that operate at both individual and team levels across a range of domains and simulation-based training scenarios. We are using GIFT to develop data-driven models of adaptive team feedback based upon the TeamCoach framework to enhance simulation-based team training exercises in the domain of crew gunnery. In the 7 following sections, we describe the crew gunnery testbed environment we are using in our work, and we discuss the use of GIFT to begin developing adaptive team feedback models in this domain.

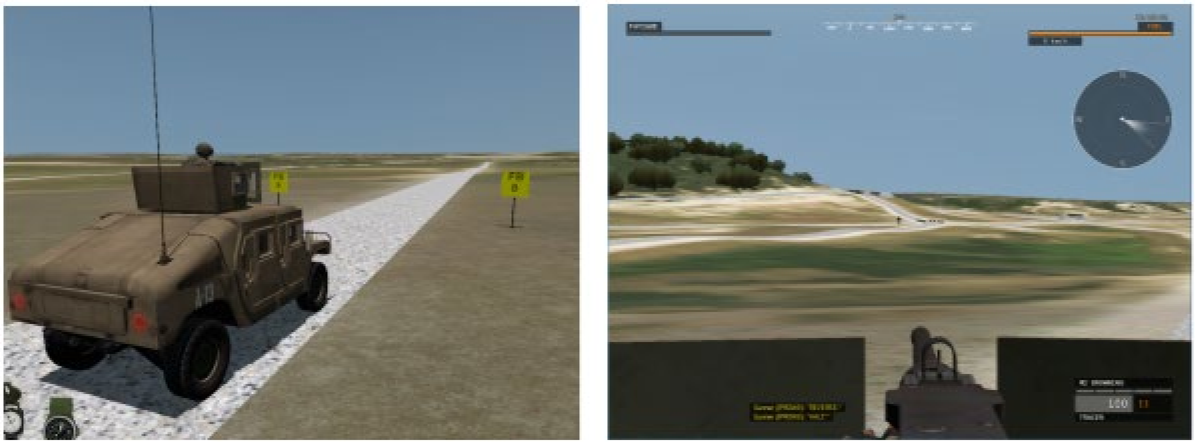


Figure 1. Crew Gunnery Training in Virtual Battlespace 3

Simulation-Based Crew Gunnery Training in Virtual Battlespace 3

To investigate the design and development of adaptive team feedback models with GIFT, we are using crew gunnery training in Virtual Battlespace 3 as a testbed. Virtual Battlespace 3 (VBS3) is a 3D game-based simulation platform for small unit training that is widely used by the U.S. military and other countries. Crew gunnery training in VBS3 centers upon 3-soldier crews as they practice skills and procedures for engaging stationary and moving targets from a mounted vehicle-based weapon system (e.g., Humvee, Truck, Stryker). Each crew member has an assigned role: Vehicle Commander, Driver, and/or Gunner. Vehicle crew members work together to detect and identify potential threats, decide how and whether to engage a target, implement a spoken fire command sequence, engage one or more stationary or moving targets, and assess the outcome of the engagement. The team must coordinate through a combination of spoken communication and behavior to engage the target(s) successfully and complete the mission. Soldiers work together in VBS3—this takes place through several networked computers, or, alternatively, a configured Form-Fit-Function vehicle simulator—in order to complete collective training events in a shared instance of the simulation-based training environment.

Assessing team performance during crew gunnery training involves examining crew members behavior and communication during training engagements. An important component of team behavior is the crew's execution of a fire command sequence, which is a well-defined protocol for communicating information and actions between team members to coordinate the effective engagement of a target. Key components of assessment include the execution of the fire command sequence, engagement completion time, and avoidance of procedural violations. Violations lead to deductions of points, and successful crews must achieve a minimum number of points in order to successfully pass a training exercise. Currently, feedback is often presented after a crew gunnery training exercise has been completed through an after-action review. The after-action review provides a forum for feedback and discussion, and it often requires time and expertise from a Vehicle Crew Evaluator (VCE) to help guide the discussion. Devising computational models of adaptive team feedback promises to complement existing crew gunnery training workflows by enhancing the feedback that can be delivered during simulation-based crew gunnery practice even when VCEs are not necessarily available.

Developing Adaptive Team Feedback Models in GIFT

GIFT offers several features that support the design and development of adaptive team feedback models grounded in the TeamCoach framework. In this section, we describe key components of GIFT that we are using to develop data-driven models of adaptive team feedback for crew gunnery training in VBS3. Key GIFT capabilities include: (1) Automated measurement of team behavior and performance, (2) Micro- and macro-adaptive feedback using GIFT Course Creator, and (3) Reflective feedback and assessment through GIFT GameMaster. In addition, we discuss potential enhancements to GIFT that would further support research and development on adaptive team feedback models for synthetic training environments.

Measuring Team Behavior and Performance during Simulation-Based Training. Team competencies are integral to team performance measurement and determining where to direct remediation to improve team processes and team performance (Cannon-Bowers & Salas; 1997). GIFT provides several measurement functionalities to capture both survey and process data at the individual and team levels. GIFT's survey tools can be utilized to create and administer standard and/or customized surveys within an online training course. The Gateway Module enables bi-directional communication with external training environments (e.g., VBS3), which supports the capture of fine-grained timestamped logs of individual and team behavior within a simulation-based training environment. In the case of VBS3, GIFT can receive and record Distributed Interactive Simulation (DIS) message traffic to capture sequential data on simulation states, which constitutes a form of interaction trace log data for teams' simulation-based training interactions. GIFT also has auxiliary logging functionalities, such as the GIFT-VBS plugin, which provide a mechanism for querying the states of specific VBS entities that are not captured in DIS records. As a complement to trace data logging, GIFT is capable of recording multimodal sensor data, including video recordings and audio recordings of crew gunnery training events, in order to capture verbal and non-verbal communication during team training exercises. Notably, this can be implemented in a lightweight version of GIFT called the GIFT Data Collector that provides a subset of GIFT's functionalities focused on instrumenting a simulation-based training exercise. We are utilizing this set of capabilities in order to develop a data collection plan for recording multimodal data on crew gunnery training in VBS3 with Army soldiers to produce a corpus for the creation of data-driven models of adaptive feedback.

Designing Micro- and Macro-Adaptive Team Feedback using GIFT Course Creator. The TeamCoach framework identifies feedback timing as an important variable in the design of adaptive team feedback. Specifically, the framework distinguishes between immediate real-time feedback, mid-action feedback, and after-action feedback. GIFT provides several mechanisms in order to support the implementation of different feedback timings across a range of domains and training environments. A key component of GIFT courses is the Domain Knowledge File (DKF), which encodes a domain model and set of assessment rules using a standardized XML format. GIFT breaks down training domains in terms of tasks, where a task is a type of performance node that can be utilized to organize evidence for inferring learner and/or team competencies. Tasks are defined in terms of start triggers and end triggers, which specify how and when tasks become active and conclude, respectively. These triggers serve as an ideal insertion point for delivering real-time feedback, such as reactive feedback when a task has been completed or proactive feedback when a new task is about to begin. The DKF also includes nodes that enable the targets of team feedback to be specified, including individuals, subsets of team members, or the team as a whole.

The GIFT Course Creator also provides a visual interface for defining the flow of training activities in a course, which is analogous to the outer loop of an intelligent tutoring system. The transitions between training activities can be defined manually or algorithmically, such as in adaptive course flow objects implemented with the Engine for the Management of Pedagogy (EMAP). By implementing adaptive feedback through course flow authoring tools in the GIFT Course Creator, it is possible to devise midaction and after-action feedback that can be sequenced within a progression of simulation-based training activities. For example, mid-action feedback could be delivered between a sequence of multiple VBS crew 9 gunnery practice missions, or it could be delivered after all of the practice missions have been completed in an after-action review context.

GIFT also provides mechanisms for implementing different feedback modalities. For example, GIFT can be utilized to play audio feedback messages during an ongoing VBS training mission, providing real-time feedback that does not interrupt the progression of a simulation-based training exercise. Feedback can also be delivered through GIFT's web-based Tutor User Interface (TUI). This form of feedback can take the form of a text passage or static webpage, or it could demand a more active learning process, such as highlighting, summarization, or even interactions with AutoTutor-style conversational agents. Different forms of feedback can also be delivered directly within a simulation-based training environment in the form of scenario adaptations. Specifically, in-simulation feedback could be delivered by a non-player character, a custom heads-up display, or in-simulation messaging features. This requires the design and implementation of feedback events within the simulation environment—for example, feedback mechanisms could be configured through the VBS Mission Editor—providing a means for delivering seamless adaptive feedback within a simulation-based training platform while leveraging the reusable AIS functionalities provided by GIFT.

Reflective Feedback and Assessment with GIFT GameMaster. GIFT GameMaster provides a suite of tools for automated playback and annotation of recorded training events, including training events that occurred within or outside of GIFT. GameMaster enables alignment and playback of video/audio recordings, trace log data, GPS position data, and real-time assessment data, providing multiple data-rich perspectives on a single training event. GIFT GameMaster can be utilized by instructors, providing opportunities for additional assessment and coaching. Learners can use GameMaster to support reflection on past training events. Researchers can also use GameMaster to examine and analyze training activities for research and development purposes. In particular, GIFT GameMaster has utility for the creation of datadriven models of adaptive team feedback, because it can be utilized to collect additional assessment and feedback data beyond that which can be collected in real-time during a training exercise. In other words, it provides a visual interface for collecting additional "labels," or "observations," to develop machine learning-based models of team performance and feedback. By augmenting real-time data with additional assessments and annotations that are collected after a training event has occurred, the resulting dataset can be utilized to train computational models for assessing team behavior and performance, as well as strategies for providing adaptive feedback based upon demonstrations by expert instructors.

Opportunities for Enhancing GIFT to Support Adaptive Team Feedback

Although GIFT provides a robust set of functionalities to support the development of adaptive team feedback models in alignment with the requirements of synthetic training environments, there are several opportunities for enhancing GIFT to further support these efforts. In this section, we identify promising directions for prospective enhancements to GIFT that would benefit the creation of data-driven models of adaptive team feedback grounded in the TeamCoach framework. The proposed enhancements complement previous suggestions for technological and methodological advances related to GIFT, including improved alignment between data-driven tutoring methodologies and the Learning Effect Model, support for balancing exploration vs. exploitation in data-driven models of team feedback, and enhancements to GIFT's Event Reporting Tool (Rowe et al., 2020).

Multimodal Assessment of Domain-Specific Team Communication and Behavior. As mentioned above, an important mechanism for governing the timing of adaptive feedback in GIFT is task start and end triggers within DKFs. Start and end triggers define the beginning and end of an assessment process within a particular domain. These triggers are driven by condition classes, which are custom Java classes containing logic that take the states of a simulation environment as input and produce as output evaluations

10 of whether selected criteria have been met. Examples of condition classes in GIFT include Avoid/Reach Location, Engage Targets, and Timer. Conditions in GIFT can be configured by providing arguments that are assigned to condition parameters, such as the name of an important location that a learner must reach, or the amount of time that should be set in a timer. Currently, GIFT condition classes are limited to operating upon information about the state of a simulation environment. In other words, they are sensor-free (Paquette et al., 2016). A major source of evidence for evaluating team processes is team communication data, including both verbal and non-verbal communication. Analyzing team communication is inherently multimodal and multichannel. Evaluating teamwork conditions requires triangulating across multiple different channels—simulation states, spoken communication, non-verbal behavior—in order to ascertain whether specific teamwork conditions have been met.

Devising multimodal analysis of real-time team communication and behavior presents significant challenges. There are multiple active projects, including projects utilizing GIFT and as well as others, that are investigating the creation of multimodal models of team communication and collaboration (Min et al., 2021; Stewart, Keirn, & D'Mello, 2021). In order to drive computational models of adaptive team feedback that are capable of operating upon a range of feedback variables under different domains and training environments, it will be necessary to gradually develop new types of conditions, or other evaluation mechanisms, to analyze multimodal teamwork data and enrich the range of possible conditions that can be evaluated for triggering and informing the design and delivery of adaptive team feedback.

Distributed Multimodal Data Collection. When collecting rich, multimodal data on team performance and behavior, it is often necessary to instrument distributed systems consisting of multiple networked computers in communication with one another. This raises issues related to multimodal data synchronization and alignment across multiple GIFT clients for the purpose of data collection, among other challenges. GIFT provides a robust set of tools for capturing and aligning learner performance data during simulation-based training on a single machine, but in a team context with multiple networked GIFT clients, effective workflows and tools for multimodal data capture and management becomes critical. For example, if multiple GIFT clients are utilized to capture video, audio, and trace log data from different team members in a crew gunnery training exercise for the purpose of creating data-driven models of adaptive team feedback, it is critical that it be possible to efficiently and accurately align and synchronize the many different data streams at a high level of precision. These challenges are especially salient data-driven learning technologies, because this type of research involves deploying simulation-based training environments at a sufficient scale that it can impose a significant burden of multimodal data synchronization and alignment if not handled automatically at the time of data collection.

Adaptive Team Feedback Prioritization. A natural question that instructors face as they assess team performance and deliver feedback is where to focus their attention. There are myriad areas toward which feedback can be directed, including learners' individual behavior, team-level performance, teamwork skills, and engagement, among others. It is neither practical nor efficient to assess and respond to every facet of team performance and behavior. Instructors must determine which areas to prioritize while giving feedback in order to optimize the effectiveness and efficiency of training. By analogy, there is an opportunity to embed guidance on how to prioritize the types and targets of adaptive feedback in GIFT for different conditions. This could be delivered directly through documentation and examples that are external to GIFT, as well as built directly into the user interfaces and models that are interwoven in GIFT's course development workflows. Devising computational models that prioritize areas of feedback is an important challenge. As adaptive team feedback capabilities are integrated into GIFT, the design of adaptive feedback models and tools will inherently foreground certain types of feedback areas and not others. Therefore, it is important to consider what messages about feedback priorities these designs communicate. Furthermore, sharing datasets, courses, and tools will similarly highlight what types of feedback targets are most likely to impact training effectiveness.

CONCLUSION AND FUTURE DIRECTIONS

Developing computational models that automatically determine when and how feedback is delivered to team members is critical for realizing the potential of team AISs. The feedback variables presented in the TeamCoach framework are a foundation for the types of feedback strategies and tactics that can be applied in a team-based AIS. The framework can be used to configure feedback and coaching features as they apply to team training strategies and techniques that are used to train individual and team-level KSAs. We are investigating the creation of data-driven adaptive team feedback models based on the TeamCoach framework in the domain of crew gunnery training using GIFT. GIFT provides a broad range of critical features that align with the requirements of adaptive team feedback in synthetic training environments, including tools and standards for measuring team performance and behavior, mechanisms for developing micro-adaptive and macro-adaptive team feedback, and opportunities for reflective assessment and feedback through the GIFT GameMaster interface. There are also several promising opportunities for prospective enhancements to GIFT that would further advance research and development of adaptive team feedback models for synthetic training environments, including enriched tools for multimodal assessment of domain-specific team communication, enhanced support for distributed multimodal data collections, and consideration of how GIFT models and tools communicate prioritization decisions about adaptive team feedback. Evaluating and developing these capabilities in GIFT in service to the creation of data-driven adaptive team feedback models in the domain of crew gunnery training, as well as other domains and echelons, is an important direction for continued and future research.

References

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. Journal of the Learning Sciences, 4(2), 167–207.

Astwood, R. (2009). Effective timing of feedback during scenario-based team training within a simulated environment. Doctoral Dissertation, University of Central Florida.

Billings, D. R. (2012). Efficacy of adaptive feedback strategies in simulation-based training. Military Psychology, 24 , 114–133.

Cannon-Bowers, J. A., & Salas, E. (1997). A framework for developing team performance measures in training. In M. T. Brannick, E. Salas, C. Prince (Eds.), Team performance assessment and measurement: Theory, methods, and applications (pp. 45-62). Mahwah, NJ: LEA.

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

Freeman, J., Haimson, C., Diedrich, F. J., & Paley, M. (2005). Training teamwork with synthetic teams. In Eds C. Bowers, E., Salas, F. Jentsch. Creating High-Tech Teams: Practical Guidance on Work Performance and Technology. (pp 261-282). American Psychological Association, Washington DC.

Goldberg, B., Nye, B., Lane, H. C., & Guadagnoli, M. (2018). Team assessment and pedagogy as informed by sports coaching and assessment. In R. A. Sottilare, A. C. Graesser, X. hu, & A. M. Sinatra (Eds.), Design 12 Recommendations for Intelligent Tutoring Systems, Volume 6: Team Tutoring. (pp 105-199). U.S. Army Research Laboratory. Orlando: FL.

Goldberg, B. (2019). Adaptation and Pedagogy at the Collective Level: Recommendations for Adaptive Instructional Systems. In the International Conference on Human-Computer Interaction (pp. 301-313). Springer, Cham.

Goodman, J. S., & Wood, R. E. (2009). Faded versus increasing feedback, task variability trajectories, and transfer of training. Human Performance, 22, 64-85.

Grand, J. A., Braun, M. T., Kuljanin, G., Kozlowski, S. W., & Chao, G. T. (2016). The dynamics of team cognition: A process-oriented theory of knowledge emergence in teams. Journal of Applied Psychology, 101(10), 1353.

Hendry, G. D., White, P., & Herbert, C. (2016). Providing exemplar-based 'feedforward' before an assessment: The role of teacher explanation. Active Learning in Higher Education, 17(2), 99-109.

Johnson, C. I., Bailey, S. K., & Van Buskirk, W. L. (2017). Designing effective feedback messages in serious games and simulations: A research review. In P. Wouters & H. van Oostendorp (Eds.), Instructional techniques to facilitate learning and motivation of serious games (pp. 119–140). Cham, Switzerland: Springer.

Lepper, M. R., Drake, M. F., & O'Donnell-Johnson, T. (1997). Scaffolding techniques of expert human tutors. In K. Hogan & M. Pressley (Eds.), Scaffolding student learning: Instructional approaches and issues (pp. 108-144). Cambridge, MA: Brookline Books.

Mayer, R. E. (2009). Multimedia learning (2nd ed.). New York, NY: Cambridge University Press.

Min, W., Spain, R., Saville, J., Mott, B., Brawner, K., Johnston, J., & Lester, J. (2021). Multidimensional team communication modeling for adaptive team training: A hybrid deep learning and graphical modeling framework. To appear in Proceedings of the Twenty-Second International Conference on Artificial Intelligence in Education.

Moallem, M., & Webb, A. (2016). Feedback and feed-forward for promoting problem-based learning in online learning environments. Malaysian Journal of Learning and Instruction, 13(2), 1-41.

Narciss, S. (2008). Feedback strategies for interactive learning tasks. In J. M. Spector, M. D. Merrill, J. J. G. van Merriënboer, & M. Driscoll (Eds.), Handbook of research on Educational communications and technology, (3rd edition, pp. 125-144). New York: Lawrence Erlbaum Associates.

O'Neil, H. F., Chuang, S. S., & Baker, E. L. (2010). Computer-based feedback for computer-based collaborative problem solving. In D. Ifenthaler et al. (Eds.), Computer-Based Diagnostics and Systematic Analysis of Knowledge, pp. 261–279.

Paquette, L., Rowe, J., Baker, R., Mott, B., Lester, J., DeFalco, J., Brawner, K., Sottilare, R., & Georgoulas, V. (2016). Sensor-Free or Sensor-Full: A Comparison of Data Modalities in Multi-Channel Affect Detection. In T. Barnes, M. Chi, & M. Feng (Eds.), Proceedings of the 8th International Conference on Educational Data Mining (EDM'16), pp. 93–100.

Rowe, J., Spain, R., Goldberg, B., Pokorny, R., Mott, B., & Lester, J. (2020). Toward data-driven models of team feedback in synthetic training environments with GIFT. In B. Goldberg (Ed.), Proceedings of the 8th Annual GIFT User Symposium (GIFTSym8), pp. 136–144.

Salas, E. (2015). Team training essentials: A research-based guide. Routledge. Sanders, M. J. (2005). The effect of immediate feedback and after-action reviews (AARs) on learning, retention and transfer. Unpublished master's thesis, University of Central Florida, Orlando.

Shute, V. J. (2008). Focus on Formative Feedback. Review of Educational Research, 78(1), 153–189.

Smith, R. E., Smoll, F. L., & Hunt, E. (1977). A system for the behavioral assessment of athletic coaches. Research Quarterly. American Alliance for Health, Physical Education and Recreation, 48(2), 401-407.

Sottilare, R. A., Burke, C. S., Salas, E., Sinatra, A. M., Johnston, J. H., & Gilbert, S. B. (2018). Designing adaptive instruction for teams: A meta-analysis. International Journal of Artificial Intelligence in Education, 28(2), 225-264. 13

Stevens, R., Gorman, J., Zachary, W., Johnston, J., Dorneich, M., & Foltz, P. (2018) How is this team doing, and why? In Eds (R. Sottilare, A. Graesser, X. Hu, & A. Sinatra) Design Recommendations for Intelligent Tutoring Systems, 77-92. US Army Research Laboratory.

Stewart, A.E.B., Keirn, Z. & D'Mello, S.K. (2021). Multimodal modeling of collaborative problem-solving facets in triads. User Modeling and User-Adapted Interaction.

Wilson, K.A., Bedwell, W.L., Lazzara, E.H., Salas, E., Burke, C.S., Estock, J.L., Orvis, K.L., & Conkey, C. (2009). Relationships between game attributes and learning outcomes. Simulation and Gaming, 40, 217-266.

ACKNOWLEDGEMENTS

The research described herein has been sponsored by the U.S. Army Research Laboratory under cooperative agreement W912CG-19-2-0001. The statements and opinions expressed in this article do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

ABOUT THE AUTHOR

Dr. Randall Spain *is a Research Psychologist in the Center for Educational Informatics at North Carolina State University where he uses principles, theories, and methods of applied psychology to design and evaluate the impact of advanced training technologies on learning and performance. Dr. Spain holds a PhD in Human Factors Psychology and serves on the editorial board for Military Psychology.*

Dr. Jonathan Rowe *is a Research Scientist in the Center for Educational Informatics at North Carolina State University, as well as an Adjunct Assistant Professor in the Department of Computer Science. He received the PhD and MS degrees in Computer Science from North Carolina State University, and a BS degree in Computer Science from Lafayette College. His research focuses on artificial intelligence in advanced learning technologies, with an emphasis on game-based learning environments, intelligent tutoring systems, multimodal learning analytics, learner modeling, and computational models of interactive narrative generation. Dr. Rowe also serves on the editorial boards of the International Journal of Artificial Intelligence in Education and IEEE Transactions on Learning Technologies.*

Dr. Benjamin Goldberg *is a senior researcher in the Learning in Intelligent Tutoring Environments (LITE) Lab at the Combat Capabilities Development Command (CCDC) Soldier Center, Simulation and Training Technology Center (STTC) in Orlando, FL. He has been conducting research in modeling and simulation for the past eight years with a focus on adaptive learning and how to leverage artificial intelligence tools and methods for adaptive computerbased instruction. Currently, he is the LITE Lab's lead scientist on instructional strategy research within adaptive training environments. Dr. Goldberg holds a PhD from the University of Central Florida in Modeling & Simulation.*

Dr. Robert Pokorny *is Principal of the Education and Training Technologies Division at Intelligent Automation, Inc. He earned his PhD in Experimental Psychology at the University of Oregon in 1985, and completed a postdoctoral appointment at the University of Texas at Austin in Artificial Intelligence. His 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. 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 Distinguished University 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 PhD 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).*

Reasoning about Team Roles and Responsibilities for Team Assessment

J.T. Folsom-Kovarik[1], Joel Sieh[1], and Anne M. Sinatra[2]
[1]Soar Technology, Inc., Orlando, Florida, USA [2]U.S. Army Combat Capabilities Development Command Soldier Center (DEVCOM-SC), Simulation and Training Technology Center (STTC), Orlando, Florida, USA

Introduction

Creating computer-based adaptive training for teams is a complex and difficult task. The task becomes even more difficult when it is being approached with the goal of creating functionality and authoring tools that promote generalizability. The Generalized Intelligent Framework for Tutoring (GIFT) and its associated tools have been developed to provide a general reusable framework for the creation of intelligent tutoring systems (Sottilare, Brawner, Sinatra, & Johnston, 2017). Careful consideration has been given to any updates that occur to GIFT to ensure that the authoring process remains flexible enough to support different types of training and scenarios.

As GIFT moves towards supporting team tutoring there are a number of additional considerations that need to be taken into account including but not limited to team communication, how to assess the performance of multiple individuals performing different roles in a training scenario, and ensuring that the authoring process does not restrict the types of teams and scenarios that can be created with GIFT. One of the most important aspects of GIFT is the GIFT authoring tool, which lets those without a background in computer science create intelligent tutoring systems. As team tutoring functionality increases, the need for understandable and consistent ways to create this tutoring increases as well. Therefore, it is important to develop approaches for defining team roles, and authoring rules that are aligned with how authors think about teams and flexible. Part of this flexibility is the utility of dynamic team roles which are linked to tasks or assessments during a scenario that are not easily assignable at the beginning of a scenario. These roles may change throughout the interaction with the system and be based on actions that have happened in the scenario, or even the proximity of the learners to an object in a scenario. Defining a process for creating, assessing, and implementing these roles in GIFT is an important next step to implementing team tutoring.

The present work is part of an effort to enhance GIFT with the ability to express team performance in a domain by differentiating heterogenous roles and functions for individuals and the ability to link individual contributions to team outcomes. Additional detail is available in (Folsom-Kovarik & Sinatra, 2020). Within this context, assessing team performance includes both observable team outcomes, such as completing a task, and inferred team process, such as coordinating activities and sharing information effectively. By suggesting experimental changes in GIFT, it becomes possible to express proper or improper team performance in terms of roles and responsibilities within the team. These are hypothesized to align with team training frameworks and make them more efficient to author as reusable assessments in GIFT.

An example framework for describing team process and observable outcomes is Team Dimensional Training (TDT; Smith-Jentsch et al., 1998; Smith-Jentsch et al., 2008). TDT defines four dimensions along which teams demonstrate differences in how they work together: Information Exchange, Communication Delivery, Supporting Behavior, and Initiative or Leadership. TDT has been applied to assess teams in numerous domains including tactical combat casualty care (TC3) and other infantry training (Johnston et al., 2019).

The overarching dimensions in TDT can be operationalized by defining Targeted Acceptable Responses to Generated Events or Tasks (TARGETs; Fowlkes & Shawn, 2004). These identify observable behaviors in the context of scenario events, which give the conditions for performance, and acceptable responses, which give the constraints to identify optimal or acceptable responses. GIFT's experimental roles and responsibilities can express how team members play different roles within a team and what responsibilities are associated with each role. As a result, GIFT gains authoring that efficiently defines when a team member is not fulfilling an expected responsibility, when another team member fills in to repair or preempt such a failure, and facts about the corrective action such as the delay and the appropriate choice of replacement.

Collectively the use of roles and responsibilities to assess team performance and individual contributions to team performance enable assessing the complex cognitive team-level skill called *team functional resilience* (Neville et al., 2020). Team functional resilience refers to the constellation of abilities that let a heterogenous team work effectively despite the compromise or loss of a function some team member contributes. As a complex skill, team functional resilience covers numerous TDT dimensions that have the potential to reveal insights about team process rather than simply team performance. Therefore, studying team functional resilience gives the present work a handle on the challenging tasks of inferring team communication, supporting behavior, and shared mental models while still maintaining traceability to observed behaviors and team outcomes. The resulting team assessments are hypothesized to support effective feedback that generalizes to help the team understand their behavior better than reporting outcomes alone.

There have been a number of different approaches to team tutoring that have been used with GIFT to date. These include creating a surveillance task in Virtual Battlespace 2 (VBS3) that first included two team members, and then was updated to include three (Gilbert et al., 2018). In the three-person version of this task the team was made up of two spotters and one sniper (Ouverson et al., 2021). These roles were entirely defined by the domain knowledge file (DKF) that was used for the individual. The roles remained consistent through the engagement with the task, and assessment at the team level was made up of an additional DKF that looked at all of the team members together. This strategy worked for a three-person team, however, as team size increases this would result in an exponential increase in needed DKFs. The next approach that was used for team tutoring in GIFT involved Virtual Battlespace 3 (VBS3), and created team assessments using one overall DKF that assessed the team's performance on cohesion and coordination measures (McCormack et al., 2019; McCormack et al., 2020). This approach allowed for assessment of the overall team, but did not focus on the individual roles of individuals on that team. The next progression in team tutoring work in GIFT involved the development of approaches to define team roles in GIFT, and then effort to improve their functionality and generalizability (Folsom-Kovarik and Sinatra, 2020).

The work described in the present paper extends team roles to let GIFT assign roles to team members dynamically during training execution. Dynamic role assignment modifies the past work on team roles by implementing a theory of symbols and referents which can change during training. Therefore, it becomes possible to bind one symbol or team role with a referent or individual learner. That binding can then be used throughout GIFT to record and reason about actions that help assess behavior and complex skills. On the other hand, it also becomes necessary to frequently re-evaluate the bindings and detect when they change, meaning that a team member takes on a new role or a role becomes empty. The technical changes to GIFT that enable this are described as a possible enhancement once fully implemented and evaluated.

The current status of the work described in this paper includes ongoing design, implementation, and refinement in the light of community feedback. Community feedback on the first year of the work helped to enhance understanding of how the designs would work under different conditions. Ongoing feedback will help to meet the widest possible variety of team training use cases and stakeholder requirements.

The second section of this paper describes a motivating example which grounds the recent enhancements and adds some concrete details. The third section describes the technical approach being designed and implemented to meet the team training needs. Finally, near-term opportunities are discussed and some recommendations for longer-term extensions to GIFT are offered.

Team Assessment with Dynamic Roles

Dynamic roles let GIFT assess team performance in challenging open-world settings. When a training environment is open it allows many learner actions and complex or emergent outcomes of actions. This makes it likely that evolving scenario context or a loss of team function can change the roles individuals play in a team. In such settings, GIFT can use information about dynamic roles and responsibilities to reason about who is in each role and recognize role changes from the observable behavior of individuals. This capability helps to assign credit for individual training events during a team scenario. It also helps characterize teams by their differing ability to change roles quickly and effectively. As a result, GIFT can support adaptive feedback to help the correct individuals and the team as a whole.

Types and Sources of Team Roles

Heterogenous team roles have previously been studied in several research contexts as described in this section: organizational structures and job titles, traits of individual team members, and task-specific factors. Roles are more dynamic when they are specific to a task or emerge in response to environment and context. Taken in this more dynamic sense, the definition of roles can even arise from the formation of dyads and larger groups interacting on a task, such as the functional roles of having situational information to share or being more or less available to take on tasking. While the present work focuses on dynamic roles, all team roles may be assessed.

First, much work on training teams in GIFT has focused on roles defined by organizational factors (e.g. Rowe et al., 2020; McCormack et al., 2019). Organizational factors include a team member's role as a member of the U.S. military, rank, occupational specialization, billet or assignment, and job duties. Since these types of roles do not change during the course of a training session, it is appropriate to define them in advance of the training instead of applying effort to infer them dynamically. Static roles were previously used to assess team functional resilience (Folsom-Kovarik and Sinatra, 2020).

Another perspective on team roles includes personality types (Peeters et al., 2006) and skillsets (Belbin, 1981). These roles describe a team member's personality, ability, patterns of thought and behavior, or other traits of the team member that may be inherent or slow to change. Team member traits may further be assigned to binary dimensions. As just one example, the roles suggested by Belbin (1981) include "chairman" and "task worker." The categorical roles could also be arranged in a continuum, e.g. relationship-oriented to task-oriented, and used to suggest more or less productive team pairings (Fisher, Hunter, & Macrosson, 1998).

However, not all team member responsibilities seem to arise from organizational roles or personal traits. Members in a functioning team have responsibilities that change in the course of accomplishing a task. This fact suggests the introduction of dynamic roles to let observers relate team members to these responsibilities. Similar concepts appear in fields of study like governance, organizational psychology, and resource allocation (Thomson, Perry, & Miller, 2009). Concepts of roles and responsibilities also appear in artificial intelligence research into task planning and intent modeling (Tambe, 1997; Georgeff et al., 1998).

For team assessment, dynamic roles complete the picture of which team members are required, expected, or able to fulfill a responsibility. The dynamic roles as implemented in an experimental branch enable GIFT to detect compromised team function, repairs to the compromise, goodness of the repair, and individual credit for the team's performance. We hypothesize dynamic roles may be categorized by the reasons team members get new responsibilities during a task:

- Situational: Roles based on location, current task requirements, or immediate/transitory states such as knowledge or availability
- Assigned: Roles set by another participant in the scenario, by standard operating procedure, or in the training realm by an instructor
- Assumed: Roles that a learner chooses to take on and must be inferred from their observable actions

Situational roles describe facts about the training scenario including the context for task performance. For example, one teammate might have information that the rest of the team needs to know, putting them in the role of a reporter. When Soldiers move in a formation or pilots fly in formation, the position relative to the rest of the team can define roles about how to move or engage enemies. A location such as the intersection of two roads or the coordinates of a wounded Soldier might also define which team member is the closest and able to quickly cover the intersection or attend the casualty. GIFT can detect changes in these roles with the world state model and history software components implemented in the experimental branch.

Assigned roles can be detected by the GIFT authoring tool, scenario configuration, or other computer entry (such as input from training observers or speech understanding if available). Even though role assignment might not be automated, the existence and manual assignment of these dynamic roles still help GIFT to automate the interpretation of events and actions. For example, assigned roles can help GIFT interpret who is or is not carrying out a responsibility. Assignments in a team setting might be construed to include orders from above, requests from subordinates, or any interaction that creates a commitment.

Assumed roles are those a team member takes on without being assigned. With the new dynamic role assignment, GIFT detects occurrences when team members assume a role by comparing their actions to the expert model. This new capability is key to understanding supporting behavior in Team Dimensional Training. A form of supporting behavior includes task sharing to help an overloaded teammate. Also, when a team function is compromised, the corrective action that restores the function can be taken by another person assuming the responsibility, meaning they are acting in the capacity of that role that is associated in the expert model. Finally, a team member who leaves a role and then resumes it may be said to have assumed the role at the time of resumption, for the purposes of assessing how long a role is vacant.

With these dynamic roles added in an experimental GIFT implementation, the assessment of team functional resilience can be more general to work in more contexts and can detect more instances of resilience for automated assessment compared to the prior capability (Folsom-Kovarik & Sinatra 2020).

Example Team Training Scenario

A team training scenario was created in Virtual Battlespace 3 (VBS3)[4]. The scenario is designed for play by a U.S. Army squad which consists of a squad leader and two fire teams of four Soldiers each. Each fire team contains a fire team leader, a rifleman, an automatic rifleman, and a grenadier. The squad also has a medic attached for this particular mission due to anticipated contact with hostile forces. Most of the static roles, like the medic or the differently equipped Soldiers, can also be assigned before training to control by the VBS3 Artificial Intelligence (AI) in order to enable team training for smaller groups.

[4] https://bisimulations.com/products/vbs3

The scenario training objective is for the squad to respond to the loss of the squad leader who is gravely wounded by a sniper. Team actions that require dynamic roles include responding to hostile contact, providing TC3 medical care under fire (CUF), and continuing the interrupted mission as a coordinated team.

The mission as briefed to the learners, before the sniper casualty, is to capture a high-value target (HVT) located by intel in a house within the small village (Figure 1). The HVT and armed hostiles (not initially known to the trainees) are indicated by the red diamond to the left of the map, while the scenario begins with the squad at the location of the blue rectangle on the right of the map. A second Army squad provides security and is not displayed.

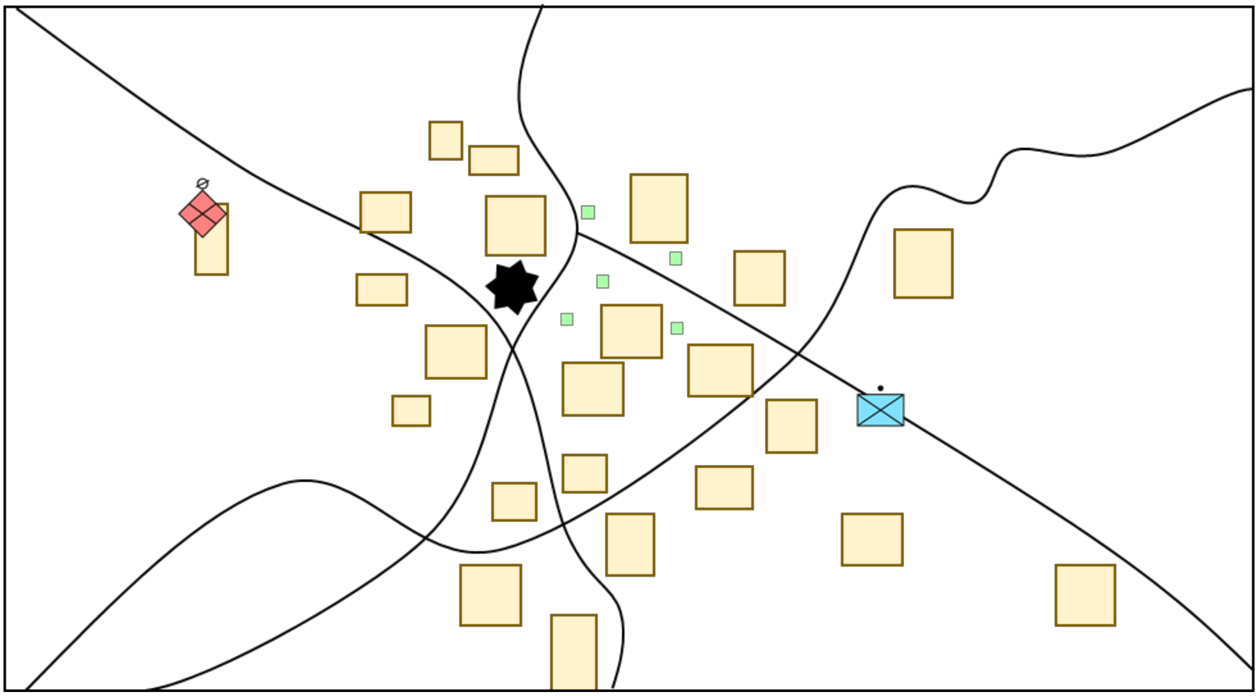


Figure 12: VBS scenario where dynamic role changes during execution help to automate team assessment.

The phases of the mission include movement through the village, the loss of the team's leader function requiring dynamic role changes and actions in response, movement to the HVT's building, and entering and clearing the HVT's building. First, during the initial phase of the mission, Soldiers will observe prebriefed indicators of hostile presence in the village (presence of vehicles near the target building and behavior of the green civilians in Figure 1). These provide an opportunity to evaluate the information exchange dimension of TDT using observed voice comms within the team.

Second, the star icon near the center of Figure 1 indicates a VBS location trigger. As the squad proceeds through town and reaches this location, the squad leader is incapacitated by a rifle fired from the HVT's building. The expected response is for members of the team to take cover, return fire, report the squad is in contact, and pass information about the casualty to the medic. Using proper voice and radio communications, terminology, and form at this point helps evaluate the team's communication delivery within TDT. Immediate CUF while the squad has suppressed the sniper by returning fire includes moving the squad leader to the cover of the nearby building and transferring care from the first responder Soldier to the medic.

Third, the squad continues the capture mission with a reduction in forces due to the injury and caring for the casualty. The squad moves to the HVT's building while providing cover, as well as coordinating the changing movement and cover roles as a team. The squad stacks on the building door and enters the building, clearing 2-3 rooms with coordinated timing and movement through the room. These actions provide opportunities to demonstrate supporting behavior and leadership within TDT. Once the building is clear, the scenario ends and the learners move to a structured after-action review (AAR).

The team training scenario enables assessments based on a subset of behavioral markers in (Sottilare et al., 2018) and the TDT operationalization used for AAR in Squad Overmatch (Townsend et al., 2018). The implemented assessments are as follows:

Table 1: Implemented team assessments.

Team Dimension	Domain Expression	GIFT Assessments
Information Exchange	Anticipate information needs	Call out prebriefed indicators around civilians and vehicles
	Provide situation updates	Call out hostile presence before sniper event
Communication Delivery	Provide complete reports	Report contact Pass info to medic
	Use clear communications	Proper terminology and order of information Limit overlap and interruption
Supporting Behavior	Provide backup	Assume casualty's roles and responsibilities
	Request backup	Coordinate movement and covering fire Do not enter early or without support
Initiative Leadership	Provide guidance	Leader orders team response Not providing CUF or chasing sniper
	Stay on mission	Cover assigned angles outside and inside building Not multiple people providing CUF or watching

The assessments surrounding information exchange and communication delivery are collected by observer conditions, while the assessments surrounding supporting behavior and leadership are automated by GIFT conditions. The observer conditions enable assessing team performance, assigning individual contributions, and providing adaptive AAR for all the dimensions of TDT. The automated assessments demonstrate how existing and new GIFT conditions can implement the assessments with code examples. The automated assessments are the focus of the technical explanation in the present paper as they assign roles to individuals and use the roles to assess team functional resilience after a casualty.

Technical Design for Dynamic Role Assignment

The technical underpinnings of the ability to reason about team roles and responsibilities are extensions to existing GIFT components and new software components that enhance the existing GIFT scenario configuration and processing of messages generated during training execution.

The focus of the current paper is on the technical implementation of dynamic team roles and responsibilities using extensions to scenario configuration using the DKF, GIFT conditions, and a new working memory component. Subcomponents of the technical description include roles, role rules, responsibilities, and the resulting role bindings to individuals. Working memory in the context of this work refers to a software component that transforms ongoing updates streaming in from the simulation environment, learner actions, and instructor observations into a semantically meaningful representation of facts about the world and the progress of training. Working memory acts as a “common operating picture” or shared interpretation of what is important to know about the training from moment to moment. It provides context for GIFT condition classes and assessments and was introduced as an experimental feature in (Folsom-Kovarik & Sinatra 2020). The enhanced software components that maintain the working memory are extended condition classes, a world state reasoner, and a new dynamic role manager.

Dynamic Role Management

In previous work, we added the capability to assign multiple static roles to team members, assigned at the beginning of a scenario. Team members could be selected by role for evaluation by a Condition, with that role being specified in a field of the Condition in the DKF.

In our previous example scenario, we found this approach adequate for training, but it limited the flexibility of the team during the scenario. In that scenario, one task performed by the team was to enter and clear a room occupied by hostile combatants. We were able to define roles that indicated which part of the room each team member should cover based on their order of entry into the room. However, since the roles were defined statically at the beginning of the scenario, our scenario would not work correctly if the team entered the room in a different ordering.

Recently, we have focused on adding Dynamic Role Management functionality to GIFT. With dynamic roles, our system can better handle situations like the enter-and-clear-room example described above. In that scenario, the system will now assign entry roles based on the actual order of when a team member enters the room, instead of using an ordering that is defined beforehand. This is an addition that makes the training scenario more robust and less prone to report “errors” that are not really errors, but simply deviations from an arbitrary script assigned before training starts.

Furthermore, dynamic roles give us the capability to continue to perform some assessments, including new kinds of assessments, when failures occur or team members become unable to perform their assigned tasks. For instance, if a team member becomes injured and unable to perform a necessary task, that member’s role can be reassigned to another nearby team member, allowing that member to perform the task. Similarly, a Fire Team Alpha Leader role can be reassigned according to chain of command rules if the fire team leader becomes injured. Once these roles are transferred, the new team members can be assessed based on their performance in place of the previous assignee.

The Role Manager is being updated to support assigning unfilled roles based on team members performing the responsibilities defined for those roles, as well as scenario-based situational rules. The Role Manager has also been updated to be able to read from Working Memory, and store new details in the Working Memory about current role assignments, role changes, and timing information.

To support our new Conditions and role-assignment capabilities, we updated the World State Updater to track more information provided by the VBS world state, such as injury status and current pose (standing, kneeling, crawling).

We have also added a number of new Conditions that build on dynamic roles, as well as some more traditional Conditions, and at least one utility Condition.

Scenario Configuration Enhancements

Within the GIFT domain module, Condition classes are modified with structures that operate on team roles and responsibilities as identifiers that can match different individuals based on the current state of the simulation world. Conditions can use the identity of the individual currently identified with a dynamic role to calculate assessments, just like trainees assigned to static roles at the start of the training scenario. Identifiers can be flexibly and efficiently described with additional elements in the DKF.

We have added new constructs to the DKF in the form of XML elements that correspond to key concepts enabling dynamic role assignment:

- *Role*: A role that can be assigned to a team member.

- *Responsibility*: A responsibility that the role is meant to fulfill. Specified in predicate logic using available predicates and variables.

- *AssignmentRule*: An optional field of the Role construct that can specify how the role is assigned, if it is not assigned by responsibility fulfillment. Also specified using predicate logic, using available predicates and variables. Predicates used for this field tend to be situational and scenario-based, e.g. closestTo(point).

We have also added *Role* as a member to the Condition DKF structure, to allow Conditions to select team members based on their role.

We have added two new Condition triggers:

- The *WorkingMemoryTrigger* is parameterized with a detail name and value, and activates when the detail exists in Working Memory with the specified value.

- The *RoleChangeTrigger* is parameterized with a role name, and whether it has become assigned or unassigned. It is activated when the specified role becomes assigned or unassigned (depending on how it is parameterized).

Lastly, we have added new Conditions that can be specified in a DKF and used in a scenario:

- The *WorkingMemoryCondition* assesses the team based on values that are set in the Working Memory. Other Conditions can inject details into Working Memory that this Condition then uses in aggregate to make an assessment.

- The *RoleChangeCondition* is a new type of GIFT Condition that is triggered by a role change. Once triggered, it evaluates if the team member selected for the role is performing in that role adequately. This Condition is evaluated based on if the new assignee is performing the responsibilities of a role.

- The *CheckDelayOfRoleChangeCondition* is another new type of GIFT Condition, also triggered by a role change. It is evaluated based on how long the role was left unfilled, and is a team-based measure indicating team cohesion and readiness.

- The *HealthCondition* is used primarily to inject details into Working Memory related to the health status of team members and others, which can then be used for role assignment.

- The *RoleChangeWatcherCondition* is used to keep track of whether a team member is monitoring a specified direction for enemy contact. It allows for role reassignment.

- The *RoleDisplayCondition* is a utility Condition that is used primarily to display current role assignments in the Game Master. Its functionality is intended to be built into the Game Master.

These additions to GIFT allow for a richer scenario design, and are driven by the needs of our working scenario. However, these new capabilities also come with an increase in complexity to scenario design. Since we chose a predicate-logic scripting approach for defining role rules and responsibilities, this requires a knowledge of the available library of predicates and Working Memory details. It also requires in some cases for Conditions to publish into Working Memory details that are useful to these predicates, especially simpler predicates that look for specific Working Memory detail values.

In order to make full use of them, a scenario designer needs to consider what can go wrong in the scenario, and how corrective actions can repair failures of team members. Responsibilities must be considered and encoded in the DKF, though it is likely a library of these can be built up and reused over time.

Some limitations to our approach are related to the use of VBS3. In the interest of time, some Conditions we have used are specified as Observed Conditions instead of being fully automated. In particular, an observer must evaluate team communications and assessments based on communication. Automating assessment of team communication is future work and can be based on modern speech recognition and natural-language understanding.

Another limitation we worked around was specifying chain-of-command rules inside of the DKF, because that data is not present in VBS3 state data. In the future, the chain of command could be separated into a reusable definition that applies to many or all military training. This could be enabled with the ability to load and deconflict multiple DKFs for each scenario.

Team Role Assignment Based on Scenario Context

Whenever individuals change roles, the information is accessible to other GIFT conditions by virtue of a shared working memory. Working memory is an experimental enhancement that lets GIFT Conditions make assessments based on a common operating picture of a simulated world or any training system state, semantically labeled memory of past world states and current context, and shared knowledge about coordinated action. Changing roles can therefore be assigned based on world facts such as: 1) trainees' current equipment or posture, 2) meaningful locations in the sim world, 3) individual actions in the context of other teammates or the scenario state.

The Reasoner, the World State Updater, and the Working Memory each provide information to the Role Manager to facilitate the assignment of dynamic roles. The Working Memory contains persistent queryable details about the shared world state, including details provided by VBS3 state info via the World State Updater, GIFT, Conditions, and computation/aggregation functions in the Reasoner.

The Reasoner contains script evaluation code that evaluates responsibilities and role rules. It can reference data items in the current world state and the Working Memory. It also contains the library of predicates that can be used by responsibilities and role rules.

Some example predicates that are used by the system, and exposed through the scripting engine contained in the Reasoner:

- *appliesTourniquets()* checks if an entity has applied a tourniquet. It is used by the FirstResponder responsibility.

- *isClosestTo(point)* checks to see if this entity is the closest entity to a given point. Mainly used in role rules after an event has occurred.

- *isClosestTo(entityName)* checks to see if this entity is the closest entity to a given entity. Mainly used in role rules after an event has occurred.

- *workingMemoryDetailExists(detailName)* checks for existence of a detail in Working Memory.

Figure 2 shows how data flows through the system to effect a role change based on changing Working Memory details.

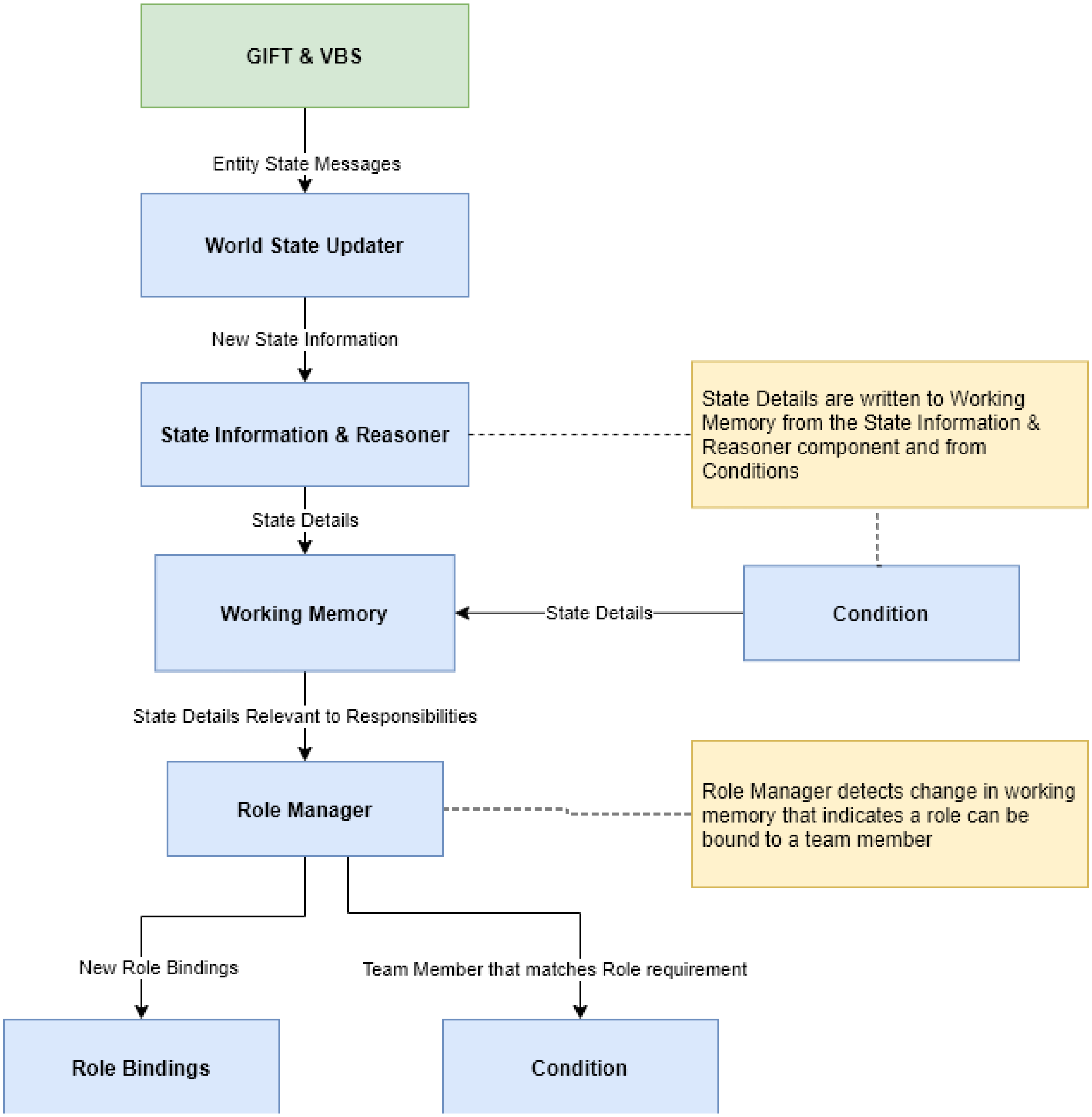

GIFT & VBS
Entity State Messages
World State Updater
New State Information
State Information & Reasoner
State Details are written to Working Memory from the State Information & Reasoner component and from Conditions
State Details
Working Memory
State Details
Condition
State Details Relevant to Responsibilities
Role Manager
Role Manager detects change in working memory that indicates a role can be bound to a team member
New Role Bindings
Team Member that matches Role requirement
Role Bindings
Condition

Figure 13: Role change evaluation dataflows.

Figure 3 shows how data flows through the system to evaluate a responsibility.

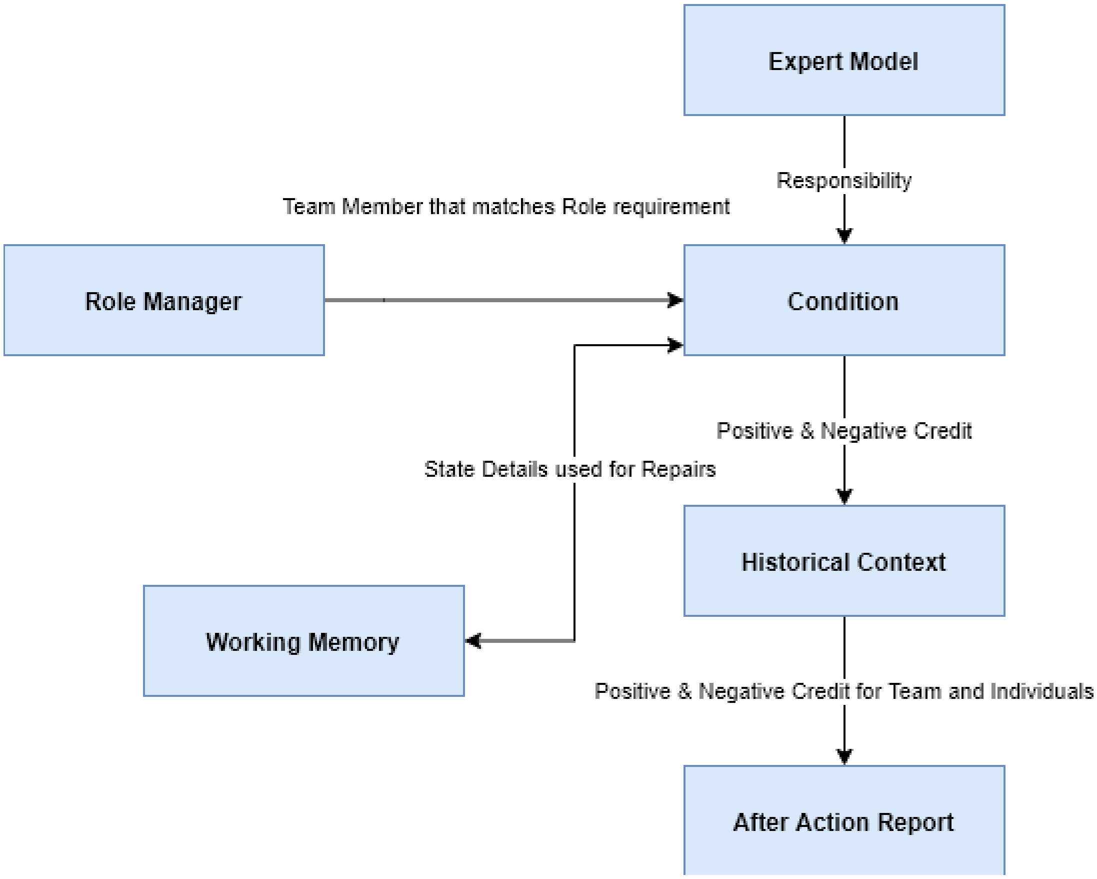

Expert Model
Responsibility
Team Member that matches Role requirement
Role Manager
Condition
Positive & Negative Credit
State Details used for Repairs
Historical Context
Working Memory
Positive & Negative Credit for Team and Individuals
After Action Report

Figure 14: Responsibility evaluation dataflows.

Team Role Assignment Based on Trainee Actions

Assumed roles need to be detected by interpreting trainee actions during training. The Role Manager uses GIFT conditions to detect actions that correspond with responsibilities in the DKF. When any trainee takes initial action to fill a role or corrective action to replace a role, GIFT can identify the action with the responsibility in the DKF and bind the team member to the corresponding role described in the DKF.

The combination of identifying team roles with responsibilities and comparing trainee actions to responsibilities enhances the ability of the DKF to encode a flexible expert model that is applicable in many different scenarios. Defining training context in terms of world states lets a given DKF apply to more VBS scenarios without needing revision. Importantly for enabling dynamic team roles, the expert model functions in a bidirectional manner. It can both evaluate when team function is degraded, because an individual does not fulfill a responsibility, and in the other direction it can identify when another trainee takes on the missing responsibility and is therefore acting to fill the missing role. This enables defining conditions that assess the time to repair a degraded function and the correctness of choosing the person who makes the repair.

As has been discussed, responsibilities defined in the DKF are associated with team roles rather than team members. The difference is minor when GIFT evaluates what we call the forward direction of the bidirectional expert model. Entities moving within a training simulation must be associated with a learner in order to update the learner model with assessments. Just as GIFT typically uses code inside Conditions to find the simulation entity associated with a team member, it uses code in the same form to find the simulation entity associated with a team role. For example, the squad leader in a VBS scenario might be the VBS entity with ID 113. The traditional or forward direction of reasoning about assessments is to identify whether the learner controlling entity 113 is demonstrating the Tasks and Concepts expected of a squad leader as determined by Conditions that monitor that entity (and, in the experimental GIFT, the world state for context).

There is an opportunity to evaluate the same DKF information in the opposite direction. That is to say, if the DKF defines actions that will be assessed "at expectations" for squad leader Tasks, then it is possible to infer that an entity executing those actions is now acting in the role of a squad leader. The second direction or so-called backward reasoning about the DKF expert model is important in cases where the learner who is in a given role is expected to change. This is the case in effective teams where team members can share the work of another person to lighten their load or can choose to assume a team function when there is more than one possible distribution of labor or when corrective action is needed to repair a loss.

The mechanism for detecting role changes and dynamically assigning assumed roles is the Role Manager. This component was added in the experimental GIFT previously, but it did not change role assignments. Now, the component will switch learner roles when facts about the world change, such as the location or medical state of their simulated character. In the present implementation, the Role Manager only assigns assumed roles when roles are empty. This means it is not yet able to detect and reason about conflicts such as too many learners taking on the same role. The two ways roles can be empty are when a Condition registers a new role or responsibility comes into scope because of scenario events or Tasks, or when a learner is explicitly removed from a role. A learner can be removed from a role with scenario-specific logic, such as when the squad leader becomes a casualty. A new role comes into scope in the same way as the DKF defines the scope of Conditions, usually starting and ending with scenario events in the VBS setting.

Next, a RoleChangeCondition monitors and updates role assignments using the usual access to simulation messages and the interpreted world state available to all GIFT Conditions. The RoleChangeCondition has been described above as a trigger for any required processing when learners assume roles through their actions. Based on this trigger, Conditions can register assessments such as the delay or time needed to fill a compromised role, or the correctness of the person who filled the role. As one example, the chain of command tells who ought to take over when the squad leader is a casualty. If a different person takes over, this is assessed "below expectations" for both the individual action and the team Leadership dimension.

Discussion

This paper discussed authoring as a key goal of GIFT. The GIFT representation of concepts such as team roles and responsibilities contributes to determining what authors can easily express when they configure training. The same ideas can be easy or hard for authors to express depending in part on how well GIFT's representation aligns with the way end users think about training.

Research questions remain regarding the proper level of expressivity that can be presented and edited in the GIFT authoring tools without requiring a programmer level of knowledge to make changes. Roles in an example scenario have already required functions like union, intersection, difference, and sequential composition. Possible future methods to reduce the complexity of these requirements for GIFT end users might be to visualize the affected personnel and relevant features instead of the underlying logic, or to provide a library of roles typical to the Army which are defined robustly enough to reuse across scenarios.

The example implementation offers a structure to let GIFT reason about team performance, including process and outcomes. The generality of the approach is supported in two ways. First, the team assessments from prior work are being reused in a new VBS scenario, showing that they are not specific to one terrain or team structure. Second, the new conditions and assessments created for this year continue to use the TDT framework, observable TARGETs, and structured AAR support. With these existing and new experimental changes, GIFT is able to assign credit and link team performance to individual contributions.

Another open question that remains is applying the new dynamic roles in more ambiguous settings. The current assumption that assumed roles will be empty simplifies the need to deconflict roles. As soon as an assumed role is dynamically assigned, it is not re-evaluated. In reality, assumed roles can change as team members hand off or divide responsibilities. There is also a separate question of ambiguity in the state of the world. Currently, the world state reasoner is not challenged by conflicting inputs, fog of war, or other reasons for uncertainty. This aligns with the fact that VBS scenarios in GIFT typically do not include sensor models that would lead to different people having false beliefs about enemy locations, movement, or capability. Even though this is true, ambiguities that arise from team behavior such as imperfect information sharing should be included in future work. These may be possible to incorporate using more Conditions without needing new assessment types, but more research is needed to show this is the case.

Dynamic assignment was shown to work in an open-world training scenario. It reduced some authoring effort, for example the author is not required to assign each member of the room entry team to the order they enter because GIFT can detect that during scenario execution with location-based role assignments. When GIFT Conditions use the context available in the world state and history, they also have the potential to be reused in more scenarios without adding to the authoring effort.

A challenge identified in the current work is that the logical relationships needed to define dynamic roles can become complex and might not be acceptable for non-technical authors to enter into a DKF. Instead, different authoring paradigms might be needed such as letting authors see examples of who is affected by combinations of roles, or enter examples of who should be affected, letting GIFT work out the logic to generalize the example. This would require future work to determine feasibility and user acceptance.

Conclusions and Recommendations for Future Research

This work introduced a method for reasoning about the many roles that team members fill during team training. The roles are assigned before and during training, and when the roles are dynamically assigned during training they can lead to assessments about individual and team performance.

The suggested enhancements to GIFT have been demonstrated with an established framework for providing team assessment and adaptive feedback, TDT. The needs of TDT assessment include understanding how a team communicates, shares information, and supports each other when team members need help. To meet these needs, the experimental GIFT contains software components and structures designed around team roles and, now, dynamic role assignments. These let end users author descriptions of team behavior in ways that align with their thinking and that support inference about team process from observable outcomes.

Near-term recommendations center on completing the implementation described here and socializing the design decisions with the GIFT community. The present paper helps to explain the proposed functional and technical approach. The status of the work is under active implementation, meaning that community input to design considerations and needs is welcome. During our ongoing implementation work, another near-term recommendation is to continue making team assessments and conditions ever more robust to different contexts using the capabilities described here. The goal is to show that advanced team assessments are possible to reuse across scenarios and, in future work, across domains because they describe team skills and performance that transcend specific scenarios.

Long-term recommendations build on the current work to add new capabilities in GIFT. One example might be to use the team assessments to enable scenario tailoring in response to team performance. Tailoring to adjust challenge levels might be easier, and a harder task might be tailoring the scenario to repair functional loss so that training can continue. This would be enabled with the team function detection in this work. It might even become possible to make the same GIFT logic drive the behavior of autonomous teammates who can reason about helping to repair team function.

In conclusion, reasoning about team roles and responsibilities in GIFT enables advanced team assessment and has the potential to enable future research into highly adaptive training.

References

Belbin, R.M. (1981). *Management Teams: Why They Succeed or Fail*. London: Heinemann.

Fisher, S., Hunter, T., & Macrosson, W. (1998). The structure of Belbin's team roles. *Journal of occupational and organizational psychology, 71*(3), 283-288.

Folsom-Kovarik, J.T., & Sinatra, A.M. (2020). *Automating Assessment and Feedback for Teamwork to Operationalize Team Functional Resilience.* Paper presented at the 8th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym).

Fowlkes, J., & Shawn, C. (2004). Targeted acceptable responses to generated events or tasks (TARGETs). *Handbook of human factors and ergonomics methods*, 47-1.

Georgeff, M., Pell, B., Pollack, M., Tambe, M., & Wooldridge, M. (1998). *The belief-desire-intention model of agency.* Paper presented at the International workshop on agent theories, architectures, and languages.

Gilbert, S. B., Slavina, A., Dorneich, M. C., Sinatra, A. M., Bonner, D., Johnston, J., ... & Winer, E. (2018). Creating a team tutor using GIFT. *International Journal of Artificial Intelligence in Education*, *28*(2), 286-313.

Johnston, J.H., Phillips, H.L., Milham, L.M., Riddle, D.L., Townsend, L.N., DeCostanza, A.H., et al. (2019). A Team Training Field Research Study: Extending a Theory of Team Development. *Frontiers in Psychology, 10*, 1480.

McCormack, R., Kilcullen, T., Sinatra, A.M., Case, A., & Howard, D. (2019). *Teamwork Training Architecture, Scenarios, and Measures in GIFT.* Paper presented at the 7th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym).

McCormack, R., Case, A., Howard, D., Logue, J., Kay, K., & Sinatra, A. M. (2020, May). Teamwork Training in GIFT: Updates on Measurement and Audio Analysis. In *Proceedings of the 8th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym8)* (p. 155). US Army Combat Capabilities Development Command–Soldier Center.

Neville, K., Chen, D.-W., Massey, L., Cowell, T., Burbelo, G.A., Blankenbeckler, P.N., Normand, S.L., & Uhl, E.R. (2020). *A Cognitive Skills Research Framework for Complex Operational Environments* (Technical Report 1379). Fort Belvoir, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Peeters, M.A., Van Tuijl, H.F., Rutte, C.G., & Reymen, I.M. (2006). Personality and team performance: a meta-analysis. *European journal of personality, 20*(5), 377-396.

Ouverson, K. M., Ostrander, A. G., Walton, J., Kohl, A., Gilbert, S. B., Dorneich, M. C., ... & Sinatra, A. M. (2021). Analysis of Communication, Team Situational Awareness, and Feedback in a Three-Person Intelligent Team Tutoring System. *Frontiers in Psychology*, *12*.

Rowe, J., Spain, R., Goldberg, B., Pokorny, R., Mott, B., & Lester, J. (2020). *Toward Data-Driven Models of Team Feedback in Synthetic Training Environments with GIFT.* Paper presented at the 8th Annual Generalized Intelligent Framework for Tutoring (GIFT) Users Symposium (GIFTSym).

Smith-Jentsch, K.A., Cannon-Bowers, J.A., Tannenbaum, S.I., & Salas, E. (2008). Guided team self-correction: Impacts on team mental models, processes, and effectiveness. *Small Group Research, 39*(3), 303-327.

Smith-Jentsch, K.A., Zeisig, R.L., Acton, B., & McPherson, J.A. (1998). Team dimensional training: a strategy for guided team self-correction. In J. A. Cannon-Bowers & E. Salas (Eds.), *Making decisions under stress: Implications for individual and team training*.

Sottilare, R. A., Brawner, K. W., Sinatra, A. M., & Johnston, J. H. (2017). An updated concept for a Generalized Intelligent Framework for Tutoring (GIFT). *GIFTtutoring. org*, 1-19.

Sottilare, R. A., Burke, C. S., Salas, E., Sinatra, A. M., Johnston, J. H., & Gilbert, S. B. (2018). Designing adaptive instruction for teams: A meta-analysis. *Int'l Journal of Artificial Intelligence in Education*, *28*(2), 225-264.

Tambe, M. (1997). Towards flexible teamwork. *Journal of artificial intelligence research, 7*, 83-124.

Thomson, A.M., Perry, J.L., & Miller, T.K. (2009). Conceptualizing and measuring collaboration. *Journal of public administration research and theory, 19*(1), 23-56.

Townsend, L., Johnston, J., Ross, W.A., Milham, L., Riddle, D., & Phillips, H. (2018). *An Integrated After Action Review (IAAR) Approach: Conducting AARs for Scenario-Based Training Across Multiple and Distinct Skill Areas.* Paper presented at the Int'l Conference on Engineering Psychology and Cognitive Ergonomics.

Acknowledgement

This material is based upon work supported by the U.S. Army Combat Capabilities Development Command Soldier Center (DEVCOM SC) SFC Paul Ray Smith Simulation & Training Technology Center (STTC) under Contract No. W912CG-19-C-0009. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the DEVCOM-SC STTC or the U.S. Government.

About the Authors

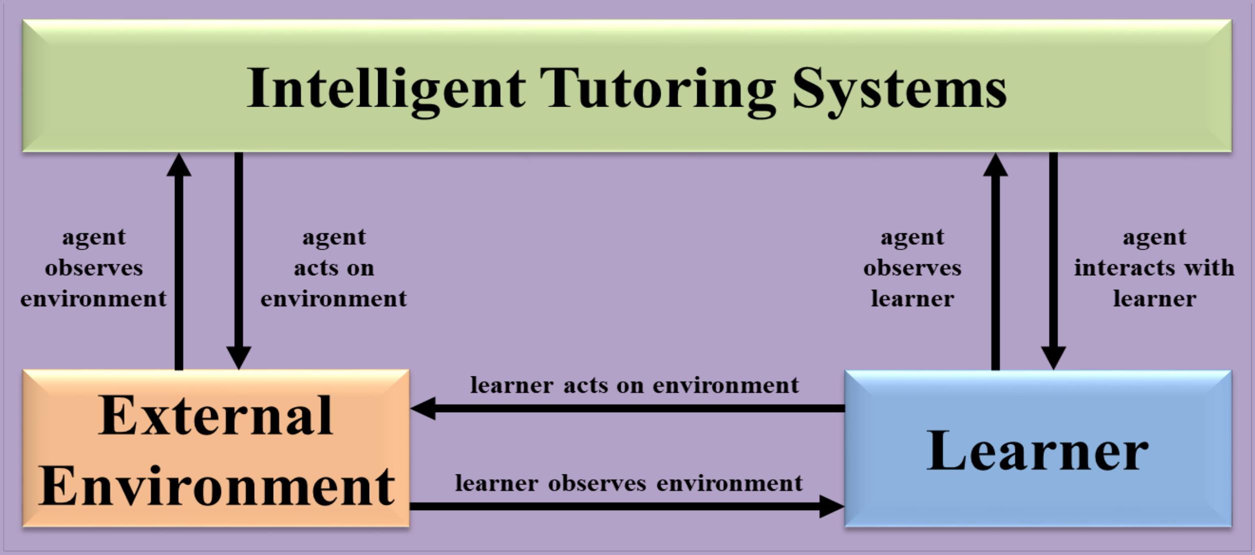
***Dr. J.T. Folsom-Kovarik** is a Senior Scientist with Soar Technology, Inc. His research focuses on computational tools for adaptive learning that work under real-world constraints such as efficient representation of complex real-world environments, efficient use of limited data for modeling, and transparency and control for end users.*

***Mr. Joel Sieh** is a Software Engineer IV with Soar Technology, Inc. His background includes both technical leadership and hands-on architecture, design, and implementation within the GIFT and other software environments.*

***Dr. Anne M. Sinatra** is a Research Psychologist, and part of the adaptive training research team within DEVCOM Soldier Center – STTC. She works on the Generalized Intelligent Framework for Tutoring (GIFT) project. Her background is in Human Factors and Cognitive Psychology.*

Proceedings of the Ninth 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 technologies for evaluating the effectiveness of tutoring applications. 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.


Intelligent Tutoring Systems
agent observes environment
agent acts on environment
agent observes learner
agent interacts with learner
External Environment
learner acts on environment
learner observes environment
Learner

The GIFT Users Symposia began 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 Editor:

- ***Dr. Benjamin S. Goldberg*** *leads adaptive training research at the U.S. Army Combat Capability Development Command – Solider Center and is a co-creator of the Generalized Intelligent Framework for Tutoring (GIFT).*

Part of the Adaptive Tutoring Series

9 780997 725797